NLP4MusA 2026

The 4th Workshop on NLP for Music and Audio

Proceedings of the Workshop

March 28, 2026

The NLP4MusA organizers gratefully acknowledge the support from the following sponsors.

Sponsors

DEEZER SiriusXM Spotify

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
317 Sidney Baker St. S
Suite 400 - 134
Kerrville, TX 78028
USA
Tel: +1-855-225-1962
acl@aclweb.org

ISBN 979-8-89176-369-2

Introduction

Welcome to the 4th Workshop on NLP for Music and Audio (NLP4MusA 2026)! The workshop is held in Rabat, Morocco, in conjunction with EACL 2026, and spans half a day, featuring four keynote talks followed by lightning presentations of accepted papers and a poster session.
Building on a tradition of cross-disciplinary impact, the intersection of NLP with music and audio-based creative media presents a frontier full of unique challenges and exciting opportunities. The Fourth Workshop on Natural Language Processing for Music and Audio (NLP4MusA) aims to explore the multimodal synergies between language, music, and sound. As NLP increasingly enables domains where language and interaction converge, the entertainment industry offers a particularly compelling case: most audio content - such as songs or podcasts - contains an inherent linguistic dimension, while user engagement often occurs through language, from search queries to social media conversations. Recent advances in language modeling and multimodal representation learning have deepened the connection between NLP and music, giving rise to a new generation of applications. These range from music transcription and captioning to the generation of lyrics, symbolic compositions, and even complex audio signals from text-based prompts. At the same time, techniques from Question Answering and Information Retrieval are transforming how people search for, recommend, and interact with music and audio. Because language also reflects culture and multilingual diversity, NLP-driven approaches open pathways for cross-cultural discovery and creativity - enriching the way we experience and understand music and sound.
The workshop focuses on topics such as:
1) NLP for Music and Audio Understanding
- Music Tagging and Auto-tagging, Knowledge Graph Construction, Semantic Ontologies
- Information Extraction, Named Entity Recognition, and Entity Linking
- Multimodal Representation Learning, Lyrics and Symbolic Representation Analysis
- Emotion and Sentiment Analysis, Culture-specific Music Understanding, Corpora Bias
- Music Captioning and Description Generation
2) NLP for Music Retrieval or Recommendation
- Conversational Interfaces, Query understanding and Intent Prediction
- Multimodal, Cross-modal Music Information Retrieval and Recommender Systems
- Natural Language User Modeling
- Music Question Answering
- Fairness and Transparency
3) NLP for Music and Audio Generation
- Lyrics Generation, Audio/Symbolic Query-driven Music Generation
- Synthetic Music Content Detection
In response to our call for papers, we received 19 submissions. After an initial screening, two submissions were desk-rejected and one was eventually withdrawn by the authors. Of the remaining submissions, each was rigorously reviewed by at least two Program Committee members selected for their expertise. Based on the reviewers' feedback, we accepted 10 papers (53% acceptance rate), of which one was not included in the proceedings due to a double submission.
We are extremely grateful to the authors for their valuable contributions and to the Program Committee members for their detailed and constructive reviews. We also thank our sponsors and the EACL Workshop Chairs for their support and prompt responses.
We hope you find the workshop insightful and inspiring!
Elena V. Epure, Sergio Oramas, SeungHeon Doh, Pedro Ramoneda, Anna Kruspe, Mohamed Sordo

Organizing Committee

Organizing Committee

Elena Epure, Deezer (France) / Idiap Research Institute (Switzerland)
Sergio Oramas, SiriusXM (Spain)
SeungHeon Doh, KAIST (South Korea)
Pedro Ramoneda, Universitat Pompeu Fabra / Songscription (Spain)
Anna Kruspe, Munich University of Applied Sciences (Germany)
Mohamed Sordo, SiriusXM (USA)

Program Committee

Reviewers

Anna Aljanaki, University of Tartu
Hayeon Bang, KAIST
Morteza Behrooz, Meta AI
Hans-Ulrich Berendes, AudioLabs Erlangen
Dimitrios Bralios, University of Illinois Urbana-Champaign
Eunjin Choi, KAIST
Matthew Davies, Sirius XM
SeungHeon Doh, KAIST
Robie Gonzales, Dalhousie University
Simon Hachmeier, Humboldt-Universität zu Berlin
Mikaela Keller, Université de Lille
Haven Kim, University of California, San Diego
Soham Korade, IIIT Hyderabad
Daeyong Kwon, KAIST
Dinh-Viet-Toan Le, Université de Lille
Jongpil Lee, Neutune
Junwon Lee, KAIST
Linxi Li, University of Warwick
Pasquale Lisena, EURECOM
Benno Weck, Universitat Pompeu Fabra
Zachary Novack, UC San Diego
Julian D Parker, Stability AI
Amit Namburi, UC San Diego
Yusuke Sakai, NAIST
Simon Schwär, AudioLabs Erlangen
Suswara Pochampally, IIITH
Rosa Stern, Sonos
Giulio Prevedello, Sony CSL
Yongyi Zang, Smule, Inc.
Shuo Zhang, Tufts University
Yixiao Zhang, ByteDance Inc.
Kongmeng Liew, University of Canterbury
Kento Watanabe, AIST
Aleksandr Petrov, Spotify
Mucun Tian, Amazon

Keynote Talk
Invited 1

Willem Zuidema
University of Amsterdam

Bio: Willem Zuidema is associate professor of NLP and Explainable AI at the Institute for Logic, Language and Computation (ILLC) in Amsterdam. He has published widely on computational models of language, including comparisons with music, animal communication and explorations of their evolutionary origins. He has done pioneering work in deep learning for NLP and interpretability methods for LSTMs and Transformers. He leads the InDeep consortium, focused on interpretability for text, speech and music, involving 7 PhD students and 5 universities in the Netherlands.

Keynote Talk
Invited 2

Harin Lee
University of Cambridge

Bio: Harin Lee is a multidisciplinary researcher combining large-scale data analysis with cross-cultural experiments to study the psychological foundations of music cognition and musical diversity worldwide. His research includes field experiments with Tsimané villagers in the Bolivian Amazon and developing human-in-the-loop online paradigms to investigate cultural evolution in artificial worlds. Currently a Junior Research Fellow at University of Cambridge, he earned his PhD at the Max Planck Institute for Human Cognitive and Brain Sciences.

Keynote Talk
Invited 3

Ilaria Manco
Google DeepMind

Bio: laria Manco is a Research Scientist in the Magenta team at Google DeepMind. Her research spans music generation and understanding, with a current focus on new forms of musical interaction via controllable, real-time generative models. Ilaria received her PhD from Queen Mary University of London, where she developed multimodal representation learning approaches to connect music and language. During her doctoral work she also collaborated with Universal Music Group on large-scale audio-caption datasets and audio-language models.

Keynote Talk
Invited 4

Enrico Palumbo
Spotify

Bio: Enrico Palumbo researches and builds Generative AI features for Search and Recommendations at Spotify, with a focus on agentic technologies and generative recommendations. Before joining Spotify, he was a Research Scientist at Amazon, developing language understanding models for Alexa in non-English locales. He holds a PhD on Knowledge Graph Embeddings for Recommender Systems, carried out jointly between the Polytechnic University of Turin, EURECOM, and Links Foundation.

Table of Contents

From Novice to Expert: Generating Audience-Dependent Concert Moderations with RAG-LLMs

Kerstin Denecke
Bern University of Applied Sciences
Quellgasse 21, 2502 Biel
Switzerland
`kerstin.denecke@bfh.ch`

Abstract

In this paper, we study the capabilities of large language models (LLMs) to adapt a concert moderation to diverse expertise levels of listeners. Our proof-of-concept concert moderator is based on retrieval-augmented generation (RAG) and uses few-shot audience modelling to infer listener's expertise. We study the capabilities of the system to adapt to three different listener's expertise levels. Two open domain LLMs are compared: gpt-oss:20b and llama3. The recognised differences among the models suggest that they vary in how directly they reproduce versus paraphrase retrieved information while maintaining semantic alignment.

1 Introduction

Live concert moderation plays a crucial role in shaping listeners' understanding and engagement. However, listeners have a wide range of musical backgrounds, from novices who are unfamiliar with technical terminology to expert musicians who expect stylistic and analytical insight. Recent advances in large language models (LLMs), conversational interfaces, and query comprehension provide new possibilities for music recommendation (Yun and Lim, 2025; Epure et al., 2025). They showed to help listeners to clarify implicit needs and support unique exploration of music (Yun and Lim, 2025). They can also help in modelling user preferences, listening intents, emotion and mood (Epure et al., 2025). Based on these initial results, we hypothesise that LLMs can be used for concert moderation, i.e. generate personalised explanations to musical pieces, adapted to different audience groups, and able to respond to ad hoc questions from the audience.

Despite the growing interest in artificial intelligence-generated explanations and recommendation systems in cultural contexts (Singh et al., 2024), to the best of our knowledge no studies have examined LLMs as real-time concert moderators. General purpose LLMs were tested for music recommendation (Yun and Lim, 2025). In this work, we want to test retrieval-augmented generation (RAG) (Gao et al., 2023) to realise a concert moderator. RAG can ground LLMs in reliable musical sources, reducing hallucinations and enabling contextually rich commentary. The system integrates: (i) a structured knowledge base that combines musicological literature with piece-specific metadata, (ii) a dynamic audience modelling module that classifies listener expertise using few-shot prompting, and (iii) audience-adapted generation rules that control complexity, technical terminology, and explanatory depth. The moderator engages in conversation with the audience, offering piece introductions, intermission commentary, and live responses to audience inquiries.

2 Material and methods

In the following, we describe the system components. All components were implemented in Python and run locally using an Ollama-hosted LLM and a separate embedding model (`mxbai-embed-large`) for retrieval. We tested and compared to LLMs: `llama3` (Grattafiori et al., 2024), `gpt-oss:20b` (OpenAI, 2025).

2.1 Knowledge base

As knowledge base for the proof-of-concept implementation, we downloaded the *Cambridge Companion to the Organ* (Thistlethwaite and Webber, 1999). The volume contains chapters on organ music from different regions and historical periods (from before 1700 to after 1800), as well as two chapters describing the pipe organ as an instrument and the fundamentals of organ playing. This choice was pragmatic, as the PDFs are accessible with an institutional license; in future work and for a publicly available moderator, we plan to replace this

Proceedings of the 4th Workshop on NLP for Music and Audio (NLP4MusA 2026), pages 1–6

with copyright-free material.

All pages of each chapter were concatenated and split into overlapping text chunks of approximately 750 characters. Each chunk was stored in an in-memory document store together with metadata (document identifier, chapter title, section index). During system initialization, each chunk was embedded using the `mxbai-embed-large` embedding model via Ollama's `/api/embeddings` endpoint, and the resulting vectors were cached in a NumPy array for efficient nearest-neighbour search.

In addition to the textbook material, a hand-crafted JSON file specified the concert program and structured information about the performed pieces and composers. For each piece, the JSON file contained fields such as title, composer, year of composition, key, catalogue number, estimated duration, liturgical context, registration tips, short program notes, and anecdotes. For each composer, the file contained biographical metadata (name, dates, era) and a short biography with optional "fun facts". On startup, this JSON file was parsed into `Piece` and `Composer` data classes. Each instance was converted into a textual document (e.g., "Piece Title", "Composer", "Program Notes", "Registration Tips") and embedded into the same vector. This ensured that both textbook knowledge and concert-specific metadata were available to the RAG pipeline as separate, searchable sources.

2.2 Audience modelling module

To adapt explanations to different listener profiles, we implemented an explicit audience modelling module that infers an audience expertise level and conditions all subsequent generation on this profile.

At the beginning of each concert session (before the first piece introduction), the moderator poses three short calibration questions to the audience via the conversational interface: 1) A self-report of familiarity with classical/organ music (e.g., "almost none", "I go to some concerts", "I am a musician"), 2) A preference regarding explanation style (stories and imagery vs. musical theory and structure vs. a mix of both), 3) An open prompt to name a composer, piece, or musical term of interest (allowing "I'm not sure").

Audience responses are collected as free-text and concatenated into a short "survey" string. This string is then passed to the LLM in a dedicated classification prompt. The classifier prompt uses a few-shot strategy: it defined the target label set (novice / intermediate / expert), provides three illustrative survey–label pairs as examples (one per level), and then asks the model to output exactly one label for the current audience. The model was instructed to use conversational cues such as explicit self-description ("I'm an organist", "I just like the sound"), mention of technical terminology ("fugue", "modal language", "registration"), and the sophistication of the named interests to choose between the three levels. The resulting label is post-processed heuristically (e.g., normalising to lowercase and falling back to `novice` in case of malformed output) to obtain a single categorical expertise level for the current session. This level is then used to dynamically construct the system prompt for all subsequent LLM calls. Specifically, we defined a base system prompt describing the role of the moderator (warm, concise introductions; use of RAG context; avoidance of hallucinated facts). On top of this, we appended an audience-profile block that specified style and complexity constraints:

- **Novice**: avoid jargon; if a technical term is necessary, define it in one simple sentence; prioritise stories, analogies, and felt experience over theory.
- **Intermediate**: assume familiarity with basic musical concepts (e.g., keys, themes, fugue); mix narrative and light structural/harmonic comments; keep explanations compact.
- **Expert**: allow concise use of specialist terminology (e.g., registration schemes, modal language, contrapuntal techniques); focus on insight and specificity while maintaining brevity for live settings.

The same audience level conditioned both the piece moderations and the answers to spontaneous audience questions, ensuring consistent adaptation of explanatory depth across the entire concert.

2.3 Retrieval-augmented generation pipeline

The system's core interaction loop followed a retrieval-augmented generation paradigm. Given a generation task (e.g., "introduce the next piece" or "answer the following audience question"), the pipeline proceeded as follows:

1. **Query formulation.** For piece moderations, the system constructs a query containing the piece title, composer, and a short description

of the desired output (e.g., era/style, one listening tip, one factual highlight). For audience questions, the query consists of the question text itself.

2. **Vector retrieval.** The query is embedded with the same embedding model as the knowledge base. Cosine similarity between the query vector and all document vectors in the in-memory store is computed, and the top-k (default $k = 5$) most similar chunks are selected.

3. **Context construction.** The retrieved chunks are concatenated into a "context block" that prefixed each chunk with a simple label (e.g., piece title, composer name, or source file and section) and the retrieval score.

4. **LLM generation.** The final call to the chat model combines: (i) the audience-adapted system prompt, (ii) the context block with retrieved knowledge, and (iii) a task-specific user message (e.g., "Please introduce the next piece succinctly for a *novice/intermediate/expert* audience…" or "Audience question: …Answer briefly and accurately for a *novice/intermediate/expert* audience."). The LLM is instructed not to contradict the provided context and not to invent dates or catalogue numbers. Generation used a moderate temperature (0.6) to balance fluency and determinism.

In the current implementation, audience inputs are collected via a text-based terminal interface (standing in for speech recognition), while outputs from the LLM are both printed to the console and synthesized using an offline text-to-speech engine (`pyttsx3`). In a real-world deployment, the same back-end architecture can be paired with a microphone-based speech-to-text front end (e.g., Whisper) and a loudspeaker output, allowing the moderator to function as a live, voice-based guide between pieces, during Q&A segments, and in intermission commentary.

2.4 Evaluation of audience adaptation

With the implementation described above, we want to study whether LLMs can reliably adapt the style and complexity of musical explanations to different audience expertise levels and to what extent does RAG impact on factual grounding relative to a no-RAG condition.

For each piece in the concert program (comprising 2 pieces in this experiment), we automatically generated five versions of the moderator's output: EXPERT, INTERMEDIATE, NOVICE, NO RAG, GENERIC. For each piece, the system generated a concise moderation in all five conditions as well as answers to three standardised audience questions (e.g., "What should I listen for?", "Why is this piece important?", "What makes this difficult to play?"). All answers were generated by the two LLMs separately.

For judging the linguistic complexity, we measured total token count of response, number of sentences and average sentence length, type–token ratio (lexical diversity, calculated by number of unique words divided by number of words), and Flesch-Kincaid Grade Level and Flesch-Reading Ease Score (readability index, calculated with https://goodcalculators.com/flesch-kincaid-calculator/). We expected novice outputs to be structurally simpler and expert outputs to show higher complexity.

To evaluate how strongly answers relied on information retrieved from the knowledge base, we computed context-overlap rate as proportion of content words in the generated text that also appear in the retrieved context block, and similarity to context, defined as cosine similarity between embeddings of the output and its context. NO RAG outputs were expected to show substantially lower values on both measures.

To verify that changes induced by adaptation primarily affected style rather than factual content, we computed embedding-based cosine similarity between each adapted output and the corresponding GENERIC output for the same piece and question. High similarity indicates stable factual content, while observable stylistic shifts were expected across audience profiles.

3 Evaluation results

The results show that gpt-oss:20b answers have more tokens in average and are 1 sentence longer (see table 1). Overall, the sentences are longer than those generated by llama3. The type-token ratio is similar for both LLMs. We can recognise a larger context overlap in answers of llama3. The similarity to generic is quite high for both LLMs (gpt-oss:20b: 93-97%, llama3: 92-95%). The similarity to context is higher for answers by gpt-oss:20b. As expected, the Flesch-Kinaid Grade Level is higher

LLM	condition	tokens	sentences	average sentence length	type-token-ratio	context overlap	similarity to generic	similarity to context	Flesch-Kinaid Grade Level	Flesch Reading Ease Score
gpt-oss:20b	Expert	191.9	9	21.4	0.69	0.36	0.93	0.85	12.08	39.175
gpt-oss:20b	Intermediate	196.7	9.33	22.05	0.65	0.34	0.97	0.85	10.7	49.7
gpt-oss:20b	Novice	203.4	10.25	20.23	0.631	0.304	0.95	0.845	8.775	62.58
gpt-oss:20b	Generic	147.8	6.5	23.55	0.69	0.39	0.99	0.84	10.925	48.25
gpt-oss:20b	No Rag	129.8	7	19.56	0.74	0	0.93	0	10.01	51.96
Llama3	Expert	163.4	8.25	19.82	0.69	0.46	0.95	0.79	11.8	41.9
Llama3	Intermediate	175.9	8.75	20.24	0.644	0.41	0.95	0.79	9.93	55.575
Llama3	Novice	133	7.375	18.45	0.697	0.41	0.926	0.76	9.44	56.84
Llama3	Generic	142.9	7	20.69	0.67	0.42	1	0.782	10.225	53.45
Llama3	No Rag	112.4	5.75	19.45	0.75	0	0.924	0	10.875	48.34

Table 1: Evaluation results. Per model we averaged the results from 7 answers per condition. GENERIC: The base system prompt was used without any audience-profile block; RAG remained active. NOVICE: System prompt included the novice audience profile. INTERMEDIATE: System prompt included the intermediate audience profile. EXPERT: System prompt included the expert audience profile. NORAG: Same base system prompt as the GENERIC condition but with retrieval disabled; no context block was provided to the LLM.

for answers in expert level. For gpt-oss:20b answers, the three levels are well recognisable in a Flesch-Kinaid Grade Level increasing from novice to expert. In llama3 answers, the Flesch-Kinaid Grade Level for intermediate and novice-targeted answers are quite similar.

In terms of content provision it can be recognised that gpt-oss:20b creates answers often with enumerations (e.g. "Technical demands", "Registration:", "Tempo:"). In contrast, llama3 provides a well readable text without any explicit enumerations. Llama3 appears more empathetic with starting phrases such as "What a wonderful piece", while gpt-oss:20b answers are more factual. The complete evaluation results are available at Zenodo (https://10.5281/zenodo.18421171)

4 Discussion and conclusions

The evaluation results indicate that audience-adaptive generation largely behaves as intended. Outputs tailored to novice listeners were shorter, contained fewer sentences, and scored lower on readability indices, reflecting a simpler linguistic structure. Conversely, outputs targeted to experts showed higher token counts, more complex syntax, and higher Flesch–Kincaid Grade Levels. This confirms that the system modulates explanatory complexity across expertise levels.

The differences among the models suggest that models vary in how directly they reproduce versus paraphrase retrieved information while maintaining semantic alignment. At the same time, we observed high similarity between the outputs of the adapted conditions and the generic baseline. This indicates that audience adaptation primarily affects style and terminology rather than the underlying factual content. The high similarity-to-context scores across all RAG-enabled conditions support this, demonstrating that both models maintain strong grounding in the retrieved musicological sources. The combination of curated program metadata and thematically relevant PDFs establishes appropriate boundaries for the domain while enabling flexibility in answering audience questions. Future systems could also incorporate cultural background cues to adapt examples, metaphors, and contextual framing.

The content accuracy was not explicitly checked in our evaluation. In table 2, we show three examples of the generated replies to the question "What should I listen for in "Präludium und Fuge in C-Dur, BWV 545" by Johann Sebastian Bach?". We selected the three conditions "Expert", "Novice" and "No RAG". It can be seen that the language of the answer targeting novices is very metaphoric while in expert-level, it uses many domain-specific terms such as augmentation, inversion, ritornello-like. In terms of correctness, the novice-level answer can be considered correct - mostly because

No.	Answer
1	What a wonderful piece! As you listen to the Präludium und Fuge in C-Dur, BWV 545, I encourage you to pay attention to the way the melody flows like a river, building momentum and energy as it goes. You might also notice how the fugue section takes that same melody and weaves it together with other musical threads, creating a rich tapestry of sound. In simple terms, think of the prelude as a gentle stream that grows into a mighty river, and the fugue as the river's many tributaries coming together to create a beautiful, complex whole.
2	What a delight to explore this magnificent work with you! In the Präludium und Fuge in C-Dur, BWV 545, I recommend listening for the prelude's bright, architecturally clear architecture, showcasing Bach's mastery of counterpoint and harmony. Pay attention to the way the principal chorus registration adds depth and clarity to the intricate figuration. As we transition into the fugue, note how Bach subjects the theme to a range of procedures, including inversion, augmentation, and inversion with augmentation, creating a truly compelling contrapuntal argument. The angular chromaticism of the subject is striking, but it's Bach's formal strategy that makes this piece stand out. Finally, appreciate the way the fugue's structure unfolds, with its ritornello-like episodes and clever use of counterpoint to create a sense of tension and release. And don't forget to listen for the subtle nuances in registration, which add texture and variety to the overall performance.
3	What a wonderful piece! In the Präludium und Fuge in C-Dur, BWV 545, I recommend listening for the way Bach's masterful hands weave together intricate counterpoint and soaring melody. Pay attention to the way the fugue subject is expertly developed throughout the piece, with each entry adding new layers of texture and interest. And don't miss the majestic, sweeping arpeggios that punctuate the prelude – they're a perfect showcase for the organ's expressive capabilities!

Table 2: Two answers generated by Llama 3 to the question: What should I listen for in "Präludium und Fuge in C-Dur, BWV 545" by Johann Sebastian Bach? in 1) novice level and 2) expert level and 3) without RAG

of its generic and metaphoric character, reducing the risk of false information. The version without RAG is very generic without specific information on the piece. The expert-level answer contains some wrong information. For example, there is no "ritornello" in the fugue. Further, the phrase "Bach subjects the theme to inversion, augmentation, and inversion with augmentation" is incorrect. One of the main elements of the fugue is the syncopation between the voices.

A more in-depth evaluation of the accuracy is still outstanding. For this purpose, the knowledge base will be extended by more detailed information on the musical pieces. In its current knowledge base, there is only the information on the pieces in the JSON file; the Cambridge Companion to the Organ describes the musical epochs but contains almost no information on specific musical pieces. Therefore, the knowledge base might be insufficient to answer the audience's questions appropriately, resulting in these generic answers as the examples show.

Additional limitations of the current implementation will inform future work. Piece and composer information was manually prepared for this prototype. Integrating structured music metadata from open knowledge graphs, such as MusicBrainz or Wikidata, could improve scalability and enable the automatic enrichment of concerts with additional contextual facts. User studies with authentic classical music audiences would provide valuable insights into acceptance, trust, and perceived usefulness.

In summary, our findings show that RAG-grounded LLMs can offer stylistically adaptive and factually grounded concert moderation. However, it still has to be assessed whether listeners will accept such system and in which contexts it could be successfully implemented.

References

Elena V. Epure, Yashar Deldjoo, Bruno Sguerra, Markus Schedl, and Manuel Moussallam. 2025. Music recommendation with large language models: Challenges, opportunities, and evaluation. *Preprint*, arXiv:2511.16478.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen

Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2(1).

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, and et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

OpenAI. 2025. gpt-oss-120b and gpt-oss-20b model card. *Preprint*, arXiv:2508.10925.

Pushpdeep Singh, Mayur Patidar, and Lovekesh Vig. 2024. Translating across cultures: LLMs for intralingual cultural adaptation. In *Proceedings of the 28th Conference on Computational Natural Language Learning*, pages 400–418, Miami, FL, USA. Association for Computational Linguistics.

Nicholas Thistlethwaite and Geoffrey Webber, editors. 1999. *The Cambridge Companion to the Organ*, 1 edition. Cambridge University Press.

Sojeong Yun and Youn-kyung Lim. 2025. User experience with llm-powered conversational recommendation systems: A case of music recommendation. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI '25, New York, NY, USA. Association for Computing Machinery.

LabelBuddy: An Open Source Music and Audio Language Annotation Tagging Tool Using AI Assistance

Ioannis Prokopiou[1,2], Ioannis Sina[3], Agisilaos Kounelis[3], Pantelis Vikatos[2], Themos Stafylakis[1,4]

[1]Athens University of Economics and Business, [2]Orfium, [3]University of Patras, [4]Archimedes/Athena R.C.,

gian.prokopiou@aueb.gr, sinaioannis@gmail.com

agis@ceid.upatras.gr, pantelis@orfium.com, tstafylakis@aueb.gr

Abstract

The advancement of Machine learning (ML), Large Audio Language Models (LALMs), and autonomous AI agents in Music Information Retrieval (MIR) necessitates a shift from static tagging to rich, human-aligned representation learning. However, the scarcity of open-source infrastructure capable of capturing the subjective nuances of audio annotation remains a critical bottleneck. This paper introduces **LabelBuddy**, an open-source collaborative auto-tagging audio annotation tool designed to bridge the gap between human intent and machine understanding. Unlike static tools, it decouples the interface from inference via containerized backends, allowing users to plug in custom models for AI-assisted pre-annotation. We describe the system architecture, which supports multi-user consensus, containerized model isolation, and a roadmap for extending agents and LALMs. Code available at https://github.com/GiannisProkopiou/gsoc2022-Label-buddy.

1 Introduction

The quality of the datasets used to train AI models constitutes a significant factor in accuracy, reliability, and generalization (Picard et al., 2020). Despite standardization efforts through public repositories like Zenodo[1] and data lakes (Espinal et al., 2022), or benchmarks like DCASE[2] and MIREX[3], available resources are often task-specific. In addition, the creation of specialized datasets can be a challenging procedure, complicated, time-consuming, and is often laborious and supported by human review (Voigtlaender et al., 2021). Building on the tradition of cross-disciplinary impact, the intersection of Natural Language Processing (NLP) with music and audio presents a frontier where language and sound converge. Most audio content contains an inherent linguistic dimension, yet creating datasets to capture these multimodal synergies remains a laborious bottleneck.

The domain of Music Information Retrieval (MIR) is currently undergoing a transition from discriminative paradigms characterized by static tag classification to generative and reasoning-based approaches. The rise of Large Audio-Language Models (LALMs) such as Music Flamingo (Ghosh et al., 2025), Qwen-Audio (Chu et al., 2023), and Audio Flamingo 3 (Goel et al., 2025) has introduced new capabilities for "chain-of-thought" reasoning and conversational audio understanding. However, the efficacy of these models is heavily dependent on the quality of alignment with human intent. Recent surveys indicate that objective metrics often fail to capture aesthetic nuance, necessitating a pivot toward Reinforcement Learning from Human Feedback (RLHF) and rigorous subjective evaluation methodologies (Kader and Karmaker, 2025).

Current workflows are often fragmented, and users resort to disjointed workflows, separating data curation from the critical phase of manual subjective evaluation (e.g., MUSHRA, GoListen, or pairwise preference testing) (Schoeffler et al., 2018; Barry et al., 2021). Users use waveform-based tools for segmentation (Grover et al., 2020), separate platforms for text handling, and distinct software for subjective evaluation (e.g., WebMUSHRA (Schoeffler et al., 2018)). This separation hinders the development of efficient *Human-in-the-Loop* (HITL) pipelines, where the uncertainty of the model's output should drive data acquisition. Furthermore, the "crisis of metrics" in generative music, where objective scores like FAD fail to correlate with human perception (Kader and Karmaker, 2025) demands tools that can seamlessly transition from annotation to subjective preference ranking.

To address this, we present **LabelBuddy**, an open-source collaborative auto-tagging audio annotation tool equipped with AI assistance with:

[1]https://zenodo.org/
[2]https://dcase.community/
[3]https://www.music-ir.org/mirex

Proceedings of the 4th Workshop on NLP for Music and Audio (NLP4MusA 2026), pages 7–12

1. **Decoupled AI-Assistance:** An isolated containerized architecture injects model predictions via declarative YAML files. We provide pre-trained models like YOHO (Venkatesh et al., 2022), musicnn (Pons and Serra, 2019), PANNs (Kong et al., 2020), and LALMs like Music Flamingo (Ghosh et al., 2025) for AI-assisted pre-annotation tags to shift user effort from creation to verification, while approved labels can be used to fine-tune the models.

2. **Collaborative Consensus:** Native support for multi-user roles (manager, annotator, reviewer) to ensure ground-truth reliability.

3. **Hybrid Workflow Support:** An architecture designed to support both region-based tagging and subjective preference aggregation.

2 Related Work

This section reviews annotation platforms, HITL workflows for LALMs, and infrastructure for subjective evaluation and RLHF.

Annotation Platforms & Domain Specificity. The landscape of data curation significantly varies by modality. For text, tools like BRAT (Stenetorp et al., 2012), Paladin (Nghiem et al., 2021), and PubAnnotation (Kim and Wang, 2012) facilitate linguistic tagging, while specialized frameworks like CAT (Bartalesi Lenzi et al., 2012) and MDSWriter (Meyer et al., 2016) handle semantic efficiency and summarization respectively. Active learning strategies have been explored in text labeling (e.g., ActiveAnno (Wiechmann et al., 2021), APLenty (Nghiem and Ananiadou, 2018)). In the visual domain, tools like VIA (Dutta and Zisserman, 2019) and Annotation Web (Smistad et al., 2021) show the need for domain-specific interfaces.

In the audio domain, tools like Audino (Grover et al., 2020) and BAT (Meléndez Catalán et al., 2017) excel at temporal tasks like Sound Event Detection (SED) and salience, while library-based solutions like Aubio (Brossier et al., 2019) facilitate feature extraction. Others, like Gecko (Levy et al., 2019), focus on voice segmentation. However, these tools generally lack the decoupled AI architecture required for modern reasoning model use. Conversely, general-purpose HITL platforms like Label Studio (Tkachenko et al., 2020-2022) and Prodigy (Montani and Honnibal, 2018) offer robust backends but often restrict collaborative features such as reviewer roles and consensus metrics to paid enterprise tiers. Furthermore, they lack native support for musical structures (e.g., bars, beats) found in specialized audio tools (Cartwright et al., 2017). A comparison between LabelBuddy and other existing tools is shown in Table 1.

Table 1: Comparison of LabelBuddy with existing tools.

Tool	Audio Specific	Decoupled AI-Assist	Open Source	Collaboratory Consensus
Audino	✓	-	✓	-
BAT	✓	-	✓	-
Aubio	✓	-	✓	-
Gecko	✓	-	✓	-
Prodigy	-	✓	-	-
Label Studio (CE)	-	✓	✓	-
LabelBuddy	✓	✓	✓	✓

LALMs & HITL Workflows. The state-of-the-art has shifted from fixed-vocabulary auto-taggers to Large Audio-Language Models (LALMs) such as Audio Flamingo 3 (Goel et al., 2025) and Qwen-Audio (Chu et al., 2023). These utilize unified encoders for "chain-of-thought" reasoning. To align them, we adopt a "Single-Iteration" HITL approach. Recent studies in video annotation (Gutiérrez et al., 2025) demonstrate that simple model-assisted pre-annotation reduces time-on-task without degrading quality, a philosophy we extend to audio similarly to NEAL (Gibbons et al., 2023).

Subjective Evaluation & RLHF. A critical bottleneck in generative music is the "crisis of metrics," where scores like Fréchet Audio Distance (FAD) fail to correlate with human perception (Kader and Karmaker, 2025; Gui et al., 2023). Consequently, the field is pivoting towards RLHF (Cideron et al., 2024; Liu et al., 2025). Currently, evaluation is decoupled from annotation, relying on standalone tools like WebMUSHRA (Schoeffler et al., 2018). LabelBuddy aims to unify this by integrating pairwise preference aggregation methods, such as Bayesian Bradley–Terry (BBQ) (Aczel et al., 2025).

3 System Architecture

LabelBuddy addresses the "coupling problem" in annotation tools where interfaces are hard-coded to specific model backends via a modular, containerized architecture. As illustrated in Figure 1, the system decouples the lightweight user interaction layer (Django) from the compute-intensive inference layer (Docker).

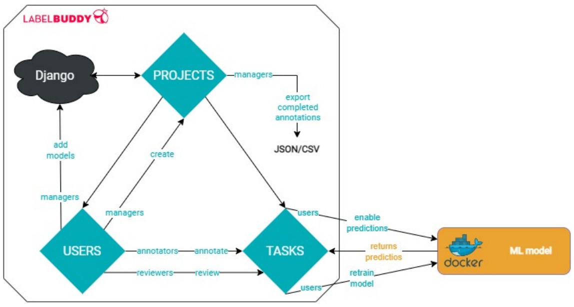

Figure 1: **System Architecture Overview**: The architecture decouples the Django web server from Dockerized ML inference.

3.1 Backend & Data Model

The core application is built on Django, utilizing a relational database to manage the three primary entities: *Projects*, *Users*, and *Tasks*.

- **RBAC & Privacy:** To prevent data leakage, the system implements Role-Based Access Control (RBAC). Managers have full oversight, while *Annotators* and *Reviewers* are restricted to their assigned task queues.
- **Data Serialization:** Annotations are stored as JSON objects that contain temporal boundaries and ontology tags. Managers can export consensus data to CSV/JSON formats for direct integration with ML training pipelines.

3.2 Containerized Inference Engine

Managers define models via a YAML configuration file, specifying the Docker image, input/output schema, and resource constraints. When AI assistance is requested, the backend communicates with the model container via a RESTful Flask API. This design ensures Sandboxing (models run in isolated environments) and Scalability (inference can be deployed on remote cloud nodes like AWS/Azure).

4 Workflow & Interface

The platform supports a comprehensive "Human-in-the-Loop" (HITL) lifecycle.

4.1 Project Setup & Task Management

The workflow begins in the **Dashboard**, where managers create projects and assign user roles.

- **Model Integration:** In the **Model Page**, managers upload YAML configuration files to attach inference containers. This interface exposes advanced controls: monitoring training loss/accuracy, downloading weight files, and triggering fine-tuning jobs (specifying epochs and learning rates) using the project's validated data.
- **Task Ingestion:** Managers upload audio (WAV/MP3) via the **Project Page**, which can be distributed to annotators via a shared pool or disjoint assignment strategies.

4.2 The Annotation Loop

The core labeling workflow utilizes `wavesurfer.js` for responsive waveform visualization.

1. **AI-Assisted Pre-Annotation:** Annotators trigger "On-Demand Prediction," which serializes the audio to the active Docker container. The system renders the returned predictions as editable regions (Fig. 2), shifting the human task from *creation* to *verification*.
2. **Review & Consensus:** Completed tasks enter the **Review Interface**, where reviewers can play back specific regions and approve or reject annotations with feedback. This Quality Assurance (QA) loop is essential for creating high-fidelity datasets for generative alignment.

5 Case Study: NLP Music Tagging

To demonstrate LabelBuddy's utility, we present a reference workflow for creating a Music Captioning Dataset, a task requiring the alignment of audio signals with rich natural language descriptions.

Model Integration The project manager defines a Docker container that wraps a multimodal model, such as a Music Flamingo checkpoint. The YAML configuration maps the model's text output to a LabelBuddy `Annotation` region:

```
image: "my-repo/music-flamingo:v1"
input_schema: { "audio": "wav" }
output_schema:
  - { "type": "text", "label": "Caption" }
resources: { "gpu": "true" }
```

The "Human-Verify" Loop. Annotators are presented with a queue of raw audio tracks. Instead of writing descriptions from scratch (which is cognitively demanding), they trigger the **Pre-Annotate** function. The backend container processes the audio and returns a candidate caption: *"A lo-fi hip-hop track with a slow tempo and vinyl crackle."*

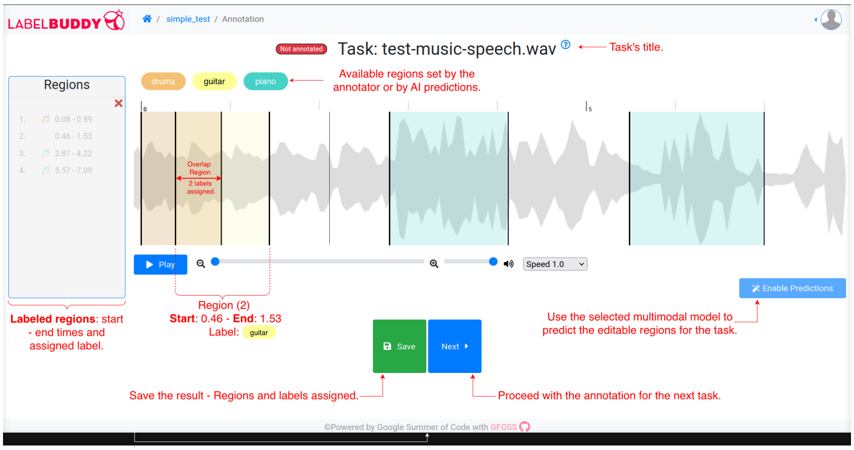

Figure 2: The **annotation interface** displaying AI-generated predictions as editable waveform regions.

Correction & Consensus. The annotator corrects specific hallucinations (e.g., changing "vinyl crackle" to "rain sounds") and adjusts timestamp boundaries. If multiple annotators process the same track, the Reviewer Interface highlights semantic disagreements in the captions, allowing for the creation of robust, consensus-based ground truth.

Multimodal Export. The finalized dataset is exported as a JSONL or CSV file containing aligned `(audio_path, text_caption)` pairs, ready for immediate use in fine-tuning downstream audio-to-text generation models.

6 Discussion & Future Roadmap

The current release of LabelBuddy solves the immediate infrastructure challenge: decoupling the annotation frontend from rapidly evolving model backends. As the field transitions towards LALMs and autonomous agents, our future roadmap is aligned with bridging the gap between human intent and machine representation:

Agentic Reasoning. While traditional active learning relies on uncertainty sampling, the rise of LALMs requires a shift toward *conversational* assistance. We are extending the backend API to support more reasoning-capable models such as Qwen-Audio (Chu et al., 2023). Future versions will allow annotators to query the model and receive "Chain-of-Thought" justifications, transforming the workflow from simple tag verification to collaborative reasoning. This aligns with recent findings that interactive reasoning reduces hallucination in complex annotation tasks.

Integrated Subjective Evaluation (RLHF). Acknowledging the "crisis of metrics" where FAD scores fail to capture aesthetic quality (Kader and Karmaker, 2025), LabelBuddy aims to evolve into a workbench for RLHF. We aim to implement a native "Pairwise Preference" interface (Dataset A vs. Dataset B) directly in the review loop. To handle noisy human raters, the backend will integrate Bayesian Bradley–Terry (BBQ) models (Aczel et al., 2025), providing robust preference aggregation to align generative models (Cideron et al., 2024).

Enhancing Perceptual Validity. To counteract the tendency of models to rely on text priors rather than audio content a flaw highlighted by the RUListening benchmark (Zang et al., 2025), we plan to introduce timestamp-required QA templates. These will force both models and human annotators to ground every semantic claim in specific spectral regions, ensuring that future datasets drive genuine auditory perception rather than text-only reasoning.

Evaluation Plan. To validate utility, we propose a pilot study on DCASE 2024 data measuring: (a) time reduction vs. *de novo* labeling, (b) inter-annotator agreement (Fleiss' Kappa), and (c) downstream PSDS gains for baseline SED models trained on LabelBuddy-curated data.

7 Conclusion

LabelBuddy serves as critical infrastructure for exploring the multimodal synergies between language and audio. By decoupling the interface from inference, it empowers the community to curate the rich, linguistically-grounded datasets required for modern NLP-driven music understanding. Whether for standard tagging or the emerging demands of RLHF, LabelBuddy offers an open, scalable workbench to deepen the connection between human perception and machine representation on audio.

8 Ethics Statement

The development of AI-assisted annotation tools raises concerns regarding labor displacement and bias. LabelBuddy is designed to augment, not replace, human expertise, keeping the human in the loop for critical judgments. Furthermore, by facilitating the creation of open datasets, we aim to democratize access to high-quality training data, countering the centralization of resources in large tech corporations. We ensure that all integrated models are used in compliance with their research licenses.

Acknowledgments

This work was supported by Google Summer of Code (GSoC) and the Open Technologies Alliance (GFOSS) and was funded by the European Union's Horizon Europe research and innovation programme under the AIXPERT project (Grant Agreement No. 101214389), which aims to develop an agentic, multi-layered, GenAI-powered framework for creating explainable, accountable and transparent AI systems.

References

Till Aczel, Lucas Theis, and Wattenhofer Roger. 2025. Efficient bayesian inference from noisy pairwise comparisons. *arXiv preprint arXiv:2510.09333*.

Dan Barry, Qijian Zhang, Pheobe Wenyi Sun, and Andrew Hines. 2021. Go listen: an end-to-end online listening test platform. *Journal of Open Research Software*, 9(1).

Valentina Bartalesi Lenzi, Giovanni Moretti, and Rachele Sprugnoli. 2012. Cat: the celct annotation tool. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 333–338. LREC.

Paul Brossier, Tintamar, Eduard Müller, Nils Philippsen, Tres Seaver, Hannes Fritz, cyclopsian, Sam Alexander, Jon Williams, James Cowgill, and Ancor Cruz. 2019. aubio/aubio: 0.4.9.

Mark Cartwright, Ayanna Seals, Justin Salamon, Alex Williams, Stefanie Mikloska, Duncan MacConnell, Edith Law, Juan P Bello, and Oded Nov. 2017. Seeing sound: Investigating the effects of visualizations and complexity on crowdsourced audio annotations. *Proceedings of the ACM on Human-Computer Interaction*, 1(CSCW):1–21.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. 2023. Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models. *arXiv preprint arXiv:2311.07919*.

Geoffrey Cideron, Sertan Girgin, Mauro Verzetti, Damien Vincent, Matej Kastelic, Zalán Borsos, Brian McWilliams, Victor Ungureanu, Olivier Bachem, Olivier Pietquin, et al. 2024. Musicrl: Aligning music generation to human preferences. *arXiv preprint arXiv:2402.04229*.

Abhishek Dutta and Andrew Zisserman. 2019. The via annotation software for images, audio and video. In *Proceedings of the 27th ACM international conference on multimedia*, pages 2276–2279.

Xavier Espinal, Maria Giuffrida, Marieke Willems, and Rita Meneses. 2022. Bringing big science experiment data to the researchers' fingertips.

Sreyan Ghosh, Arushi Goel, Lasha Koroshinadze, Sang-gil Lee, Zhifeng Kong, Joao Felipe Santos, Ramani Duraiswami, Dinesh Manocha, Wei Ping, Mohammad Shoeybi, et al. 2025. Music flamingo: Scaling music understanding in audio language models. *arXiv preprint arXiv:2511.10289*.

Anthony Gibbons, Ian Donohue, Courtney Gorman, Emma King, and Andrew Parnell. 2023. Neal: an open-source tool for audio annotation. *PeerJ*, 11:e15913.

Arushi Goel, Sreyan Ghosh, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang-gil Lee, Chao-Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael Valle, et al. 2025. Audio flamingo 3: Advancing audio intelligence with fully open large audio language models. *arXiv preprint arXiv:2507.08128*.

Manraj Singh Grover, Pakhi Bamdev, Yaman Kumar, Mika Hama, and Rajiv Ratn Shah. 2020. audino: A modern annotation tool for audio and speech. *arXiv preprint arXiv:2006.05236*.

Azalea Gui, Hannes Gamper, Sebastian Braun, and Dimitra Emmanouilidou. 2023. Adapting frechet audio distance for generative music evaluation. *arXiv preprint arXiv:2311.01616*.

Juan Gutiérrez, Ángel Mora, Pablo Regodón, Silvia Rodriguez, and José Luis Blanco. 2025. Ai-boosted video annotation: Assessing the process enhancement. *arXiv preprint arXiv:2510.21798*.

Faria Binte Kader and Santu Karmaker. 2025. A survey on evaluation metrics for music generation. *arXiv preprint arXiv:2509.00051*.

Jin-Dong Kim and Yue Wang. 2012. Pubannotation-a persistent and sharable corpus and annotation repository. In *BioNLP: Proceedings of the 2012 Workshop on Biomedical Natural Language Processing*, pages 202–205.

Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley. 2020. Panns:

Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894.

Golan Levy, Raquel Sitman, Ido Amir, Eduard Golshtein, Ran Mochary, Eilon Reshef, Roi Reichart, and Omri Allouche. 2019. Gecko-a tool for effective annotation of human conversations. In *INTERSPEECH*, pages 3677–3678.

Cheng Liu, Hui Wang, Jinghua Zhao, Shiwan Zhao, Hui Bu, Xin Xu, Jiaming Zhou, Haoqin Sun, and Yong Qin. 2025. Musiceval: A generative music dataset with expert ratings for automatic text-to-music evaluation. *arXiv preprint arXiv:2501.10811*.

Blai Meléndez Catalán, Emilio Molina, and Emilia Gómez Gutiérrez. 2017. Bat: An open-source, web-based audio events annotation tool.

Christian M Meyer, Darina Benikova, Margot Mieskes, and Iryna Gurevych. 2016. Mdswriter: annotation tool for creating high-quality multi-document summarization corpora. In *Proceedings of ACL-2016 System Demonstrations*, pages 97–102.

Ines Montani and Matthew Honnibal. 2018. Prodigy: A new annotation tool for radically efficient machine teaching. *Artificial Intelligence to appear*.

Minh-Quoc Nghiem and Sophia Ananiadou. 2018. Aplenty: annotation tool for creating high-quality datasets using active and proactive learning. In *EMNLP (Demonstration)*, pages 108–113.

Minh-Quoc Nghiem, Paul Baylis, and Sophia Ananiadou. 2021. Paladin: an annotation tool based on active and proactive learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 238–243.

Sylvaine Picard, Camille Chapdelaine, Cyril Cappi, Laurent Gardes, Eric Jenn, Baptiste Lefevre, and Thomas Soumarmon. 2020. Ensuring dataset quality for machine learning certification. In *2020 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, pages 275–282. IEEE.

Jordi Pons and Xavier Serra. 2019. musicnn: Pretrained convolutional neural networks for music audio tagging. *arXiv preprint arXiv:1909.06654*.

Michael Schoeffler, Sarah Bartoschek, Fabian-Robert Stöter, Marlene Roess, Susanne Westphal, Bernd Edler, and Jürgen Herre. 2018. webmushra—a comprehensive framework for web-based listening tests. *Journal of Open Research Software*, 6(1).

Erik Smistad, Andreas Østvik, and Lasse Løvstakken. 2021. Annotation web-an open-source web-based annotation tool for ultrasound images. In *2021 IEEE International Ultrasonics Symposium (IUS)*, pages 1–4. IEEE.

Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012. Brat: a web-based tool for nlp-assisted text annotation. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107.

Maxim Tkachenko, Mikhail Malyuk, Andrey Holmanyuk, and Nikolai Liubimov. 2020-2022. Label Studio: Data labeling software. Open source software available from https://github.com/heartexlabs/label-studio.

Satvik Venkatesh, David Moffat, and Eduardo Reck Miranda. 2022. You only hear once: a yolo-like algorithm for audio segmentation and sound event detection. *Applied Sciences*, 12(7):3293.

Paul Voigtlaender, Lishu Luo, Chun Yuan, Yong Jiang, and Bastian Leibe. 2021. Reducing the annotation effort for video object segmentation datasets. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3060–3069.

Max Wiechmann, Seid Muhie Yimam, and Chris Biemann. 2021. Activeanno: General-purpose document-level annotation tool with active learning integration. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations*, pages 99–105.

Yongyi Zang, Sean O'Brien, Taylor Berg-Kirkpatrick, Julian McAuley, and Zachary Novack. 2025. Are you really listening? boosting perceptual awareness in music-qa benchmarks. *arXiv preprint arXiv:2504.00369*.

Stochastic Parrots or True Virtuosos?

Digging Deeper Into the Audio-Video Understanding of AVQA Models

Sara Pernille Jensen
Department of Philosophy, Classics, History of Art and Ideas
University of Oslo
sarapje@ifikk.uio.no

Hallvard Innset Hurum
Department of Informatics
University of Oslo
hallvaih@ifi.uio.no

Anna-Maria Christodoulou
RITMO,
Department of Musicology
University of Oslo
annammc@uio.no

Abstract

Audio-video question answering (AVQA) systems for music show signs of multimodal "understanding", but it is unclear which inputs they rely on or whether their behavior reflects genuine audio-video reasoning. Existing evaluations focus on overall accuracy and rarely examine modality dependence. We address this gap by suggesting a method of using counterfactual evaluations to analyse the audio-video understanding of the models, illustrated with a case study on the audio-video spatial-temporal (AVST) architecture. This includes interventions that zero out or swap audio, video, or both, where results are benchmarked against a baseline based on linguistic patterns alone. Results show stronger reliance on audio than video, yet performance persists when either modality is removed, indicating learned cross-modal representations. The AVQA system studied thus exhibits non-trivial multimodal integration, though its "understanding" remains uneven.

1 Introduction

An increasingly popular use of machine-learning (ML) models is computational music analysis through audio, language, and video (Manco et al., 2022; Simonetta et al., 2019; Li et al., 2022; Christodoulou et al., 2025). Such analyses are believed to require a degree of *understanding* of music, motivating research into how ML models reason about it. Audio-video question–answering (AVQA) (Li et al., 2022) is a computational task that requires a model to answer questions based on both audio and video.

Various architectures have been proposed to improve music-related AVQA performance (Lin et al., 2023; Li et al., 2022; Christodoulou et al., 2025), but benchmark scores alone reveal little about how models reason from audio-video inputs. Understanding these dependencies can guide improvements in both models and data.

Counterfactual interventions offer a way to probe multimodal reasoning by systematically manipulating inputs and observing the ensuing changes in predictions, revealing dependencies, biases, and failure modes. As a case study, we apply such analyses to an AVST model (Li et al., 2022) trained on the MusiQAl dataset (Christodoulou et al., 2025). We introduce interventions that zero out modalities or create conflicting cross-modal pairs, and examine changes in predictions and confidence. We also analyse dataset patterns to distinguish multimodal reasoning from biases. Results indicate meaningful audio-video processing and possibly understanding, but uneven reliance on modalities, highlighting directions for future music-oriented QA models.

2 Related work

The AVST model (Li et al., 2022) was one of the first to explore spatial-temporal reasoning in musical performances. Tested on the MUSIC-AVQA dataset with over 45,867 QA pairs across five categories (existential, counting, location, comparative, and temporal) the model includes spatial and temporal grounding modules, enabling localization of instrument sounds and reasoning about their timing. Multimodal inputs improved performance over audio- or video-only approaches. The question templates in the MUSIC-AVQA dataset include placeholders, e.g., "Which instrument makes sounds $\langle$BA$\rangle$ the $\langle$Object$\rangle$?", with $\langle$BA$\rangle$ as "before/after" and $\langle$Object$\rangle$ as a musical object. Questions are also labeled as audio, video, or audio-video, indicating which modality is presumably needed to answer the question, though the model does not receive this information during training.

The MusiQAl dataset (Christodoulou et al., 2025) contains 310 videos and 11,793 QA pairs spanning diverse cultures and genres. Based on the MUSIC-AVQA's question templates, it extends this by introducing 47 new ones, as well as including

Proceedings of the 4th Workshop on NLP for Music and Audio (NLP4MusA 2026), pages 13–19
March 28, 2026 ©2026 Association for Computational Linguistics

two new question categories, 'Causal' and 'Purpose', to probe deeper understanding of music performance. Christodoulou et al. (2025) trained both AVST (Li et al., 2022) and LAVISH (Lin et al., 2023) models on MusiQAl. A schematic of the pipeline for how the data is split up, fed into the model, and used for prediction is shown in Fig. 1. The QA task is treated as classification over the answer vocabulary, with the model output interpreted as probabilities via softmax; the highest-probability answer is taken as the prediction. Results revealed strengths in areas like performer tracking, but highlight that further development is needed in complex scenarios like style classification or source separation in ensemble performances. The overall accuracy was ≈ 71%.

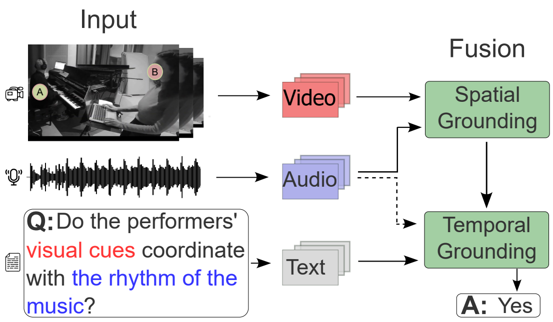

Figure 1: Illustration of the basic components of the ML model used in (Christodoulou et al., 2025) during inference, showing how the audio, video, and question are inputted to the model, and used for making the final prediction of the answer. Figure borrowed from (Christodoulou et al., 2025) with permission.

Recent work has also examined what multimodal models actually learn (Weck et al., 2024; Zang et al., 2025). Both studies analysed music QA models, investigating to what extent performance relied on processing the auditory or linguistic input data, and hence the type of understanding it is indicative of. Strong linguistic biases were found, as models often reached near-original accuracy using only the questions. This highlights the need for careful evaluation of which modalities models truly exploit.

3 Methods

To analyse how the multimodal AVST model reasons based on the different modalities, we applied a set of counterfactual interventions to the inputs and observed the resulting changes in prediction accuracy. We also examined the distribution of the model's confidence in its predicted answers to assess how its certainty shifts in response to the interventions. This constitutes a black-box interpretability approach, as we did not inspect changes in the model's internals.

The goal was to investigate the model's relative reliance on the different modalities, on which we systematically intervened, using two methods: modality ablations and modality conflicts. In addition, we derived estimates of the chance of correct predictions based on random guessing and statistical patterns in the linguistic question–answer pairs alone, and used these as a baseline for evaluating the model's relative success. All models were trained by us according to the specifications given in (Christodoulou et al., 2025). All figures are included in the accompanying appendix.

3.1 Modality Ablation

Modality ablation involved removing one or both modalities by setting the audio and/or video input tensors to zero, forcing the model to rely on the question alone or on the question and a single modality. Changes in prediction accuracy under these conditions indicate which modality the model relies on most heavily, both overall and across question categories.

To further understand differences in accuracy, test samples were also split based on whether the model answered them correctly when given the complete input. This highlights not only overall accuracy changes, but also how the modalities given affect the predictions in both directions, for better and worse. Model confidence under each ablation was also examined.

A potential limitation is that the model was trained on non-zero inputs, so zeroed tensors may not be interpreted as silence or blank images, as intended. To mitigate this, three additional models were trained with the same dataset, architecture, and hyperparameters, but with audio, video, or both modalities ablated, providing a reference to validate the ablation results for the main model.

3.2 Modality Conflict

The modality conflict intervention mixes audio and video across different question–answer pairs, providing the model with signals that do not fit together. We observe whether the model can still answer questions that depend on a single modality, and how it handles conflicting inputs when given questions that depend on both modalities.

We tested three types of conflict interventions: one in which the audio was replaced with an incorrect sample (relative to the question), while the video was kept unchanged, one in which the video was replaced with an incorrect sample, while the audio remained the same, and one in which both audio and video were replaced with incorrect samples. Replacements were drawn randomly from the 310 unique performances in the dataset. Since performances are unevenly distributed across QA pairs, each of the 310 modalities was used to replace 3-4 samples ($\frac{1183}{310} \approx 3.8$) to reduce representation bias. For each conflict type, five independent runs were performed with different reproducible seeds, ensuring unique replacements across runs. In the "both" mode, audio and video were replaced with samples from separate performances.

3.3 Dataset Analysis

To investigate what the model had learned, we analysed the chances of success assuming no audio-video understanding. Positive evidence alone is insufficient to infer a hypothesis; plausible alternative hypotheses must also be considered and falsified (Reiss, 2015). The main hypothesis was that the model had learned to understand audio-video content across sub-categories, rather than relying on simpler strategies such as random guessing or exploiting statistical patterns in the question–answer pairs. For example, always predicting the most frequent answer for a given question template provides a baseline that requires no use of audio or video inputs, and consequently does not reflect genuine audio-video understanding.

These alternatives were formalised by computing expected accuracy from random guessing and from always predicting the most frequent answer per question template. These baselines allow for a deeper analysis of the model performance across question categories. Performance above them indicates reliance on the audio-video inputs, while performance at or below could be explained by the model's relying on linguistic patterns alone. Expected accuracy was calculated per question template, accounting for the actual number of possible answers, which varies (e.g., "yes/no?", versus "which instrument?"), and reflects the subset of answers the model could plausibly infer.

4 Results

4.1 Dataset Analysis

The expected accuracy on test data from uniform guessing is 32.14%, whereas always predicting the most frequent answer from the training data, based on the question template, gives an accuracy of 47.84%. The reason for the high uniform accuracy is that many of the questions only have a few possibly answers, with 70% of the samples having five or fewer alternatives.

Appendix Fig. 2 compares guessing strategies across question categories. Frequentist guessing generally outperforms uniform guessing, reflecting learnable linguistic patterns, such as common instrument usage. The AVST model surpasses both baselines in almost all categories. However, sample sizes differ widely, with some categories only including a few questions, which should be considered when interpreting evidence for understanding.

4.2 Modality Ablation

Overall test accuracies for the different models and modality combinations are given in Table 1. First, the standard model trained on complete data was tested with video, audio, and both modalities removed. Appendix Fig. 3 shows that removing video slightly reduces accuracy, audio more so, and both modalities cause a substantial drop. There is no clear alignment between the question label (audio, video, or both) and sensitivity to ablations, but overall, the model depends more on audio than video.

To further probe this, the subset of test questions answered incorrectly with full input was analysed under the same ablations. Removing both modalities gives the highest accuracy in this subset, audio second, and video lowest, showing which ablations lead to the greatest changes in predictions. The inverse pattern is found when looking at the questions that were answered correctly when given complete input. Both findings are consistent with the previous results in terms of relative modality significance. The resulting overall accuracy for the different interventions, on both the initially incorrectly answered questions and the complementary subset (correctly answered questions), is given in Table 1.

Separate models trained on ablated data (audio, video, or both removed) show only marginally better performance than applying ablations to the standard model, confirming the validity of the method. Hyperparameters were not tuned for these alternative models, so results are indicative rather than precise measures of information extracted from each modality.

Appendix Fig. 4 shows the distribution of pre-

Table 1: Overall test accuracy for the standard model (except last column: models trained on corresponding ablated data) under specified modality ablations. 'Success'/'Failure' indicates subsets of test data where the standard model succeeds/fails with full input.

Modality Excluded	Standard	Success	Failure	Alt. mod.
None	70.58%	100%	0%	—
Video	66.27%	87.78%	14.66%	67.79%
Audio	56.13%	70.78%	20.98%	58.50%
Both	46.49%	54.13%	28.16%	50.80%

dicted probabilities for correct and incorrect answers under each ablation. Correct predictions are most confident with full data (peak approximately 90%, mean 76%), slightly lower with audio only, lower with video only, and lowest with questions alone. Incorrect predictions show similar distributions across ablations. Differences are likely underestimated, since questions with more possible answers naturally have lower maximum predicted probabilities; models with more modalities can answer these complex questions, lowering the average predicted probability.

4.3 Modality conflict

We examine how the model responds to conflicting audio and video signals.

Table 2 shows overall accuracy under different conflict conditions compared to the baseline without conflict (approximately 71%). Video conflict reduced accuracy to 61%, audio conflict to 51%, and conflicts in both modalities to 44%, again indicating greater reliance on audio than video.

Table 2: Overall test accuracy for the standard model under modality conflicts. 'Success'/'Failure' indicates subsets of test data where the standard model succeeds/fails with full input.

Modality Conflicted	Standard	Success	Failure
None	70.58%	100%	0%
Video	61.05%	79.16%	16.72%
Audio	50.97%	65.25%	20.34%
Both	43.79%	53.84%	24.31%

The 'success' and 'failure' subsets show complementary trends. In the failure subset, video conflict allowed the model to correct 17% of previously incorrect answers, audio conflict 20%, and both conflicts 25%. This confirms that audio conflicts impact predictions more than video conflicts, consistent with overall findings.

Detailed subcategory analysis (Appendix Fig. 5) reveals that audio conflicts consistently reduce performance in audio and audio-video questions. Video-existential questions remain largely unaffected by video conflict, likely due to language bias (e.g., the correct answer to "Is the dancer in the video always dancing?" is "yes" 80% of the time). Audio-video comparative questions maintain high accuracy under all conflicts, possibly due to language bias in yes/no questions (e.g., "Do the performers' video cues coordinate with the rhythm of the music?"; 94% "yes"). Audio-video temporal questions drop sharply under audio conflict, indicating that they are primarily audio-driven. Some audio-video questions may be mislabelled, relying almost entirely on one modality.

To probe model certainty, we analysed the highest softmax probability per prediction (Appendix Fig. 6). Confidence was higher for correct answers (74–76%) than incorrect ones (57–61%). Correct predictions show small differences across conflict types, highest with no conflict (76%). Incorrect predictions show increased confidence under dual-modality conflict (61%), suggesting the model becomes overconfident when inputs conflict. Variations in accuracy do not always reflect variations in probability distributions due to differing numbers of correct/incorrect samples per conflict type.

5 Discussion and Conclusion

Across all experiments, the AVST model's performance cannot be explained solely by random guessing or question–answer statistics, but it is clearly shaped by dataset bias and asymmetric modality use. Relatively high baseline accuracies (32% for uniform guessing and 48% for frequentist guessing) reveal strong linguistic and structural biases in the dataset. Nevertheless, the model's overall test accuracy of 71% indicates that it leverages information from audio and video inputs beyond question priors.

Modality ablation and conflict experiments reveal a clear hierarchy in modality importance: audio is the dominant modality, with audio removal or conflict leading to substantially larger performance drops than corresponding video interventions. When both modalities are removed or conflicted, accuracy falls below the frequentist baseline but remains above uniform guessing, suggesting that the model integrates multimodal inputs while still relying on residual language cues. The weak alignment between nominal question labels and ablation sensitivity further highlights limitations in the dataset's labelling scheme, particularly for questions that are answerable from either modality (Audio *or* Video, instead of Audio-Video). This distinction was not included in the labelling scheme, which it should be in future work.

Subcategory analyses show that several nominally multimodal questions are effectively unimodal or driven by answer imbalances, while temporal questions rely primarily on audio. Increased confidence under conflicting inputs suggests limited reasoning about cross-modal inconsistency and a tendency towards overconfident decisions when evidence disagrees.

Overall, the model processes audio and video in a fused representation but relies predominantly on auditory cues and linguistic biases. Performance improvements over baseline, therefore, reflect a combination of genuine multimodal processing and shortcut learning, motivating the need for improved question design and evaluation methods that more directly probe cross-modal reasoning.

As this investigation was limited to a single AVST model, it is left for future work to determine whether these findings generalise to other models trained for similar tasks. Yet, we believe the results are sufficient to motivate the need for probing deeper into the somewhat superficial accuracy scores that are usually reported, providing a richer understanding of how such models work.

Ethics and Consent

This study builds on (Christodoulou et al., 2025), using their publicly available AVST model and dataset. Models were independently trained to verify reproducibility. The dataset contains diverse musical cultures and question-answer annotations; we relied on the original annotations.

Acknowledgements

This work was partially supported by the Research Council of Norway through its Centres of Excellence scheme, project number 262762.

6 Open Access Statement

Our GitHub repository with all code and instructions to reproduce our results can be found at `https://github.com/silyeah/MusiQAl/tree/in5490`.

References

Anna-Maria Christodoulou, Kyrre Glette, Olivier Lartillot, and Alexander Refsum Jensenius. 2025. Musiqal: A dataset for music question–answering through audio–video fusion. *Transactions of the International Society for Music Information Retrieval*.

Guangyao Li, Yake Wei, Yapeng Tian, Chenliang Xu, Ji-Rong Wen, and Di Hu. 2022. Learning to answer questions in dynamic audio-visual scenarios. *Preprint*, arXiv:2203.14072.

Yan-Bo Lin, Yi-Lin Sung, Jie Lei, Mohit Bansal, and Gedas Bertasius. 2023. Vision transformers are parameter-efficient audio-visual learners. *Preprint*, arXiv:2212.07983.

Ilaria Manco, Emmanouil Benetos, Elio Quinton, and György Fazekas. 2022. Contrastive audio-language learning for music. In *Proceedings of the 23rd International Society for Music Information Retrieval Conference (ISMIR)*.

Julian Reiss. 2015. A pragmatist theory of evidence. *Philosophy of Science*, 82(3):341–362.

Federico Simonetta, Stavros Ntalampiras, and Federico Avanzini. 2019. Multimodal music information processing and retrieval: Survey and future challenges. In *Proceedings - 2019 International Workshop on Multilayer Music Representation and Processing, MMRP 2019*.

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

Yongyi Zang, Sean O'Brien, Taylor Berg-Kirkpatrick, Julian McAuley, and Zachary Novack. 2025. Are you really listening? boosting perceptual awareness in music-qa benchmarks. *Preprint*, arXiv:2504.00369.

Appendix

Due to page limitations, all supplementary visualizations of results are included in the appendix. These figures provide additional context and insights relevant to the analyses presented in the main manuscript.

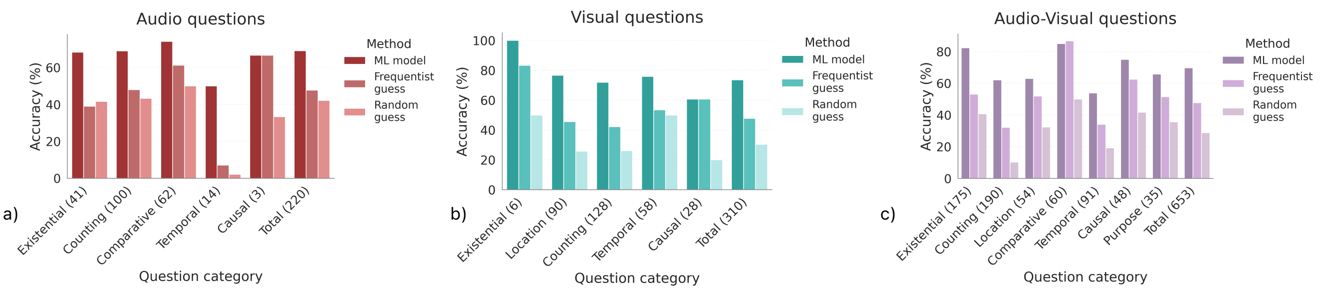

Figure 2: Comparison of AVST model performance with random and frequentist guessing across question categories. Total datapoints per category are indicated in brackets. Subgraphs: a) audio questions, b) video questions, c) audio-video questions.

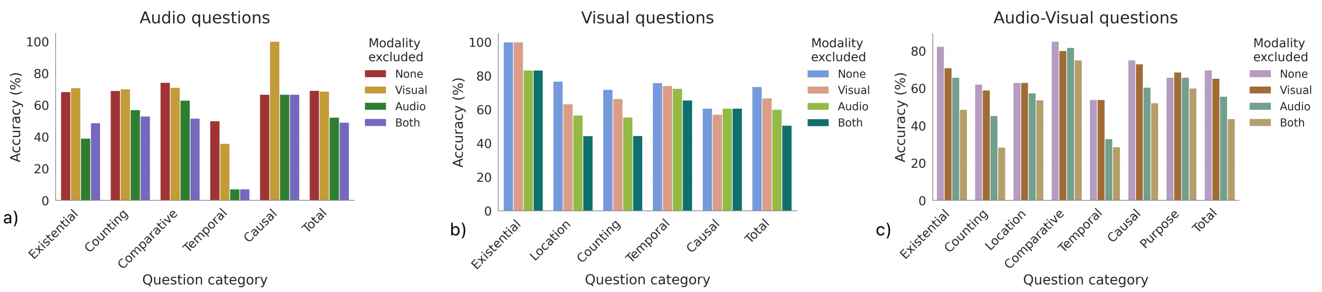

Figure 3: Test accuracy for the standard model with different modalities removed, across question categories. Subgraphs: a) audio, b) video, c) audio-video questions.

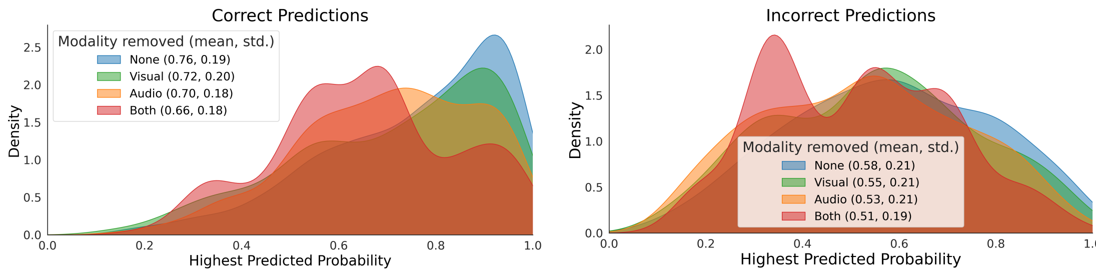

Figure 4: Distribution of the highest predicted probability for the standard model under modality ablations. Left: correct answers, right: incorrect answers. Mean and standard deviation are indicated.

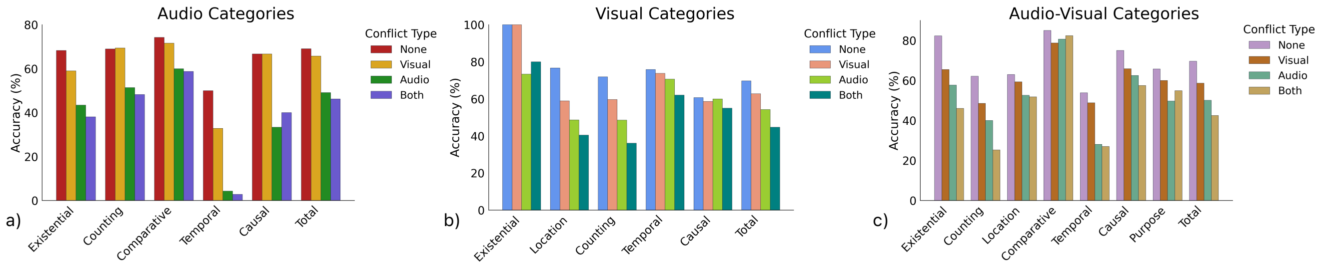

Figure 5: Overall accuracy on the test dataset with conflicts applied for a) Audio Sub-categories b) Video Sub-categories c) Audio-video Sub-categories.

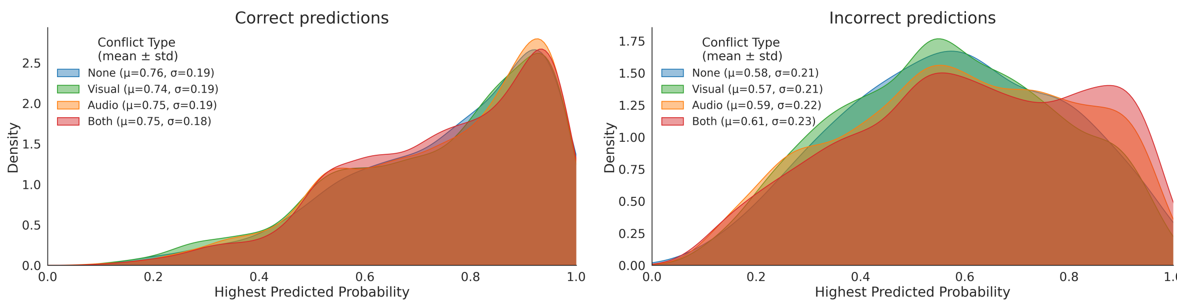

Figure 6: Distribution of highest predicted probability for the standard model on test data, for different conflict types. Mean highest predicted probability with standard deviation included. Correct answers shown to the left, incorrect to the right.

Beyond Musical Descriptors: Extracting Preference-Bearing Intent in Music Queries

Marion Baranes[1], Romain Hennequin[1] and Elena V. Epure[1, 2]
research@deezer.com

[1]Deezer Research, Paris, France,
[2]Idiap Research Institute, Martigny, Switzerland.

Abstract

Although annotated music descriptor datasets for user queries are increasingly common, few consider the user's intent behind these descriptors, which is essential for effectively meeting their needs. We introduce MusicRecoIntent, a manually annotated corpus of 2,291 Reddit music requests, labeling musical descriptors across seven categories with positive, negative, or referential preference-bearing roles. We then investigate how reliably large language models (LLMs) can extract these music descriptors, finding that they do capture explicit descriptors but struggle with context-dependent ones. This work can further serve as a benchmark for fine-grained modeling of user intent and for gaining insights into improving LLM-based music understanding systems.

1 Introduction

Users increasingly expect machines to understand complex and subjective natural-language queries for music search or recommendation (Doh et al., 2023; Gupta et al., 2023; Porcaro and Saggion, 2019; Delcluze et al., 2025; Palumbo et al., 2025; Melchiorre et al., 2025). While search engines reliably handle focused queries—those naming a specific artist, track, or album—they remain much less effective for open queries where user intent is exploratory (Hosey et al., 2019; Sguerra et al., 2022). Addressing open queries requires interpreting complex musical descriptors like genre (e.g., *pop, jazz*), mood (e.g., *sad, calm*), listening context (e.g., *party, driving*), instrumentation (e.g., *guitar, piano*), time period (e.g., *80s*), or geographical origin (e.g., *Spanish music*).

However, understanding descriptors alone is insufficient: it is also necessary to determine how each relates to the user's intent—whether it expresses a desired attribute (e.g., *I want to listen some Rock*), an undesired one (e.g., *Recommend me anything except Elvis Presley.*), or serves as a reference point (e.g., similarity: *I want more recent music, but like Elvis Presley.*). We capture this distinction by assigning each descriptor a **preference-bearing intent**, expressing either *positive* affinity (+), *negative* aversion (-), or a *referential* role with softer notions of similarity (∼). Although several works (Hachmeier and Jäschke, 2025; Salganik et al., 2025a) focus on music descriptor extraction or annotation, none consider the associated preference-bearing intents, and it remains unclear whether LLMs can robustly capture both the breadth of musical attributes and the preference-bearing roles of these descriptors.

To fill these gaps, we propose: (1) *MusicRecoIntent*, an annotated corpus of music-related queries, with each descriptor linked to a preference-bearing role, enabling fine-grained analysis of user intents.[1] (2) A benchmark of popular LLMs for this extraction task. (3) A qualitative analysis of systematic errors in manual and automatic annotations.

2 Related Work

Several datasets focus on music metadata extraction, particularly named entities (NE) such as artists or track titles. Epure and Hennequin (2022) created *MusicRecoNER*, a Reddit-based corpus of music recommendation queries, showing that extracting musical entities is non-trivial. Hachmeier and Jäschke (2024, 2025) extended this work using LLMs on queries extracted from Reddit, but also YouTube. Other studies address semantic aspects of musical descriptions. Salganik et al. (2025a,b) introduced MusicSem, a large language–audio corpus annotated across five semantic categories, while Weck et al. (2024) extract descriptors such as genre, style, mood, instrumentation, and tempo from Wikipedia. Additional datasets, like MusicCaps (Agostinelli et al., 2023) or JamendoMax-

[1]Dataset is available at https://github.com/deezer/MusicRecoIntent-NLP4MusA26.

Proceedings of the 4th Workshop on NLP for Music and Audio (NLP4MusA 2026), pages 20–26
March 28, 2026 ©2026 Association for Computational Linguistics

Caps (Roy et al., 2025), provide captions and aspect lists for musical segments, linking audio tracks with semantic descriptions.

Information Extraction (IE) aims to extract structured information from unstructured text, encompassing tasks such as Named Entity Recognition. In the musical domain, Hachmeier and Jäschke (2025) show that LLMs outperform smaller models in detecting musical entities, although performance strongly depends on prior exposure of LLMs to the entities. Similarly, Salganik et al. (2025b) demonstrate that LLMs can extract detailed semantic information from music descriptions, highlighting their growing adoption in music-related IE.

Beyond IE, determining the preference-bearing role of each descriptors remains challenging. Negation detection studies indicate that LLMs often struggle, where semantically contradictory statements are treated as equivalent (Kim et al., 2025; Vrabcová et al., 2025). Similarity detection in user musical queries has been shown by Palumbo et al. (2025) to be reliably handled by LLMs.

Despite recent advances, existing datasets and solutions fail to capture complex user intentions, including negation, or softer notions of similarity. This work fills these gaps by introducing a dataset for benchmarking LLMs' ability to model musical descriptors and their preference-bearing roles, supporting a richer understanding of user intent.

3 MusicRecoIntent Dataset

Our corpus is based on *MusicRecoNER* (Epure and Hennequin, 2023), which contains English-language music recommendation requests collected from Reddit[2]. These requests, wrote by users for other users, have not been corrected or standardized, which explains their variable quality. They are mostly open-ended with their length varying according to the level of detail and complexity. For the purpose of the present study, only one third of the dataset was retained, namely 2,291 user queries.

Manual Annotation. The annotation task aimed to label all musical descriptors in each query according to whether the user wanted them, wanted something similar, or wanted to avoid them. Two annotators were instructed to annotate descriptors in the categories introduced in Section 1, together with their corresponding preference-bearing intent.

[2] https://www.reddit.com/

Validation. We measured inter-annotator agreement separately for descriptor extraction and preference-bearing roles. Cohen's Kappa is unsuitable for descriptor identification, which is multi-label and span-based. Thus, agreement was computed at the span level as a percentage. This metric naturally accounts for multiple descriptors per query and variations in annotated elements. For the preference-bearing intents, we relied on Cohen's Kappa. Results are reported in Table 1.

	Aggr. (%)	Pref. intent (Kappa)	# Descriptors in Common
Decade	86.7	0.889	92
Genre	78.2	0.634	716
Instrument	66.1	1.000	253
LC	66.8	1.000	160
Mood	69.4	0.898	410
NE	85.4	0.752	1677
Country	81.7	1.000	53
Global	77.1	0.927	3361

Table 1: Inter-annotators Agreements per Category

Overall, the agreement on descriptor extraction is substantial, with a global rate of 0.771. Agreement varies across categories, ranging from 0.661 for *Instrument* to 0.867 for *Decade*. The main disagreements are often due to typographical variations (e.g., *oppressive / oprressive*), spontaneous normalization or correction (e.g., *1980s / 80s*), segmentation differences (e.g., *frank oceans blonde / frank ocean, blonde*), or context-driven additions or omissions (e.g., *klassische / klassische musik*). The annotators noted the absence of a category dedicated to more structural musical information, such as rhythm or song composition.

The preference-bearing intent agreement was computed on descriptors extracted by both annotators, and is generally high: 0.927, indicating a strong agreement. Category-level Kappa scores range from 0.634 for *Genre* to 1 for *Instrument*, *Listening Context* (LC), and *Country*. Most descriptors were marked as desired (positive affinity), while named entities were frequently marked as referential, rather than being explicit targets. Negative preferences are quite rare across all categories.

Final Dataset Overview. To create the final dataset, the annotators reviewed all points of disagreement together and reached a consensus on each annotation. The final corpus contains a total of 3,935 annotations. Table 2 provides a detailed overview of descriptors per category, indicating whether they were annotated with a positive (+), negative (-), or referential (~) preference-bearing role. On average, each query contains between 1 and 3 elements per category.

	#Descr.	+	~	-	Coverage
Decade	94	84	9	1	3.03
Genre	792	749	25	18	3.37
Instru.	352	337	1	14	1.81
LC	217	215	2	0	1.12
Mood	551	539	0	12	2.13
NE	1870	253	1613	4	1.10
Country	59	57	2	0	1.59
Total	3935	2234	1652	49	1.51

Table 2: Descriptive Statistics of the Corpus by Category

4 Automatic Annotation with LLMs

To extract descriptors and preference-bearing intents, we rely on Ollama[3], an open-source framework that facilitates the use of a wide range of LLMs. We experiment with a diverse set of models and sizes: Gemma 3 (4B, 12B, 27B) (Gemma Team et al., 2025), LLaMA 3 (1.8B) (Grattafiori et al., 2024), Mistral (7B) (Jiang et al., 2023), and Qwen 3 (8B, 32B) (Yang et al., 2025). We design a single prompt that instructs the model to identify music entities, as well as the other descriptor types, using a broad definition to avoid imposing too many restrictions. The full prompt is provided in Appendix B.1. To guide the model and enhance consistency and accuracy, the task was illustrated with six concrete examples. A second prompt was employed to determine the preference-bearing intent of each descriptor. This subsequent prompt is provided in Appendix B.2.

Table 3 summarizes the number of descriptors extracted by the tested models. Models generally produced more descriptors than those in the manually annotated dataset. To ensure reliability and limit hallucinations, only descriptors similar enough to the original text were retained. The preference-bearing intent was predicted afterward for the descriptors generated by Gemma3:27b, the model with the best performance on the initial task.

Evaluation Metrics. The most common evaluation metrics used in the extraction of musical entities are precision, recall, and F-score. We extend these metrics to better analyze the types of errors our system may produce, following the methodology proposed by Batista (2018). For each prediction, we classify it as correct, missing, spurious, incorrect, or partial (overlaps partially with the expected entity). Then, results are evaluated for exact match only (a prediction is considered correct if its segmentation is exact) and partial match (A prediction is considered correct if its segmentation is at least partially accurate, i.e. if it has at least one word in common with the expected prediction). It should be noted that these scores depend both on annotation quality and on model performance.

LLM	Exact	Partial	# Descr.
Gemma3:4b	0.66	0.74	4405
Gemma3:12b	0.68	0.75	5010
Gemma3:27b	**0.69**	**0.76**	4860
LLaMA3.1:8b	0.60	0.68	4535
Mistral:7b	0.56	0.64	4813
Qwen3:8b	**0.69**	**0.76**	4628
Qwen3:32b	0.67	0.73	5034

Table 3: Overall F1-scores (Exact and Partial) and Number of Descriptors Extracted by Different LLMs.

Overall Results. Table 3 shows the performance of various LLMs. Gemma (12b, 27b) and Qwen 8b achieve the strongest overall performance, with the highest exact and partial F1-scores. In contrast, Llama 8b and Mistral 7b perform less consistently across descriptor categories. Overall, larger or more recent architectures tend to generalize better at this extraction task. Based on this, we selected Gemma 27B (which was faster than Qwen3-8B) for a detailed analysis and further experiments.

Descriptor categories	**Exact Match**			**Partial Match**		
	P	R	F1	P	R	F1
NE	0.86	0.82	0.84	0.93	0.88	0.90
Genre	0.83	0.80	0.82	0.92	0.88	0.90
Mood	0.85	0.78	0.82	0.93	0.85	0.89
LC	0.47	0.42	0.45	0.74	0.65	0.69
Instrument	0.69	0.62	0.66	0.85	0.76	0.80
Country	0.96	0.88	0.92	0.98	0.90	0.94
Decade	0.87	0.83	0.85	0.93	0.89	0.91
Overall	0.62	0.77	0.69	0.69	0.85	0.76

Table 4: Precision, Recall and F1-scores with Gemma3:27B

Results on Descriptors Extraction. Tables 4 shows that NEs are well recognized, with an Exact Match F1-score of 0.84. The model accurately identifies explicit and well-defined descriptors, such as Country, which achieves a F1-score above 0.90. Decade, Genre, and Mood are also well extracted. The comparison between Exact Match and Partial Match illustrates the effect of segmentation tolerance, boosting F1-score from 0.69 to 0.76. This indicates that the model often identifies descriptors correctly, but token boundaries are imperfect.

The relatively lower scores for Listening Context (LC) and Instrument are partly due to limitations

[3] https://ollama.com/

in the manual annotation process. LC often involves detailed descriptors, leading to annotation variability (e.g., "*I want music reco, in the summer my dad and me are doing a trip to norway*" - *[summer, norway]* vs. *['dad', 'summer', 'trip to norway']*), while Instrument performance is limited by segmentation issues (e.g., *electric guitar* vs. *guitar*). These factors explain the gap between Exact and Partial Match scores, highlighting the difficulty of achieving consistent span annotations, whether human- or LLM-generated.

Results on Preference-Bearing Intent Prediction. For this evaluation, we consider only the descriptors annotated both manually and by Gemma3:27B (3,030 descriptors). Among these descriptors, 89% of LLM predictions matched the ground-truth. Table 5 summarizes the results. Overall, the confusion matrix shows that the model performs well, though it tends to overpredict positive preferences at the expense of referential cases—231 cases are incorrectly predicted as positive. Although negative preferences are accurately extracted, their low frequency limits the robustness of the conclusions.

True \ Pred	+	–	~	#Descr.
+	0.94	0.01	0.05	1627
–	0.00	0.90	0.10	31
~	0.17	0.01	0.82	1372

Table 5: Normalized Confusion Matrix (per True Class) for Preference-Bearing Intent Prediction with Gemma 3:27B.

Qualitative analysis. When comparing manual annotations with Gemma-3 predictions (cf. Table 6), several recurring sources of disagreement emerge. In many cases, both the human annotator and the model identify the same underlying information but disagree on segmentation—some merge multiple tokens into a single descriptor, others split them into separate units. Boundary disagreements also arise from differences in expected granularity: predicted spans may be shorter or longer than the ground-truth, often reflecting annotation guideline ambiguities rather than genuine model errors.

Additional discrepancies involve truncated or slightly altered descriptors, particularly for named entities. These cases generally reflect minor lexical drifts rather than semantic misunderstanding. The model also tends to over-annotate highly generic musical terms (e.g., *music, song, album*), which, while domain-relevant, do not serve as meaningful descriptors. Omissions are also common, especially for song titles and artist names, whose surface forms often resemble ordinary text, making them difficult to distinguish from non-descriptive content. The scores in Table 4 illustrate the impact of these discrepancies. Exact match penalizes any deviation—including segmentation differences—whereas partial match tolerates boundary variations. Segmentation disagreements are pronounced for Listening Context and Instrument descriptors, partly due to guideline limitations that did not anticipate certain edge cases, explaining some MusicRecoIntent boundary inconsistencies.

For the second annotation task—determining the preference-bearing role of a descriptor as positive, negative, or referential—additional sources of disagreement arise. The most frequent occur when similarity requests are interpreted as positive preferences, particularly when phrasing includes "*like*" questioning the prompt design. Interestingly, nearly 80% of these cases involve named entities. Less common but more challenging are sentences with strong negation (e.g., *hate, not, unless*) while expressing an overall positive intent; in these cases, half of the errors concern musical genres. Examples include: "*I hate most rap but want to get more into the genre*" or "*Don't listen to a lot of EDM but I really like Porter Robinson's Shelter.*"

Finally, a subset of disagreements stems from genuine ambiguity, where multiple interpretations are possible. For example, "*more songs like 1000 Rounds by Pouya and Ghostemane*" may refer either to the song alone or to the song and its artists, and both readings are plausible.

5 Conclusion

By introducing *MusicRecoIntent*, a corpus annotated with musical descriptors and preference-bearing roles, this work provides a benchmark for fine-grained modeling of user intent in music-related queries. Our results show that LLMs reliably capture explicit, well-defined descriptors—such as named entities, country, genre, and mood, and predict preference-bearing roles with high accuracy for positive and negative cases, while referential roles are more challenging and often overpredicted. Beyond model performance, our analysis reveals shared challenges for both annotators and LLMs, particularly around boundary decisions, granularity, and semantic ambiguity, suggesting that future improvements will require clearer annotation guidelines and prompts.

References

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

David S. Batista. 2018. Named-entity evaluation metrics based on entity-level.

Mathieu Delcluze, Antoine Khoury, Clémence Vast, Valerio Arnaudo, Léa Briand, Walid Bendada, and Thomas Bouabça. 2025. Text2playlist: Generating personalized playlists from text on deezer. In *Advances in Information Retrieval: 47th European Conference on Information Retrieval, ECIR 2025, Lucca, Italy, April 6–10, 2025, Proceedings, Part V*, page 164–170, Berlin, Heidelberg. Springer-Verlag.

SeungHeon Doh, Minz Won, Keunwoo Choi, and Juhan Nam. 2023. Toward universal text-to-music retrieval. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Elena Epure and Romain Hennequin. 2023. A human subject study of named entity recognition in conversational music recommendation queries. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1281–1296, Dubrovnik, Croatia. Association for Computational Linguistics.

Elena V. Epure and Romain Hennequin. 2022. Probing pre-trained auto-regressive language models for named entity typing and recognition. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1408–1417, Marseille, France. European Language Resources Association.

. Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Raghav Gupta, Renat Aksitov, Samrat Phatale, Simral Chaudhary, Harrison Lee, and Abhinav Rastogi. 2023. Conversational recommendation as retrieval: A simple, strong baseline. In *Proceedings of the 5th Workshop on NLP for Conversational AI (NLP4ConvAI 2023)*, pages 155–160, Toronto, Canada. Association for Computational Linguistics.

Simon Hachmeier and Robert Jäschke. 2024. Information extraction of music entities in conversational music queries. In *Proceedings of the 3rd Workshop on NLP for Music and Audio (NLP4MusA)*.

Simon Hachmeier and Robert Jäschke. 2025. A benchmark and robustness study of in-context-learning with large language models in music entity detection. In *Proceedings of the 31th International Conference on Computational Linguistics*.

Christine Hosey, Lara Vujović, Brian St. Thomas, Jean Garcia-Gathright, and Jennifer Thom. 2019. Just give me what i want: How people use and evaluate music search. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, page 1–12, New York, NY, USA. Association for Computing Machinery.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Jinsung Kim, Seonmin Koo, and Heui-Seok Lim. 2025. Semantic inversion, identical replies: Revisiting negation blindness in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 21445–21482.

Alessandro B. Melchiorre, Elena V. Epure, Shahed Masoudian, Gustavo Escobedo, Anna Hausberger, Manuel Moussallam, and Markus Schedl. 2025. Just ask for music (jam): Multimodal and personalized natural language music recommendation. In *Proceedings of the Nineteenth ACM Conference on Recommender Systems*, RecSys '25, page 615–620. ACM.

Enrico Palumbo, Marcus Isaksson, Alexandre Tamborrino, Maria Movin, Catalin Dincu, Ali Vardasbi, Lev Nikeshkin, Oksana Gorobets, Anders Nyman, Poppy Newdick, Hugues Bouchard, Paul Bennett, Mounia Lalmas, Dani Doro, Christine Doig Cardet, and Ziad Sultan. 2025. You say search, i say recs: A scalable agentic approach to query understanding and exploratory search at spotify. In *Proceedings of the Nineteenth ACM Conference on Recommender Systems*, RecSys '25, page 1117–1121, New York, NY, USA. Association for Computing Machinery.

Lorenzo Porcaro and Horacio Saggion. 2019. Recognizing musical entities in user-generated content. *Computación y Sistemas*, 23(3):1079–1088.

Abhinaba Roy, Renhang Liu, Tongyu Lu, and Dorien Herremans. 2025. Jamendomaxcaps: A large scale music-caption dataset with imputed metadata. *arXiv preprint arXiv:2502.07461*.

Rebecca Salganik, Teng Tu, Fei-Yueh Chen, Xiaohao Liu, Kaifeng Lu, Ethan Luvisia, Zhiyao Duan, Guillaume Salha-Galvan, Anson Kahng, Yunshan Ma, and Jian Kang. 2025a. Musicsem: A dataset of music descriptions on reddit capturing musical semantics. In *Late-Breaking Demo, 26th Conference of the International Society for Music Information Retrieval (ISMIR 2025)*.

Rebecca Salganik, Teng Tu, Fei-Yueh Chen, Xiaohao Liu, Kaifeng Lu, Ethan Luvisia, Zhiyao Duan, Guillaume Salha-Galvan, Anson Kahng, Yunshan Ma, and Jian Kang. 2025b. Musicsem: A semantically rich language-audio dataset of organic musical discourse. In *AI for Music Workshop, 39th Conference on Neural Information Processing Systems (NeurIPS 2025)*.

Bruno Sguerra, Marion Baranes, Romain Hennequin, and Manuel Moussallam. 2022. Navigational, informational or punk-rock? an exploration of search intent in the musical domain. In *Proceedings of the 30th ACM Conference on User Modeling, Adaptation and Personalization*, UMAP '22, page 202–211, New York, NY, USA. Association for Computing Machinery.

Tereza Vrabcová, Marek Kadlčík, Petr Sojka, Michal Štefánik, and Michal Spiegel. 2025. Negation: A pink elephant in the large language models' room? *Preprint*, arXiv:2503.22395.

Benno Weck, Holger Kirchhoff, Peter Grosche, and Xavier Serra. 2024. *WikiMuTe: A Web-Sourced Dataset of Semantic Descriptions for Music Audio*, page 42–56. Springer Nature Switzerland.

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

A Disagreements Examples in the Dataset

Table 6 shows examples of disagreements observed between manual annotations and Gemma3:27b annotations.

Query	**Manual annotations**	**Gemma3:27b annotations**	**Disagreements Type**
songs similar to japanese ceremonial tea	['japanese ceremonial tea']	['japanese', 'ceremonial', 'tea']	Segmentation
more genre busting like wu tang vs beatles	['wu tang vs beatles']	['wu tang', 'beatles']	Segmentation
songs like kendrick lamars maad city	['kendrick lamars', 'maad city']	['kendrick lamars maad city']	Segmentation
nostalgic indie pop alt rock songs	['nostalgic', 'indie', 'pop', 'alt rock']	['nostalgic', 'indie pop', 'alt rock']	Segmentation
looking for a wedding first dance song	['wedding first dance song']	['wedding', 'dance']	Segmentation
music similar to these songs do you feel it chaos chaos	['do you feel it', 'chaos chaos']	['chaos chaos']	Omission
looking for lovechild of metallica	['lovechild', 'metallica']	['metallica']	Omission
songs based off of eminems phenomenal	['eminems', 'phenomenal']	['eminem', 'phenomenal']	Truncation
similar to otherside by the red hot chili peppers	['the red hot chili peppers', 'otherside']	['red hot chili peppers', otherside']	Truncation
music genre for grandiose stylistic trumpet	['grandiose stylistic trumpet']	['music', 'grandiose', 'stylistic', 'trumpet']	Over-detection + Segmentation
a sad song	['sad']	['sad', 'song']	Over-detection

Table 6: Examples of disagreements between manual annotations and Gemma3:27b annotations

B Prompts

B.1 Prompt for Musical Descriptors Extraction

You're an assistant specialized in music. Your aim is to detect and extract musical descriptors mentioned in sentences. We want to extract all types of musical descriptors cited in the given text: artist names (or group/band names) and all musical work of art as song title or album title, musical genre, decade, location, mood, listening context, instruments, etc. Output must be a list that can contains all the descriptors found. Here some examples of the output expected:
- eg 1 : text : "I love rock and roll" - output: ['rock and roll'],
- eg 2 : text : "I love sad spanish love songs " - output: ['sad', 'spanish', 'love'].
- eg 3 : text : "je veux du gros rap des années 80 pour faire la fête" - output: ['rap', 'années 80', 'fête'],
- eg 4 : text : "I want some french songs like j'irai ou tu iras - Céline dion and JJ Goldman" - output: [french, "Céline dion", "JJ Goldman", "j'irai ou tu iras"].
- eg 5 : text : "Pop music to cook and sing" - output: ['pop', 'cook', 'sing].
- eg 6 : text : "I love Whenever, Wherever - Shakira " - output: ["Shakira", "Whenever, Wherever"].
Do not explain what you are doing, do not add any information that is not in the text to process and do not modify or correct the extracted text. Write only the output in one line. If you don't find descriptors, write [].</Task>
Please extract the musical descriptors cited in the following text : "sentence"

B.2 Prompt for Preference-Bearing Intent Extraction

You're an assistant specialized in natural langage processing. You will receive: a user query, and a list of descriptors that appear in the query. For each descriptor, determine the user's intention toward it using the following labels:
"'+' : the user is explicitly looking for this descriptor.
"'~' : the user is looking for something similar or related, but not necessarily exactly this descriptor.
"'-' : the user wants to exclude this descriptor (indicated by negations such as no, not, without, avoid, exclude, etc.).
Output format is strict: Return only a Python list of tuples of the form: [(descriptor, intention), ...]. No explanations, no extra text.
Rules:
a) Assign '+' if the descriptor is explicitly mentioned in the query as something the user wants, without being negated or rejected.
b) Assign '-' if the query explicitly negates, excludes, rejects (e.g., no X, not X, without X, exclude X, avoid X, etc.).
c) Assign '~' if the descriptor is mentioned in a way that suggests the user is looking for something related, similar, or loosely connected, but not exactly that descriptor.
Here some examples of the output expected:
- eg 1: User query: 'dark 90s music' ; Descriptors: '['dark', '90s']' -> Expected output: [('dark', '+'), ('90s', '+')]
- eg 2: User query: 'music like Abba but more rock' ; Descriptors: '['Abba', 'rock']' -> Expected output: [('Abba', '~'), ('rock', '+')]
- eg 3: User query: 'rock music without guitar' ; Descriptors: '['rock', 'guitar']' -> Expected output: [('rock', '+'), ('guitar', '-')]
- eg 4: User query: 'Celine dion song without Goldman' ; Descriptors: '['Celine dion', 'Goldman']' -> Expected output: [('Celine dion', '+'), ('Goldman', '-')]
- eg 5: User query: 'Calm rock song similar to the beatles' ; Descriptors: '['Calm ', 'rock', the beatles]' -> Expected output: [('Calm', '+'), ('rock', '+'), ('the beatles', '~')]
Now process the following instance: User query: ¨sentence;̈ Descriptors: "desc"

How Far Can Pretrained LLMs Go in Symbolic Music? Controlled Comparisons of Supervised and Preference-based Adaptation

Deepak Kumar*, **Emmanouil Karystinaios***,
Gerhard Widmer, **Markus Schedl**
Johannes Kepler University Linz, Austria
deepak.kumar,emmanouil.karystinaios,gerhard.widmer,markus.schedl@jku.at

Abstract

Music often shares notable parallels with language, motivating the use of pretrained large language models (LLMs) for symbolic music understanding and generation. Despite growing interest, the practical effectiveness of adapting instruction-tuned LLMs to symbolic music remains insufficiently characterized. We present a controlled comparative study of finetuning strategies for ABC-based generation and understanding, comparing an off-the-shelf instruction-tuned backbone to domain-adapted variants and a music-specialized LLM baseline. Across multiple symbolic music corpora and evaluation signals, we provide some insights into adaptation choices for symbolic music applications. We highlight the domain adaptation vs. preserving prior information tradeoff as well as the distinct behaviour of metrics used to measure the domain adaptation for symbolic music.

1 Introduction

Symbolic music can be represented as discrete token sequences that exhibit long-range dependencies, hierarchical organization, and strong syntactic and structural constraints. These properties make pretrained large language models (LLMs) a great candidate for symbolic music generation and for text-conditioned musical reasoning tasks. At the same time, symbolic music differs from standard text domains in that small local errors can invalidate entire sequences, and musically meaningful structure often spans many measures, creating adaptation and evaluation challenges that are not well understood.

Recent work has demonstrated that transformer-based models can generate symbolic music and answer music-theoretic questions, either through training from scratch, music-domain pretraining, or instruction-tuned pipelines (Yuan et al., 2024; Wang et al., 2025; Xu et al., 2025; Bhandari et al., 2025; Mundada et al., 2025). Of these works, ChatMusician (Yuan et al., 2024) is the only work to adapt a language model for the symbolic music domain in the literature. The effectiveness of adapting general LLMs to symbolic music therefore remains insufficiently characterized. In practice, researchers and practitioners face a pragmatic question: given an off-the-shelf LLM, can it be adapted to the symbolic music domain via standard domain adaptation practices of NLP as seen for other domains (e.g., medicine, law, coding) (Ling et al., 2025; Li et al., 2024; Chen et al., 2025).

In this work, we study the adaptation of a language model to symbolic music in a controlled setting using ABC notation, a plain-text format that encodes notes, rhythms, meter, and key as ASCII characters. Its compact structure and broad public corpus availability simplify tokenization and training compared with the more verbose **kern or XML-based encodings. We compare a base instruction-tuned backbone to domain-adapted variants trained with supervised instruction tuning and preference-based optimization, and we include a music-specialized LLM baseline for reference. To support both symbolic generation and grounded musical reasoning, we curate and unify instruction data from multiple sources and task formats, combining composition-oriented corpora with concept-focused understanding prompts paired with ABC excerpts.

Our study addresses two research questions: i) how effectively do common finetuning strategies adapt an LLM for symbolic music understanding and generation, and ii) what tradeoffs arise between music-domain gains and retention of general-domain capability. We provide empirical comparisons across symbolic corpora and general benchmarks, and we discuss vocabulary expansion through music-specific tokens as a complementary direction for improving representation and genera-

*These authors contributed equally to this work.

Proceedings of the 4th Workshop on NLP for Music and Audio (NLP4MusA 2026), pages 27–32
March 28, 2026 ©2026 Association for Computational Linguistics

tion fidelity.

2 Related Work

LLMs for symbolic music have recently been explored by representing music notation as text-like sequences, enabling next-token prediction, prompting, and instruction following over symbolic scores. Music-specialized models and pipelines have demonstrated promising capabilities for composition and music-theoretic question answering when trained or adapted on large corpora of symbolic data, often using ABC, **kern, or MusicXML-derived encodings, such as ChatMusician (Yuan et al., 2024), and MuPT (Qu et al., 2025).

Instruction tuning for music tasks extends these foundations by pairing natural-language prompts with symbolic outputs or analyses, with the goal of controllable generation and grounded reasoning about musical structure. Prior systems show that careful prompt templating and multi-stage generation can improve controllability, for example, by separating planning from realization or by providing intermediate constraints such as chord progressions or form descriptions, such as ComposerX (Deng et al., 2024), MuseCoco (Lu et al., 2023), etc.

Training from scratch on symbolic music, typically with music-native objectives and large-scale corpora, remains a strong alternative to adapting general LLMs. Such models can benefit from domain-specific inductive bias in data and tokenization, and may achieve strong distributional fidelity and coherence when the training scale is sufficient, e.g. NotaGen (Wang et al., 2025).

Alignment and preference learning methods, widely used in NLP to refine instruction-following behavior, have also begun to appear in music settings (Wang et al., 2025). Techniques such as preference-based optimization and related alignment objectives can be used to bias generation toward outputs that satisfy structural constraints, stylistic goals, or user intent, potentially reducing the gap between likelihood-based training and human-facing quality criteria.

3 Method

3.1 Task and Evaluation Scope

We focus on symbolic music in ABC notation and study two tasks: (1) generation or continuation, where the model produces ABC sequences conditioned on prompts or partial contexts, and (2) answering concept-rooted questions about music using ABC excerpts and metadata as context. In relation to both capabilities, we focus on symbolic music data from different sources, split into short sequence lengths and long sequence lengths. This setup allows us to study how LLMs adapt to symbolic music when trained on ABC data sources of varying quality and under different context length constraints.

Algorithm 1: ABC Degradation Pipeline

```
Input: ABC notation string A, maximum removable bars B_max
Output: Degraded ABC notation string A'
Split A into a list of lines L;
Initialize L' ← L;
Key Changing;
foreach line l ∈ L' do
    if l is a key declaration then
        Replace key signature with a fixed alternative key (e.g.,
          D#);
Random Pitch Swap;
Define note pool P = {C, D, E, F, G, A, B, c, d, e};
foreach line l ∈ L' do
    if l is not a metadata line then
        foreach character c in l do
            With probability p, replace c with a random note
              from P if c ∈ P;
Truncate bars;
Extract all musical (non-metadata) lines M from L';
if |M| > 1 then
    Sample k ~ U(1, min(B_max, |M| − 1));
    Randomly select k indices from M;
    Remove the selected lines from M;
    Set L' ← M;
return Concatenate lines in L' to form A'
```

Statistic	Short Train	Short Test	Long Train	Long Test
# Samples	88,575	22,306	104,270	26,229
Avg. Input Length	178.2	178.4	201.7	199.8
Avg. Target Length	275.8	275.4	4,808.4	4,733.4
Max Target Length	500	500	615,538	710,425
Avg. Bars	10.2	10.3	233.0	229.1
Notes per Bar	6.11	6.14	4.79	4.77

Table 1: Dataset statistics for short- and long-target ABC datasets.

3.2 Data Curation and Unification

We build a unified instruction-style training set by combining multiple sources that cover complementary aspects of symbolic music. We include composition-oriented corpora, such as collections of ABC tunes (monophonic) and structured symbolic datasets such as PDMX, high-quality classical music translated to ABC such as the Distant Listening Corpus (DLC) (Hentschel et al., 2025), Open Lieder (Gotham and Jonas, 2022) and Open String Quartets (Gotham et al., 2023), and understanding-

Model	Musicpile-Short		PDMX-Short		ABCTunes-Short		MMLU	Musicpile-Long		PDMX-Long		ABCTunes-Long	
	PPL↓	FMD ↓	PPL↓	FMD ↓	PPL↓	FMD ↓	↑	PPL↓	FMD ↓	PPL↓	FMD ↓	PPL↓	FMD ↓
LLaMA 3.1 Inst. 8B (Weerawardhena et al., 2025)	$693.9_{0.0}$	$66.5_{0.2}$	$9.0_{0.0}$	$320.4_{0.1}$	$28.2_{0.0}$	$467.2_{0.3}$	$0.59_{0.0}$	$6.0_{0.0}$	$130.1_{0.0}$	$3.3_{0.0}$	$475.8_{0.0}$	$5.1_{0.0}$	$449.6_{0.1}$
ChatMusician	$502.3_{0.0}$	$66.0_{0.3}$	$3.5_{0.2}$	$298.0_{0.0}$	$9.0_{0.0}$	$172.0_{0.0}$	$0.24_{0.0}$	$3.0_{0.0}$	$250.7_{1.3}$	$2.3_{0.6}$	na	$2.8_{0.0}$	na
Trained on short music sequences													
DPO	$401.5_{14.2}$	$72.7_{0.0}$	$9.0_{0.4}$	$349.6_{0.0}$	$517.3_{3.9}$	$348.1_{0.0}$	$0.31_{0.0}$	$6.6_{0.2}$	$233.1_{10.5}$	$3.4_{0.3}$	$502.0_{0.0}$	$21.3_{13.4}$	$240.5_{0.0}$
SFT	$36.3_{0.5}$	$362.1_{1.5}$	$3.7_{0.0}$	$350.3_{0.6}$	$9.3_{0.0}$	$265.25_{23.4}$	$0.29_{0.0}$	$4.5_{0.3}$	$291.3_{0.0}$	$2.4_{0.0}$	$455.5_{5.0}$	$3.4_{0.0}$	$253.9_{7.5}$
Trained on long music sequences													
DPO	$432.7_{0.0}$	$72.4_{0.0}$	$7.2_{0.0}$	$327.1_{0.0}$	$27.4_{0.0}$	$356.5_{0.0}$	$0.33_{0.0}$	$5.7_{0.0}$	$130.0_{2.0}$	$3.2_{0.1}$	$496.3_{4.2}$	$5.7_{0.2}$	$351.7_{19.2}$
SFT	$56.2_{0.0}$	$67.5_{0.0}$	$3.7_{0.0}$	$366.6_{0.0}$	$12.7_{0.0}$	$218.8_{0.0}$	$0.27_{0.0}$	$4.7_{0.0}$	$130.9_{5.5}$	$1.9_{0.0}$	$433.0_{0.8}$	$4.0_{0.3}$	$209.2_{10.4}$

Table 2: Perplexity (PPL) and Frechét Music Distance (FMD) across short sequence length symbolic music datasets and MMLU.

We report **"na"** for ChatMusician as results could not be calculated due to non-redable generation for FMD calcualtion and extremely slow evaluation due to lack padding, which restricted in resolving the issue

oriented prompts based on MusicPile-sft (Yuan et al., 2024). For the latter, we filter out examples dominated by caption-like or metadata-centric content (for example, YouTube-style descriptions) to better emphasize grounded musical concepts, terminology, and reasoning. All sources are normalized into a common ABC-centric schema with consistent separators and explicit fields for prompt, context, and target.

The data is split into two sets: long and short, based on the average sequence length of the target across the datasets. In the unified dataset, this splitting criterion is a sequence length of 500. Note that for short sequence length, the data from only three sources qualify, while for long sequence length, data from all sources qualify. Table 1 provides the key statistics of the unified datasets' splits.

3.3 Finetuning Strategies

We compare several practical adaptation strategies that are common in NLP but underexplored for symbolic music. Our primary baseline is an off-the-shelf instruction-tuned LLM (LLaMA 3.1 Inst. 8B) (Weerawardhena et al., 2025). We then train a supervised finetuning (SFT) (Ouyang et al., 2022) variant on the unified instruction data, optimizing next-token likelihood over the concatenated prompt, context and target with standard token-level cross-entropy loss for next token prediction. In addition, we train a preference-based variant using paired outputs for the same prompt, optimizing a contrastive objective that increases preference for the chosen output relative to a rejected alternative.

For preference-based training, we construct rejected outputs via a set of musically motivated degradations applied to the chosen target, as specified in Algorithm 1, while preserving ABC syntactic validity throughout. This provides a controllable source of negative samples that differ from the chosen output primarily in musical quality rather than formatting errors. We treat this signal as weak preference supervision rather than direct human preference data. Then we use the direct preference optimization (DPO) (Rafailov et al., 2023) technique to train the preference-based model.

We report results for Base, SFT, and DPO variants, and we include ChatMusician as a music-specialized reference point in our comparison. To maintain comparability with ChatMusician, we use a model from the LLaMA family (Touvron et al., 2023). We further adopt an instruction-tuned version, as instruction tuning has been shown to facilitate more effective knowledge acquisition under continued training (Jiang et al., 2024).

3.4 Metrics

We want to evaluate the LLM's performance on different music datasets to show understanding (MusicPile) and generation (rest) capabilities. We use perplexity (PPL) to measure the token-level closeness of generation to the target for understanding and music generation. In contrast, Fréchet Music Distance (Retkowski et al., 2025) (FMD) over CLAMP2 (Wu et al., 2025) is used to measure the global-level musical similarity of the generated music. We also measure the change in prior capabilities of the LLM. In this regard, we use the Massive Multitask Language Understanding (MMLU) benchmark (Hendrycks et al., 2020) to see the cost of adapting an LLM for the music domain. MMLU is a multiple-choice question answering benchmark over 57 different subjects (e.g., history, law) with varying levels of expertise.

4 Results and Analysis

Table 2 presents the results over the short sequence length music data and the MMLU benchmark. We observe that for music understanding (i.e., Musicpile), all models show improvement compared to the base model, and the SFT variant appears to be the best-performing. For music generation, if we focus on token-level similarity (i.e., PPL), we observe ChatMusician and SFT are performing well while DPO deteriorates. In contrast, global-level similarity, as assessed using FMD, does not exhibit clear trends across datasets. Prior knowledge retention seems inversely related to global-level similarity, with DPO least affected and ChatMusician most affected. The LLM trained on long sequences also showed some improvement on the short sequence test set. However, the improvement is generally weaker than the models trained on short sequences. In contrast to the short sequence test set, Table 2 shows that the long sequence test set proves to be harder to adapt for all variants.

5 Discussion and Limitations

General Capability Retention. Finetuning for symbolic music can degrade general-domain performance, particularly when training is narrow in domain or overly aggressive in optimization. This tradeoff is especially relevant for instruction-tuned backbones that are expected to maintain broad conversational and reasoning skills. In practice, preserving general capability may require careful data mixing, conservative training budgets, and multi-objective training. The inclusion of a music-specialized baseline provides a useful reference point: music-pretrained systems can deliver strong in-domain behavior, but may deteriorate in general instruction-following and out-of-domain robustness.

Evaluation Considerations. Likelihood-based metrics such as perplexity capture token-level fit to a corpus, but they do not fully reflect musical quality, long-range structure, or instruction adherence. Distributional proxy metrics, such as FMD, can complement perplexity, but their correlation with human judgments for symbolic music is imperfect and may depend on representation and decoding settings. Overall, we highlight that metrics for symbolic music generation and understanding might be insufficient or incomplete.

Limitations. First, our preference-based training uses weak preference signals induced by musically motivated degradations while enforcing ABC syntactic validity. This reduces formatting confounds but does not replace human preference data, and it may bias learning toward specific error patterns. Second, our study focuses on ABC as a compact text representation; results may not transfer directly to other symbolic encodings with different granularity and constraints. Third, while we curate and unify multiple data sources, coverage remains uneven across genres, instrumentation, and levels of theoretical sophistication. Fourth, evaluation of "understanding" remains sensitive to dataset construction and prompt design, and broader task coverage is left for future work. Finally, compute and context-length constraints limit exploration of very long-form structure and multi-part scores.

6 Conclusion

In this work, we presented a controlled comparative study of practical finetuning strategies for adapting instruction-tuned LLM to symbolic music in ABC notation, spanning generation and concept-rooted understanding. By comparing an off-the-shelf backbone, domain-adapted variants, and a music-specialized reference model, we characterized consistent tradeoffs between in-domain symbolic music behavior and retention of general language capabilities.

As future work, we plan to evaluate vocabulary expansion by introducing music-specific special tokens that capture recurring ABC constructs and structural markers (for example, barlines, section labels, voice separators, or compact representations of common ornaments). This requires extending the tokenizer, initializing new embeddings, and adapting the output projection head. While left for future work, we outline this procedure as a promising way to reduce fragmentation of musically meaningful units and improve generation fidelity under limited finetuning budgets. As an additional adaptation direction, we will complete vocabulary expansion experiments, expand the task suite and evaluation toward deeper music-theoretic understanding and longer-horizon generation, and investigate complementary mechanisms such as vocabulary expansion and structured prompting or planning to improve coherence, controllability, and grounded reasoning.

Acknowledgments

This work was supported by the European Research Council (ERC) under Horizon 2020 grant #101019375 "Whither Music?", the Austrian Science Fund (FWF): Cluster of Excellence *Bilat-*

eral Artificial Intelligence (https://doi.org/10.55776/COE12), the doc.funds.connect project *Human-Centered Artificial Intelligence* (https://doi.org/10.55776/DFH23), and the *Intent-aware Music Recommender Systems* (https://doi.org/10.55776/P36413).

References

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

Rubing Chen, Jiaxin Wu, Jian Wang, Xulu Zhang, Wenqi Fan, Chenghua Lin, Xiaoyong Wei, and Li Qing. 2025. Benchmarking for domain-specific llms: A case study on academia and beyond. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 11606–11619.

Qixin Deng, Qikai Yang, Ruibin Yuan, Yipeng Huang, Yi Wang, Xubo Liu, Zeyue Tian, Jiahao Pan, Ge Zhang, Hanfeng Lin, et al. 2024. Composerx: Multi-agent symbolic music composition with llms. *arXiv preprint arXiv:2404.18081*.

Mark R. H. Gotham, Maureen Redbond, Bruno Bower, and Peter Jonas. 2023. The “OpenScore String Quartet” Corpus. In *Proceedings of the 10th International Conference on Digital Libraries for Musicology*, pages 49–57, Milan Italy. ACM.

Mark Robert Haigh Gotham and Peter Jonas. 2022. The OpenScore Lieder Corpus. In *Music Encoding Conference Proceedings 2021*, pages 131–136. Humanities Commons.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. In *International Conference on Learning Representations 2020*.

Johannes Hentschel, Yannis Rammos, Markus Neuwirth, and Martin Rohrmeier. 2025. A corpus and a modular infrastructure for the empirical study of (an) notated music. *Scientific Data*, 12(1):685.

Zhengbao Jiang, Zhiqing Sun, Weijia Shi, Pedro Rodriguez, Chunting Zhou, Graham Neubig, Xi Lin, Wen-tau Yih, and Srini Iyer. 2024. Instruction-tuned language models are better knowledge learners. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5421–5434.

Jiawei Li, Yizhe Yang, Yu Bai, Xiaofeng Zhou, Yinghao Li, Huashan Sun, Yuhang Liu, Xingpeng Si, Yuhao Ye, Yixiao Wu, et al. 2024. Fundamental capabilities of large language models and their applications in domain scenarios: A survey. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11116–11141.

Chen Ling, Xujiang Zhao, Jiaying Lu, Chengyuan Deng, Can Zheng, Junxiang Wang, Tanmoy Chowdhury, Yun Li, Hejie Cui, Xuchao Zhang, et al. 2025. Domain specialization as the key to make large language models disruptive: A comprehensive survey. *ACM Computing Surveys*, 58(3):1–39.

Peiling Lu, Xin Xu, Chenfei Kang, Botao Yu, Chengyi Xing, Xu Tan, and Jiang Bian. 2023. Musecoco: Generating symbolic music from text. *arXiv preprint arXiv:2306.00110*.

Gagan Mundada, Yash Vishe, Amit Namburi, Xin Xu, Zachary Novack, Julian McAuley, and Junda Wu. 2025. Wildscore: Benchmarking mllms in-the-wild symbolic music reasoning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 16858–16874.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Xingwei Qu, Yinghao Ma, Ziya Zhou, Ka Man Lo, Jiaheng Liu, Ruibin Yuan, Lejun Min, Xueling Liu, Tianyu Zhang, Xeron Du, et al. 2025. Mupt: A generative symbolic music pretrained transformer. In *The Thirteenth International Conference on Learning Representations*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.

Jan Retkowski, Jakub Stępniak, and Mateusz Modrzejewski. 2025. Frechet music distance: A metric for generative symbolic music evaluation. In *The 39th Annual AAAI Conference on Artificial Intelligence*. AAAI (Association for the Advancement of Artificial Intelligence.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Yashan Wang, Shangda Wu, Jianhuai Hu, Xingjian Du, Yueqi Peng, Yongxin Huang, Shuai Fan, Xiaobing Li, Feng Yu, and Maosong Sun. 2025. Notagen: Advancing musicality in symbolic music generation with large language model training paradigms. *arXiv preprint arXiv:2502.18008*.

Sajana Weerawardhena, Paul Kassianik, Blaine Nelson, Baturay Saglam, Anu Vellore, Aman Priyanshu, Supriti Vijay, Massimo Aufiero, Arthur Goldblatt, Fraser Burch, et al. 2025. Llama-3.1-foundationai-securityllm-8b-instruct technical report. *arXiv preprint arXiv:2508.01059*.

Shangda Wu, Yashan Wang, Ruibin Yuan, Guo Zhancheng, Xu Tan, Ge Zhang, Monan Zhou, Jing Chen, Xuefeng Mu, Yuejie Gao, et al. 2025. Clamp 2: Multimodal music information retrieval across 101 languages using large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 435–451.

Weihan Xu, Julian McAuley, Taylor Berg-Kirkpatrick, Shlomo Dubnov, and Hao-Wen Dong. 2025. Generating symbolic music from natural language prompts using an llm-enhanced dataset.

Ruibin Yuan, Hanfeng Lin, Yi Wang, Zeyue Tian, Shangda Wu, Tianhao Shen, Ge Zhang, Yuhang Wu, Cong Liu, Ziya Zhou, et al. 2024. Chatmusician: Understanding and generating music intrinsically with llm. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 6252–6271.

MIDI-PHOR: Multi-View Distillation for Music Understanding and Captioning

Steven Au
Independent Researcher
steventinwing@gmail.com

Abstract

A central limitation of current music understanding frameworks is the reliance on audio embeddings, which frequently yields interpretations lacking traceable ties to explicit musical elements such as notes, dynamics, and instrumentation. We address this gap with MIDI-PHOR, a MIDI-first framework that converts symbolic data into structured, queryable representations for reasoning. MIDI-PHOR distills each piece into three complementary views: a symbolic view capturing pitch, meter, and key; a time-series (TS) view that tracks rhythmic salience, texture, and role activity; and an instrument-role graph encoding ensemble interactions. With evidence-linked claims, experiments demonstrate reduced hallucinations compared to raw-MIDI baselines and offer a robust, auditable bridge between symbolic data and semantic music understanding.

1 Introduction

Music understanding is a foundational challenge in multimodal artificial intelligence, requiring systems to not only classify audio but to analyze, describe, and reason about complex structural relationships. While natural language processing has achieved remarkable interpretability through Large Language Models (LLMs), progress in music AI has been dominated by generative objectives. Systems such as MusicLM (Agostinelli et al., 2023) and MusicGen (Copet et al., 2024) focus primarily on learning probabilistic audio-text correspondences for synthesis. While these models excel at creative generation, they function as "black boxes" regarding musical structure, lacking the ability to provide traceable, verifiable justification for their outputs (e.g., citing specific notes, harmonic progressions, or orchestration techniques).

Symbolic data formats like MIDI offer a promising alternative by encoding music as discrete events—pitch, rhythm, instrumentation, and control changes—enabling analysis at a granular level. However, two significant barriers have hindered symbolic music understanding. First, symbolic data lacks the implicit timbral cues (e.g., specific synth textures or production effects) that standard models rely on to infer genre and mood. Second, existing attempts to bridge this gap, such as MIDI-Caps (Melechovsky et al., 2024), often rely on rendering MIDI to audio to extract features. We argue that this approach is flawed: it introduces non-deterministic variance based on the synthesizer or SoundFont used, confusing the model with artifacts that do not exist in the underlying composition.

We introduce MIDI-PHOR (**P**olyphonic **H**armony **O**rchestration **R**easoner), a framework that targets an explicit grounded reasoning layer with three queryable views. MIDI-PHOR treats MIDI not merely as a sequence of tokens, but as a structured knowledge object for music understanding rather than synthesis. To overcome the semantic gap without relying on unstable audio rendering, we distill MIDI into three complementary views:

1. A **Symbolic View** that captures ground-truth pitch, meter, key regions, and harmonic progression;
2. A **Time-Series View** that projects discrete tokens into continuous signal proxies (e.g., "symbolic spectral flux," "groove entropy") to represent texture and dynamics natively;
3. An **Orchestration Graph** that encodes ensemble roles and interactions between instruments(e.g., *locking*, *call-and-response*) as explicit edge relations.

We demonstrate that this multi-view representation significantly reduces hallucination compared to raw-MIDI baselines and offers a robust, invariant bridge between symbolic data and semantic description. Our core contribution is the introduction of a grounded reasoning layer that separates

Proceedings of the 4th Workshop on NLP for Music and Audio (NLP4MusA 2026), pages 33–43
March 28, 2026

language generation from musical truth verification. By distilling MIDI into a queryable relational backend, we provide a representation that constrains LLMs to descriptive claims strictly supported by MIDI-level evidence.

2 Related Work

Audio-Language Models. The field has bifurcated into generation and retrieval. Generative systems like MusicLM (Agostinelli et al., 2023) and MusicGen (Copet et al., 2024) utilize vast audio-text pairs to synthesize music, but they function as "black boxes" lacking explicit structural reasoning. Conversely, joint embedding models like CLAP (Elizalde et al., 2022) and MuLan (Huang et al., 2022) align audio and text in a shared latent space for zero-shot retrieval. While effective for global semantics (e.g., "happy jazz"), these embeddings often fail to capture granular musical details such as specific chord inversions or instrument roles. To address data scarcity, LP-MusicCaps (Doh et al., 2023) and MusicLingo (Deng et al., 2024) leverage LLMs to generate pseudo-captions, yet these remain bound by the subjectivity of the underlying audio analysis—a limitation highlighted by the Chordify Annotator Subjectivity Dataset, which shows only ∼54% expert agreement on complex harmony (Koops et al., 2019).

Symbolic Music Understanding. Symbolic data offers a path to objective analysis. Early pretraining approaches like MusicBERT (Zeng et al., 2021) applied masked language modeling to MIDI to learn structural representations, while recent foundation models like Moonbeam (Guo and Dixon, 2025) improve this with domain-aware tokenization. However, these models are primarily designed for sequence classification or generation, not interpretability. The closest precedent to our work is MIDI-Caps (Melechovsky et al., 2024), which introduced a large-scale MIDI-caption dataset. However, as shown in Table 1, MIDI-Caps and similar frameworks largely rely on *rendering* MIDI to audio to extract timbre and genre features. This dependency re-introduces the variance of audio synthesis (soundfonts, reverb) into the pipeline. MIDI-PHOR diverges from this by proposing a symbolic-only bridge, deriving texture and timbral proxies directly from token distributions to ensure invariant, auditable reasoning.

3 Methodology

We refit the music captioning task as a grounded reasoning problem, where the LLM functions as a claim selector over a structured musical knowledge base. To enable this, we construct a unified, bar-aligned representation of musical data designed for arrangement, structure, and symbolic reasoning. The pipeline consists of three view-specific extractors—(i) symbolic analysis, (ii) symbolic time-series projection, and (iii) orchestration graph construction—whose outputs are materialized in a relational DuckDB backend. This design makes every high-level caption claim traceable to explicit musical evidence (notes, tracks, bars, chords, and derived bar-level signals). We then distill these tables into LLM-friendly artifacts (ScoreSpec, ScoreSpec-Lite, enhanced facts, hierarchical facts) used for captioning and retrieval as shown in Figure 1.

To facilitate this architectural design, which ensures every high-level descriptive claim is traceable to explicit musical evidence, we define the following data hierarchy:

- *View*: One of the three analytical perspectives used to interpret the MIDI source.
- *Fact*: An atomic, bar-indexed entry in the relational backend.
- *Claim*: A natural language statement produced by the LLM describing a musical phenomenon.
- *Evidence Pointer*: A metadata link anchoring a generated claim to its corresponding fact in the database.

We distill these relational tables into four hierarchical artifacts—ScoreSpec, ScoreSpec-Lite, enhanced facts, and hierarchical facts—used for captioning and retrieval as shown in Figure 1. By using these artifacts, we effectively transform the task into a form of grounded extractive summarization, ensuring the generation process is strictly conditioned on retrieved facts rather than the latent priors of the language model. Detailed specifications for these artifacts are provided in Appendix A.1

3.1 Symbolic View Analysis

We process MIDI files using the robust event parsing of `miditoolkit` and `pretty_midi` to populate normalized tracks and notes tables, aligned to a bar backbone derived from the MIDI time-signature maps. For harmonic analysis, we leverage the musicological theory engine of `music21` to perform chordification, generating Roman-numeral annota-

Table 1: Comparison of current captioning systems by modality and output granularity. **TS** denotes Symbolic Time-Series proxies; **MIR** stands for Music Information Retrieval. **Claims** refers to evidence-linked descriptive statements.

System	Modality	Signal	Output(s)	Global	Section
LP-MusicCaps	Audio	Pseudo-captions	Caption	✓	
MusiLingo	Audio	Captioning	Caption	✓	
FUTGA	Audio	Temporal	5 captions	✓	✓
SonicVerse	Audio	Temporal/MIR	Long captions	✓	✓
MidiCaps	MIDI	Symbolic + Audio	Captions	✓	
MIDI-PHOR (Ours)	MIDI	Sym + TS + Graph	Claims + Evidence	✓	✓

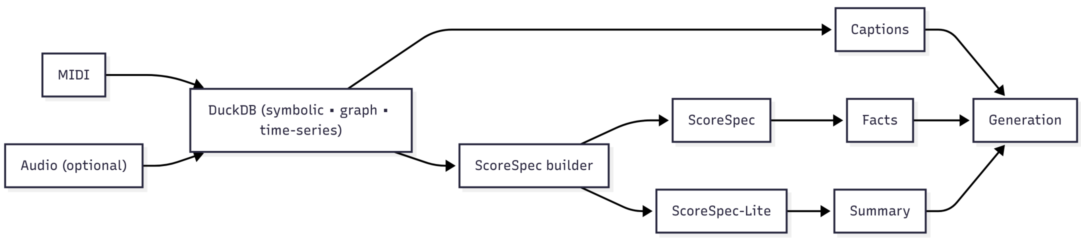

Figure 1: Pipeline from MIDI/audio to ScoreSpec, facts, and generation.

tions (rn) for harmonic summaries and deterministic cadence detection.

To handle modulation without destabilizing Roman-numeral labeling, we separate global and local key regions. A single global key estimate is stored in `key_changes` as the reference for Roman numerals, while a windowed tonal-center tracker stores time-varying local keys in `key_regions` (bar spans with confidence). This preserves consistency for RN-based reasoning while enabling analysis of local tonal shifts.

To ensure robustness to tempo changes, core arrangement statistics are bar-indexed rather than time-indexed. In particular, polyphony is computed on a tick-aligned grid inside each bar (yielding invariance to tempo scaling), while density is computed as normalized note-on activity per bar.

3.2 Symbolic Time-Series Projection

A key goal of MIDI-PHOR is to avoid dependence on rendered audio for structural cues. While prior work renders MIDI to audio and then applies signal processing for segmentation and tag prediction, synthesized audio introduces confounds tied to the chosen soundfont and rendering settings. Instead, we project discrete symbolic events into bar-level signals stored in `ts_bar`, enabling robust structure reasoning without waveform processing.

We compute pitch-class distributions (bar-level chroma) and derive a symbolic novelty signal (`novelty_bar`) from the distance between consecutive chroma vectors. We also compute repetition measures (`repeat_score_bar`, `recurrence_density_bar`) based on cosine similarity between bar chroma vectors. Rhythmic salience is estimated using onset entropy on a quantized 16th-note grid (`onset_entropy_16th`), alongside bar-level syncopation and backbeat proxies derived directly from symbolic onsets. These signals support deterministic structural event extraction (e.g., cadence peaks from Roman-numeral transitions) and enable section boundary discovery using novelty peaks.

3.3 Orchestration Graph Construction

To represent ensemble structure beyond token sequences, we construct a heterogeneous orchestration graph. Nodes represent tracks and sections, and edges represent interactions inferred from symbolic evidence. For example, a `RHYTHMIC_LOCK` relation is created when bass and drums exhibit strong co-activity patterns over multiple bars. We also infer accompaniment-style relations (e.g., `SUPPORTS_HARMONY_OF`) when pad/comp roles co-occur with melody across sustained spans, and `CALL_RESPONSE` patterns via bar-level alternation of track activity. Each edge is stored with evidence spans in bar units (`edge_evidence`), enabling auditable statements such as “the pad supports the melody” grounded in explicit bar ranges.

Table 2: Invariance checks ($N = 100$). Pass rate denotes the fraction of tracks where structural boundaries remain stable (Jaccard ≥ 0.9) under transformation.

Transformation	Pass	Expected Behavior
Transpose ($\pm$12 semi)	0.929	Harmonic content rotates.
Tempo ($\times$0.8–1.25)	0.929	Tick-based features remain.
Velocity ($\times$0.5–1.5)	0.929	Dynamics shift predictably.

Table 3: Factuality evaluation (Claim–Evidence verification, $N = 97$). MIDI-PHOR significantly reduces hallucinations compared to baselines.

Method	Claim Cov. ↑	Evid. Cov. ↑	Verify Pass ↑	Halluc. Rate ↓
MIDI-PHOR (Ours)	0.718	**0.990**	**0.804**	**0.186**
Symbolic-Only	**1.000**	0.974	0.701	0.317
Render+Audio	**1.000**	0.954	0.486	0.538

3.4 Distillation and Integration

The DuckDB representation is distilled into three complementary LLM-facing formats: (i) a section-aware scaffold (merged sections with active roles and salient signals), (ii) an enhanced flat fact list for prompt injection, and (iii) a hierarchical fact index organizing facts by domain (harmony, rhythm, orchestration, motifs). These are serialized through a standardized SCORESPEC JSON interface, ensuring that any generated caption is strictly conditioned on retrieved, auditable facts rather than latent priors.

4 Evaluation

We evaluate MIDI-PHOR across three dimensions: (i) feature reliability via invariance testing, (ii) factual grounding via claim verification, and (iii) computational efficiency. Experiments utilize a random sample of 100 tracks from the Clean Lakh MIDI dataset (Raffel, 2016).

4.1 Reliability via Invariances

Lacking large-scale ground truth for structural events (e.g., section boundaries), we use *invariance checks* as a proxy for robustness. We apply controlled perturbations—transposition, tempo scaling, and velocity scaling—to the input MIDI and measure the stability of detected section boundaries. A trial is considered a pass if the Jaccard similarity between the original and perturbed boundary sets is ≥ 0.9. Table 2 confirms that MIDI-PHOR's symbolic extractors remain stable under standard musical transformations, whereas audio-based features often drift due to synthesis artifacts.

4.2 Factuality and Hallucination

To quantify grounding, we employ a *Claim–Evidence* verification protocol. The model generates a caption as a JSON list of claims, where each claim includes a value and a specific evidence pointer (e.g., a bar range). We verify these claims against the underlying database.

We compare MIDI-PHOR against two baselines: (1) *Symbolic-Only* (standard LLM prompting with raw MIDI tokens), and (2) *Render+Audio* (synthesized audio with MIR-based taggers, mimicking MIDI-Caps). As shown in Table 3, MIDI-PHOR achieves the highest verification pass rate (0.804) and the lowest hallucination rate (0.186). The *Render+Audio* baseline struggles with hallucination (0.538), primarily because the LLM generates descriptions unsupported by the specific synthesis used. See Appendix **??** for baseline specifications

Note that claim coverage should be interpreted under a fixed claim budget rather than as absolute completeness.[1] Within this budget, MIDI-PHOR utilizes available slots most effectively (0.990 evidence coverage).

4.3 Prompt Sensitivity

To assess stability, we generated captions using three distinct prompts on a 30-song subset. The set of retrieved slots remained consistent (Jaccard similarity 0.769). When a specific slot was claimed, the value agreed across prompts 84.3% of the time, with perfect consistency in evidence pointers. This indicates that MIDI-PHOR's extracted view effectively constrains the LLM to the underlying data regardless of phrasing changes.

4.4 Computational Profile

We report end-to-end runtime normalized by song duration. MIDI-PHOR averages 2.36 s/min on a standard CPU, noticeably outperforming the Render+Audio baseline (3.53 s/min). This performance gap validates our hypothesis regarding the limitations of audio-based approaches: the baseline

[1] We cap the number of emitted claims to K=10 (`max_claims`) to keep outputs concise and comparable across methods, bound token costs, and prevent trivial coverage inflation from listing every available slot. Under this budget, claim coverage reflects how effectively a method uses a fixed claim budget, not how many total facts exist in the representation.

is dominated by the synthesis step (2.72 s/min), which introduces significant external dependencies and disk overhead. By bypassing this bottleneck, MIDI-PHOR not only achieves faster processing but also maintains a self-contained pipeline with a direct, inspectable link from symbolic events to the generated evidence.

5 Conclusion

We have introduced MIDI-PHOR, a framework that addresses the lack of interpretability in music understanding by reframing captioning as a grounded reasoning task. By distilling MIDI into a structured knowledge base comprising a symbolic scaffold, a synthesis-free time-series projection, and an orchestration graph, we provide a representation that is both machine-readable and human-auditable.

Our approach departs from "black-box" audio-text mapping to prioritize explicit reasoning over grounded evidence. The evaluation results validate this architectural choice: MIDI-PHOR demonstrates robustness to musical transformations where audio features frequently degrade, while achieving a significant reduction in hallucination rates (18.6% vs. 53.8% for audio-based baselines). By forcing the generation process to cite specific bars, tracks, and graph edges, we ensure that claims about harmony, texture, and interplay are strictly traceable to the source data. This framework offers a practical path toward trusted music AI, where descriptive outputs are not just plausible, but verifiable.

Limitations

Lack of Ground Truth. The primary limitation of this work is the scarcity of large-scale, human-annotated ground truth for the structural phenomena we extract (e.g., specific *drop* locations, *call-and-response* segments). Consequently, our evaluation relies on internal consistency checks—such as invariance testing and claim-evidence verification—rather than precision/recall metrics against a gold standard. While invariance confirms that our detectors are stable, it does not strictly prove their perceptual validity to a human listener.

Absence of Vocal and Lyrical Content. Standard MIDI files encode instrumental performance data but inherently lack audio vocals and lyrical text. Consequently, MIDI-PHOR cannot extract features related to vocal timbre or thematic content, which are often central to high-level musical understanding. While MIDI can encode a vocal melody line symbolically, recovering the full semantic context represents a challenge intrinsic to the format. Addressing this limitation would necessitate the development of fusion models that integrate symbolic representations with aligned audio or textual modalities.

Heuristic Nature and Data Dependency. Several graph edge detectors (e.g., `RHYTHMIC_LOCK`) rely on threshold-based heuristics. While tuned for pop/rock structures in the Lakh MIDI Dataset, these heuristics may require recalibration for genres with different rhythmic conventions, such as Jazz or Classical. Furthermore, the system is bound by the quality of the input symbolic data; MIDI files with missing percussion tracks, extreme quantization, or malformed events will inevitably degrade the quality of the extracted features and the resulting captions.

Ethics Statement

Subjectivity and Affective Labeling. Music description is inherently subjective. A core ethical risk in automated captioning is the presentation of affective labels (e.g., *sad*, *triumphant*) or stylistic interpretations as objective facts. MIDI-PHOR is designed to mitigate this by prioritizing structural evidence over emotive language. However, users should remain aware that even symbolic analysis can encode bias; for example, the definition of *complexity* or *consonance* is culturally specific. Generated captions should be interpreted as evidence-based descriptions of the file content, not authoritative musicological or emotional analysis.

Bias in Symbolic Corpora. Symbolic datasets like the Lakh MIDI Dataset predominantly represent Western popular music traditions. Consequently, the extracted features—particularly those related to harmony (Roman Numerals) and meter (4/4 backbeat)—may under-represent or mischaracterize music from non-Western traditions or avant-garde traditions that do not adhere to standard tonal and metric grids. We position MIDI-PHOR as a complementary tool to audio-first approaches, acknowledging that it is most effective within the specific cultural contexts where MIDI is a standard representation.

References

Andrea Agostinelli, Timo I. Denk, Zalán 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. *Preprint*, arXiv:2301.11325.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. 2024. Simple and controllable music generation. *Preprint*, arXiv:2306.05284.

Zihao Deng, Yinghao Ma, Yudong Liu, Rongchen Guo, Ge Zhang, Wenhu Chen, Wenhao Huang, and Emmanouil Benetos. 2024. MusiLingo: Bridging music and text with pre-trained language models for music captioning and query response. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3643–3655, Mexico City, Mexico. Association for Computational Linguistics.

SeungHeon Doh, Keunwoo Choi, Jongpil Lee, and Juhan Nam. 2023. Lp-musiccaps: Llm-based pseudo music captioning. *Preprint*, arXiv:2307.16372.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. 2022. Clap: Learning audio concepts from natural language supervision. *Preprint*, arXiv:2206.04769.

Zixun Guo and Simon Dixon. 2025. Moonbeam: A midi foundation model using both absolute and relative music attributes. *Preprint*, arXiv:2505.15559.

Qingqing Huang, Aren Jansen, Joonseok Lee, Ravi Ganti, Judith Yue Li, and Daniel P. W. Ellis. 2022. Mulan: A joint embedding of music audio and natural language. *Preprint*, arXiv:2208.12415.

Hendrik Vincent Koops, W. Bas de Haas, John Ashley Burgoyne, Jeroen Bransen, Anna Kent-Muller, and Anja Volk. 2019. Annotator subjectivity in harmony annotations of popular music. *Journal of New Music Research*, 48(3):232–252.

Jan Melechovsky, Abhinaba Roy, and Dorien Herremans. 2024. Midicaps: A large-scale MIDI dataset with text captions. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR 2024)*.

Colin Raffel. 2016. *Learning-Based Methods for Comparing Sequences, with Applications to Audio-to-MIDI Alignment and Matching*. Ph.D. thesis, Columbia University.

Mingliang Zeng, Xu Tan, Rui Wang, Zeqian Ju, Tao Qin, and Tie-Yan Liu. 2021. MusicBERT: Symbolic music understanding with large-scale pre-training. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 791–800, Online. Association for Computational Linguistics.

A Supplementary Results

This appendix consolidates evaluation details and qualitative examples that are referenced but not expanded in the main paper.

A.1 Artifact Design and Reproducibility

To ensure reproducibility and provide a template for the captioning task, we extract analysis at four abstraction levels. These artifacts were designed to distill every note and control change from the granular MIDI event stream into a structured, queryable JSON representation.

- **ScoreSpec**: Encodes fine-grained musical data, including segments, instrument metadata, pitch-class spans, and motif occurrences, alongside a complete structural graph.
- **ScoreSpec-Lite**: A section-level summary (tempo, bar ranges, density, active instruments, and harmonic summaries) designed for rapid inspection and lower token overhead.
- **Enhanced Facts**: A human-readable bullet list of salient features generated from structured summaries for direct prompt injection.
- **Hierarchical Facts**: Reorganizes findings into analytic categories (structure, harmony, rhythm, orchestration, motifs, and form) to support topic-focused reporting and precise querying.

A.2 LLM Auditing Metrics and Prompt Sensitivity

We introduce a suite of auditing metrics to measure the structural integrity of the generated captions, verifying them against the ground-truth database rather than relying solely on surface-level text overlap. Table 4 details these results. The *Evidence coverage* score of 0.990 shows that the model consistently cites specific bar ranges or track IDs, allowing users to trace statements back to the source signal. *Claim coverage* (0.718) reflects the fraction of available database slots that appear in the final output. This value remains below 1.0 because the system encourages concise captions capped at a fixed claim budget, prioritizing readability over exhaustive enumeration. Despite this selectivity, the *Verification pass rate* of 0.804 and low *Hallucination rate* of 0.186 demonstrate that the model maintains high factual accuracy for the claims it generates.

We further assessed the robustness of the model's grounding by testing three prompt variations on a subset of 30 songs. The *Slot-set stability* score of 0.769 indicates that changing the prompt effectively steers the model to select different features, such as focusing on harmony versus rhythm. Even when the topical focus shifts, the underlying

Table 4: LLM auditing metrics (measured). Values are computed by matching generated claim slots and evidence pointers to extracted database facts.

Metric	Value	Interpretation
Claim coverage	0.718	Fraction of expected slots that the model attempts to fill
Evidence coverage	0.990	Fraction of claimed slots that include a valid evidence pointer
Verification pass rate	0.804	Fraction of claimed slots whose values match DB ground truth
Hallucination rate	0.186	Fraction of claimed slots that fail verification
Slot-set stability (Jaccard)	0.769	Prompt sensitivity: overlap of which slots are claimed (30 songs, 3 prompts)
Value agreement	0.843	Prompt sensitivity: agreement of claimed values when the slot is claimed
Evidence agreement	1.000	Prompt sensitivity: agreement of evidence pointers under prompt variants

reasoning remains deterministic. The perfect *Evidence agreement* of 1.000 confirms that whenever different prompts lead the model to cite the same feature, they point to the exact same evidence in the database.

B Extended Experimental Protocols

B.1 Baseline Architecture and Audio Optionality

We compare MIDI-PHOR against a MIDICAPS-style hybrid baseline, which renders MIDI to audio to extract timbral and genre features. We reproduced this experimental setup to align with prior frameworks and to quantify the impact of synthesis-induced variance on grounding.

A key distinction in our implementation is the reduction of **dependency friction**. The original audio-based extraction requires heavy external libraries (e.g., C++-backed audio processors) that are often difficult to configure. In contrast, MIDI-PHOR functions as a self-contained NLP pipeline. While we treat audio rendering as optional to ensure invariant reasoning, our architecture remains flexible: it supports audio conversion modules if multimodal fusion is required, offering a superset of the baseline's capabilities without enforcing its toolchain overhead.

B.2 Computational Profile and Amortization

We benchmark the end-to-end computational cost of both systems on a standard CPU setup. MIDI-PHOR (symbolic-only) averages 2.40 s/min ($p_{50} = 2.22$), whereas the baseline averages 3.53 s/min.

The primary difference lies in the **synthesis bottleneck**. MIDI→WAV rendering dominates the baseline pipeline, accounting for approximately 77% of total runtime (2.72 s out of 3.53 s) and introducing significant disk usage. Furthermore, the two costs serve different ends:

- **One-off vs. Reusable:** The baseline primarily incurs preprocessing costs to produce disposable features for captioning.
- **Amortized Extraction:** MIDI-PHOR materializes reusable, queryable tables in `DuckDB`. Once computed, these artifacts support multiple downstream tasks (retrieval, QA, analysis) without re-processing the source MIDI.

B.3 Measuring Reliability via Jaccard Invariance

Given the scarcity of purely symbolic ground truth, we sought a rigorous metric to evaluate the stability of the system's reasoning. We employ Jaccard similarity (Intersection over Union) to measure the consistency of detected section boundaries under controlled perturbations, such as transposition and tempo scaling.

In this context, Jaccard similarity serves as a proxy for **"hallucination resistance."** If the system's structural understanding (e.g., identifying a Chorus) shifts significantly due to a simple key change, the model is effectively hallucinating new musical facts. MIDI-PHOR maintains a robust 0.929 pass rate, demonstrating that it anchors its claims in invariant symbolic relationships rather than fragile audio artifacts.

C Feature Extraction Details

This appendix details the extraction logic for the three views used in MIDI-PHOR. A summary of all extracted features is provided in Table 6.

C.1 Symbolic Feature Logic

Tempo-Invariant Density. Standard note density (notes per second) is sensitive to tempo changes. To ensure our features describe the *composition* rather than the *performance speed*, we calculate

Table 5: Compute profile (measured). We compare MIDI-PHOR (symbolic-only) against a MIDICAPS-style reproduction (which renders MIDI $\rightarrow$ WAV audio $\rightarrow$ MTG taggers + Essentia chords). Runtime is reported as seconds per minute of music across songs.

Stage	Runtime (s/min)			Disk	Notes
	Mean	P50	P95		
MIDI-PHOR					
`symbolic.run`	2.241	2.075	3.768	Low	Parsing + bar grid + symbolic features
`merge_for_song`	0.005	0.005	0.010	Low	Section merge via symbolic novelty
`graph_ext.run`	0.155	0.135	0.277	Low	Interaction graph w/ evidence spans
Total	**2.402**	**2.218**	**4.059**	**Low**	End-to-end (no audio rendering)
MIDICAPS-style					
Audio Render	2.716	2.645	3.357	High	Synthesis dominates runtime
MIDI parse (meta)	0.054	0.051	0.095	Low	Tempo/key/instrument parsing
MTG taggers	0.659	0.640	0.803	Low	Mood/genre from audio embeddings
Chords (Essentia)	0.020	0.019	0.023	Low	Chord extraction from audio
Total	**3.529**	**3.485**	**4.351**	**High**	Audio-driven pipeline

`polyphony_max` and `density_note` using absolute MIDI ticks.

Key Regions. While global key estimation is useful for high-level metadata, it fails to capture modulations (e.g., a bridge in the relative minor). We employ the Krumhansl-Schmuckler key-finding algorithm over a sliding 4-bar window to populate the `key_regions` table. This allows specific sections to be tagged with their local tonal center while preserving a stable global key for Roman Numeral analysis.

C.2 Time-Series Proxies (Symbolic Signals)

To avoid the computational cost and synthesis artifacts of audio rendering, we derive continuous signals directly from symbolic tokens.

- **Symbolic Flux:** We approximate spectral flux by combining velocity variance (ΔV) and pitch-class distribution shifts (ΔC) between consecutive bars. Sudden spikes in this signal robustly indicate section boundaries.
- **Groove Entropy:** To measure rhythmic complexity without audio, we project onsets onto a quantized 16th-note grid and compute the Shannon entropy of the resulting distribution. High entropy correlates with syncopated or "busy" rhythms, while low entropy indicates straight quantization.
- **Cadence Strength:** We calculate a heuristic "resolution score" that peaks when a Dominant (V) harmony resolves to a Tonic (I) on a strong metric downbeat.

C.3 Graph Interaction Logic

Interaction edges are inferred through rule-based heuristics applied to the track activity curves. Figure 2 shows an arrangement summary to visualize a view of sections and roles.

- **Rhythmic Lock:** Defined as a high Intersection-over-Union (IoU > 0.6) of onset locations between a Bass track and a Drum track over a 4-bar window.
- **Call and Response:** Detected by calculating the rolling correlation coefficient between the amplitude envelopes of two melodic tracks. A strong negative correlation ($r < -0.5$) implies an alternating "call and response" pattern.
- **Support:** A directed edge established when a "Pad" role's sustained duration covers $> 80\%$ of a "Melody" role's active timeline.

D Schema and Prompt Examples

We contrast the sparse input context available to the comparison method (MidiCaps) versus the rich structural context provided by MIDI-PHOR in Figure 3. To isolate the contribution of the input representation (static tags vs. temporal graph)).

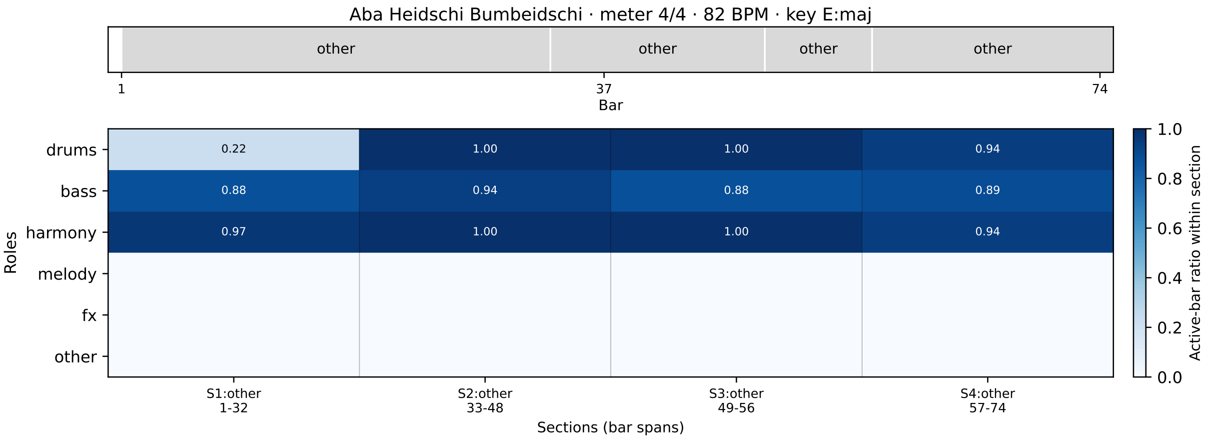

Figure 2: Arrangement view for *Aba Heidschi Bumbeidschi*. Top: merged section timeline over bars. Bottom: section$\times$role activity heatmap (active-bar ratio per role), summarizing how drums/bass/harmony/melody enter and persist across sections. This visualization is derived purely from MIDI notes and the merged section boundaries and serves as an interpretable companion to the full interaction graph.

Table 6: Comprehensive list of features extracted by MIDI-PHOR across all three views.

Category	**Feature Name**	**Description**
Symbolic	`density_note`	Count of note onsets per bar.
	`density_rhythm`	Ratio of occupied 16th-note grid positions.
	`polyphony_max`	Max simultaneous voices (computed in ticks).
	`pitch_range`	Semitone distance between lowest/highest pitch.
	`key_local`	Windowed key estimate (4-bar sliding window).
	`harmony_roman`	Roman numeral relative to local key.
	`motif_ngram`	Top-5 recurring interval-duration sequences.
Time-Series	`flux_sym`	Proxy for spectral change (ΔVelocity + ΔChroma).
	`groove_entropy`	Shannon entropy of quantized onset grid.
	`role_activity`	Normalized density curves per instrument role.
	`cadence_strength`	Heuristic score for V$\rightarrow$I resolution on downbeats.
	`novelty_bar`	Self-similarity peak detection for boundaries.
Graph	`RHYTHMIC_LOCK`	High onset IoU between Bass and Drums.
	`CALL_RESPONSE`	Anti-correlated activity envelopes between tracks.
	`SUPPORT`	Sustained Pad coverage of Melodic lines.
	`DOUBLES`	High pitch/onset identity ($> 90\%$) between tracks.

Table 7: Side-by-side qualitative comparison for *Aba Heidschi Bumbeidschi*. We contrast the generated free-form caption with the intermediate structural representation to demonstrate explicit slot filling and prompt grounding.

MIDI-PHOR (Ours)	**MIDICAPS Baseline**
Generated Free-form Caption	
A laid-back groove anchors this track at a relaxed 82 BPM, with drums and layered textures creating a warm, full sound. The harmonic movement cycles between a bright major tonality and a darker, flattened seventh chord that adds subtle tension. The interplay between the main rhythm and punchy accents keeps the energy grounded.	A serene and uplifting composition in E major that blends orchestral and electronic elements beautifully. The gentle harp and lush string ensemble create a warm, melodic foundation, while ethereal pads and synth strings add a modern, meditative quality.
Intermediate Structure	
4/4 major key composition at 82 BPM in E major with I-bVII7#2 progression. Average polyphony of 11.6 voices, density 9.92, and syncopation 0.24. Weak backbeat (0.19). Orchestrated with harp, synth strings, steel guitar, string ensemble, and acoustic bass.	Section in E major at 82 BPM, 4/4 meter. Chord palette: B, Eb, Gb, E, D. Instrumentation: Orchestral Harp, Synth Strings 1, Acoustic Guitar (steel), String Ensemble 1, Acoustic Bass.
Grounded Claims & Evidence	**Detected Tags**
• `meter = 4/4` • `tempo_bpm = 82` • `key = E:maj` • **`progression = I-bVII7#2`** • `chord_summ = [[“Eb”, “Gb”], 8]` • **`polyphony = 11.61`** • **`density = 9.92`**	• `tempo_bpm = 82` • `meter = 4/4` • `key = E:maj` • `chord_palette = [“B”...]` • `instruments = [“Harp”...]`

MIDICAPS Comparison (Tag-based)

System Prompt

I will give you paired data of tags extracted from songs and the original song captions. The task is to generate the caption from the tags.

Input Representation (Sourced from Claude 3.5 Opus Dataset)

- `Genre:` [['alternative', 'electronic'], [0.24, 0.24]]
- `Mood:` [['meditative', 'dark', 'relaxing'], [0.11, 0.11, 0.10]]
- `Instruments:` [Piano, Electric Bass, Drums]
- `Time Signature:` 4/4 `Tempo:` [125, Fast]
- `Key:` A major
- `Chords:` [['A', 'E7', 'A7', 'D'], 3]

Generated Caption (gpt-4o-mini)

A piano plays a lead melody with chords in the background while being accompanied by a bass guitar and string ensemble. The song evokes a sense of zen, darkness and has a meditative feeling to it. The time signature is 4/4 and the tempo is fast. It's in the key of A major and the main chord progression is A, E7, A7, D.

MIDI-PHOR Ours (Graph-based)

System Prompt

Write a concise, human-friendly music caption. Use 2 sentences for short; 3–4 for medium. Avoid jargon; describe feel, groove, and harmonic motion.

Input Context (Structured Analysis – Abridged)

- **Summary:** Length: $\sim$188s (94 bars) | Tempo: 120 BPM
- **Form & Texture:**
  - M1 [1-8]: density low; active: Vocal, ElecGtr... | roles comp:2, bass:1
  - M2 [9-16]: density high; active: Vocal, ElecGtr...
  - M11 [81-94]: density low; active: Vocal, Rhythm&SE
- **Harmony & Relationships:**
  - M1: Fourth B$\times$180 $\rightarrow$ C#m$\times$90
  - DOUBLES: Vocal $\rightarrow$ SteelGtr [6–94]
  - RHYTHMIC_LOCK: Bass $\rightarrow$ Rhythm&SE [6–94]

Generated Caption (gpt-4o-mini)

This energetic pop track features a steady 120 BPM groove, driven by strong backbeats and a rich texture of bass, steel guitar, and rhythmic accents. The harmonic motion shifts between minor and major chords, creating an engaging tension that invites movement. With varying dynamics across its sections, the song maintains a vibrant energy that keeps listeners captivated.

Figure 3: Vertical comparison of prompt contexts. The MIDICAPS comparison (top) uses the dataset output from the Claude 3.5 Opus generation. MIDI-PHOR (bottom) ingests a serialized graph including section-level texture, harmonic rhythm, and instrument interaction. Both captions were generated using `gpt-4o-mini` to strictly compare the input representations.

Read Between the Tracks: Exploring LLM-driven Intent-based Music Recommendations

Anna Hausberger
Johannes Kepler University
anna.hausberger@jku.at

Petra Jósár
Johannes Kepler University

Markus Schedl
Johannes Kepler University
Linz Institute of Technology
markus.schedl@jku.at

Abstract

This paper evaluates the effectiveness of large language models (LLMs) on the task of context-aware music recommendation, specifically focusing on the alignment of music tracks with a listening intent, in addition to user preferences. We present a preliminary investigation in which five LLMs (variants of *LLama*, *Qwen*, and *Mistral*) are tasked with ranking a candidate set of tracks containing both ground-truth items (associated with specific user-intent pairs) and distractor items (containing user-relevant, intent-relevant, or non-user and non-intent relevant items). Our results show that LLMs rank intent-user-relevant items higher than the distractors, with *Llama-3.1-8B-Instruct* having the best performance (NDCG of $0.32_{0.20}$ vs. $0.20_{0.15}$). We further investigate whether performance differs when mentioning the listening intent explicitly in the prompt vs. implicitly given solely music preferences. Surprisingly, the LLMs achieved the best performance through an implicit indication of intent, versus explicitly adding it to the prompt, with *Mistral-7B-Instruct-v0.3* performing the best (NDCG of $0.37_{0.22}$ vs. $0.29_{0.18}$).

1 Introduction

Listening to music is a common part of everyday life, and for most listeners today this experience is shaped by large-scale streaming platforms. Within these mediated listening contexts, music serves purposes that extend beyond mere entertainment. Prior psychological work by Schäfer et al. (2013) identified 129 non-redundant functions of music listening, organized along three underlying dimensions: self-awareness, social relatedness, and arousal and mood regulation. These functions can be further classified into higher-level clusters that reflect a user's *listening intent*, such as relaxation or reflection (Hausberger et al., 2025).

Exploring the listening intent of a user raises multiple challenges, e.g., a listener might not be aware of how they use music or how music serves them in a specific listening session. Also, the availability of labeled data is scarce. Music listening interactions are often available without any further contextual information on the listening intent of the user. Detecting and identifying music tracks that are used for the same listening intent by the user is still an open problem. This work explores, in a preliminary study, if LLMs can detect and recommend music tracks that have been labeled for being listening to with the same listening intent. Therefore, we put forward two research questions:

- **RQ1:** To what extent can LLMs select and recommend music tracks for a specific listening intent and user from a constrained candidate set?
- **RQ2:** How does providing information about the user's music preferences and listening intent in the prompt affect the performance of the LLMs in the recommendation task?

The generated data and analysis code are released on GitHub.[1]

2 Related Work

The work at hand connects to two stands of research: context-aware and LLM-based recommender systems. As for the former, various approaches have been proposed to incorporate contextual information into music recommender systems (Adomavicius and Tuzhilin, 2010; Lozano Murciego et al., 2021). Methods vary with respect to observability, availability, and acquisition of context data as well as system design (Lozano Murciego et al., 2021).

An important category of contextual information is the user's listening intent. It has been studied in conversational music recommendation from several perspectives, including the specificity of user

[1] https://github.com/hcai-mms/intent-aware-llms

Proceedings of the 4th Workshop on NLP for Music and Audio (NLP4MusA 2026), pages 44–50
March 28, 2026 ©2026 Association for Computational Linguistics

requests (Zhang et al., 2025), user interaction patterns within dialogue-based systems (Jannach et al., 2021), and the use of user queries to refine or filter recommendation results (Doh et al., 2024).

Recently, large language models (LLMs) have emerged as a promising paradigm for recommender systems (Jósár, 2025; Wang et al., 2025; Lin et al., 2025; Epure et al., 2025). Owing to their strong representation and reasoning capabilities, LLMs can be employed to enhance user modeling, extract high-level semantic features, and perform scoring, ranking, or recommendation in zero- or few-shot settings, either via ranking-based approaches (Hou et al., 2024) or generative methods based on token prediction (Doh et al., 2024). Moreover, LLM-based interfaces enable users to articulate their preferences, needs, and listening intents in a more natural and expressive manner (Yun and Lim, 2025), while simultaneously increasing user agency and control over the recommendation process (Friedman et al., 2023).

In this preliminary study, we extend these lines of research by *examining the capacity of LLMs to align users' music listening histories with their underlying listening intents* for the purpose of recommending additional, intent-consistent tracks. Specifically, we evaluate whether LLMs can identify and group music tracks associated with the same listening intent, based on explicitly or implicitly expressed intent cues provided in the prompts.

3 Methodology

In this section, we describe the setup of the experiment: what data was used (Section 3.1), how the listening intent was assigned (Section 3.2), how the prompts were formed, and the text was generated (Section 3.3), and finally, how we evaluate the generated recommendations (Section 3.4).

3.1 Datasets

For our study, we required information regarding both the tracks a user listened to and their corresponding listening intents. Listening intents have been previously studied (Hausberger et al., 2025) using Spotify's Million Playlist Dataset (Chen et al., 2018). This dataset consists of 1,000,000 playlists with title and constituting track names, and covering a total of 2,262,292 unique music tracks (described by title, artist, and Spotify URI). In addition, the LFM-2b Last.fm dataset (Schedl et al., 2022) provides rich information on users and their listening sessions. We created a subset of LFM-2b containing only songs that also occur in the Million Playlist Dataset. This subset contains $119,969$ unique users, $451,728$ unique music tracks, and $361,487,213$ listening events. We additionally dropped the top and bottom 25% of users based on their count of interacted songs, resulting in a minimum of 91 and a maximum of 858 unique tracks per user. The final dataset consists of $59,906$ users.

3.2 Listening Intent Assignment

An approach to assign listening intents to tracks in the Million Playlist Dataset has previously been proposed in (Hausberger et al., 2025). The authors use a text-similarity approach to map the playlist titles to 32 psychological listening intents (e.g., "Calming" or "Support"). We extend this approach by computing the mean textual similarity of the playlist titles to each listening intent name provided in (Hausberger et al., 2025) using cosine similarity over the embeddings created by the "google/embeddinggemma-300m" encoder model (Vera et al., 2025). For each track t in the LFM-2b subset that we created, we subsequently compute the list of playlists in which t occurs. For each track, the mean similarity vector to the listening intents over all occurring playlists is calculated. We further standardized the similarity scores using z-score scaling of each listening intent over all songs to zero mean and unit standard deviation. Z-score standardization makes similarity scores comparable across intents by removing intent-specific scale and variance biases, so that selecting the maximum reflects the strongest relative evidence for an intent rather than artifacts of differing score distributions. To each song, the listening intent with the highest score is assigned.

For each intent, we create a list of users that have listened at least 10 distinct songs with that intent assigned. On average, a user is assigned to 9.0 intents. To further reduce the set of users for computational costs, we randomly sample users equally from each list for each listening intent. In total, this results in $10,670$ unique users and on average 396.4 unique users per listening intent.

Ultimately, each track t is assigned 2 binary labels depending on user and listening intent: i) *User-relevance* (t_u), i.e., if the user u has interacted with the track t, and ii) *Intent-relevance* (t_i), i.e., if the track t is assigned to a specific listening intent i.

For simplicity, we then denote as U the set of all user-relevant tracks per user ($\bigcup t_u$) and $Intent$

describes the set of all intent-relevant tracks per intent ($\bigcup t_i$); also illustrated in Fig. 1.

For each pair of user u and intent i in the final dataset, we then create four sets of tracks without overlaps: i) $U \cap Intent$: all tracks that are user- and intent-relevant, ii) $U \setminus Intent$: user-relevant but not intent-relevant tracks, iii) $Intent \setminus U$: intent-relevant but not user-relevant tracks, iv) $\neg(U \cup Intent)$: tracks that are neither user- nor intent-relevant.

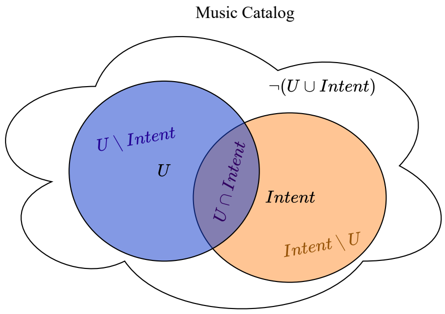

Figure 1: Sets of music track items per user.

3.3 Prompt Generation for Recommendation

The LLM under investigation was given one or two of the following information: i) 3 tracks that a user had listened to for a certain listening intent (further denoted as *music preference*), and ii) the name of the listening intent. Three variations of prompts were formed, in which the LLM got only the intent name (*explicit*), only the music preference of the user in an intent (*implicit*), or *both*. These information variations are further referred to *prompt types*. Then, it received a list of 40 candidate songs consisting of 10 randomly sampled songs from each set of tracks in the format "artist - track". The 40 songs are randomly shuffled for each prompt variation. The LLM is tasked to recommend the tracks sorted based on relevancy to the user and listening intent, and to return 10 tracks sorted in descending order. All prompt types and variations with examples can be found in the Appendix A.

For generating the recommendations, we use 5 different models ("Qwen2-7B-Instruct" (Team et al., 2024), "Mistral-7B-Instruct-v0.3" (Jiang et al., 2023), and models from the Llama-3 family (Dubey et al., 2024) ("Llama-3.2-1B-Instruct", "Llama-3.2-3B-Instruct", "Llama-3.1-8B-Instruct)), each with 3 different prompt types (explicit, implicit, and both) and 3 variations of each prompt. We set the parameters for the generation process to the following values for all LLMs: ($do\_sample = True$, $temperature = 1.0$, $top\_p = 0.9$, $repetition\_penalty = 1.0$, $num\_beams = 1$, $p = 0.9$).

The LLMs were run on four NVIDIA GeForce RTX 2080 Ti GPUs. For a batch size of 12, the average inference time per batch was highest for "Llama-3.1-8B-Instruct" (15.43 s), followed by "Mistral-7B-Instruct-v0.3" (14.91 s), "Qwen2-7B-Instruct" (12.28 s), "Llama-3.2-3B-Instruct" (10.09 s), and "Llama-3.2-1B-Instruct" (4.56 s).

3.4 Evaluation

We filter the set of recommendations for a given user to only include songs that have been given to the LLM as options to select from, and report the percentage of hallucinated tracks per model. Since the outputs may contain repeated tracks, we remove duplicates from predictions by retaining the first occurrence of each track, and finally consider the first 10 recommended tracks. We do not check if a LLM successfully recommends exact 10 music tracks. We evaluate the relevance of recommendations provided by the LLM through the Normalized Discounted Cumulative Gain metric computed on the top 10 recommended items (NDCG@10) over each of the four sets of tracks (see Section 3.2) defining the relevant items.

4 Results & Discussion

In this section, we analyze the results in two different ways. First, we look at the different sets of relevant items discussed before and evaluate how well the models ranked those individual sets (Section 4.1). Second, we examine to which extent explicitly mentioning the listening intent in the prompt influences the LLM's performance in recommending and ranking relevant versus non-relevant items (Section 4.2).

The LLMs produced on average 7.0% recommendations that did not match with any song provided in the candidate set, with the model "Llama-3.2-3B-Instruct" producing on average the most hallucinated recommendations (15.6%), followed by "Llama-3.2-1B-Instruct" (10.3 %), "Mistral-7B-Instruct-v0.3" (6.0%), "Llama-3.1-8B-Instruct" (2.0%), and the model "Qwen2-7B-Instruct" (1.0 %). 1.2 % of recommendation lists included only songs that did not match any item in the candidate list.

4.1 Ranking of Relevant Items (RQ1)

Fig. 2 shows the NDCG@10 scores of each LLM under investigation, computed using each set of tracks as relevant ones (see Section 3.2). As a baseline, we include results of a popularity-based recommender (Pop), which recommends the top k items of the candidate list with the highest count of interactions over all users, sorted, and a random recommender, which recommends the first 10 items of a random shuffled list of items. While the popularity based approach prioritizes generally popular and thus often user-relevant items, it largely fails to capture intent relevance. In contrast, all LLM-based methods consistently rank items that are both user- and intent-relevant higher than unrelated candidates, indicating their ability to model relationships between listening intent, previously consumed tracks, and candidate items. Larger models were able to prioritize user-relevant tracks higher than smaller ones, getting a better performance than random. This trend suggests that increased model capacity enhances the ability to leverage contextual and semantic information for intent-aware recommendation.

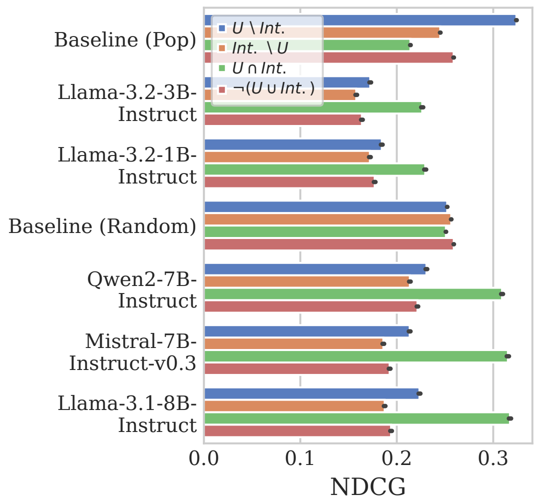

Figure 2: NDCG@10 for the baseline and 5 LLMs, for different user- and intent-relevant track sets. Error bars indicate 95% confidence intervals (CI).

4.2 Effect of Intent and Music Preference Inclusion in Prompt (RQ2)

Furthermore, we examine how the results vary with specific information provided in the prompt (explicit and/or implicit). In Fig. 3, the NDCG scores of the user- and intent-relevant items are shown when grouping by prompt type. It is seen that specifying the intent through intent name generally lowers the performance in ranking intent- and user-relevant items in comparison to only including the music tracks that have been listened to in the same listening intent. This performance gap is particularly noteworthy for larger models. It may seem counterintuitive, but it could indicate that some latent confound is picked up by the (larger) LLMs from the given music preferences alone, rather than by explicitly specifying the intent. This observation gives rise to studying other ways of classifying intent than through a limited set of playlist titles.

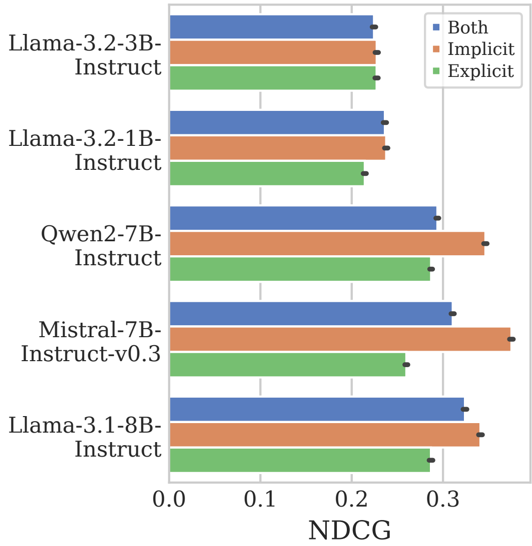

Figure 3: NDCG@10 scores computed on user- and intent-relevant item set ($U \cap Intent$), grouped by the prompt type. Error bars indicate 95% confidence intervals (CI).

5 Conclusion & Future Work

This work has explored whether LLMs are able to identify and rank relevant music tracks for specific users and listening intents. We found that LLMs are able to identify relevant items, and that implicitly indicating an intent alongside tracks the user has interacted with worked best for LLMs to rank intent- and user-relevant tracks. For future work, we will investigate other ways of identifying, encoding, and leveraging listening intent with LLMs to enhance intent-aware music recommendation systems.

6 Limitations

This work is limited by the small number of tested LLMs. We are aware that larger language models probably have a more advanced performance. Further, we acknowledge that text generation could vary depending on the prompt and run. Finally, the evaluation employs a strict string-matching heuristic for track identification, which may underestimate model performance by failing to account for minor formatting variations in the generated output.

Acknowledgments

This research was funded in whole or in part by the Austrian Science Fund (FWF): Cluster of Excellence *Bilateral Artificial Intelligence* (https://doi.org/10.55776/COE12), the doc.funds.connect project *Human-Centered Artificial Intelligence* (https://doi.org/10.55776/DFH23), and the PI project *Intent-aware Music Recommender Systems* (https://doi.org/10.55776/P36413).

References

Gediminas Adomavicius and Alexander Tuzhilin. 2010. Context-aware recommender systems. In *Recommender systems handbook*, pages 217–253. Springer.

Ching-Wei Chen, Paul Lamere, Markus Schedl, and Hamed Zamani. 2018. Recsys challenge 2018: Automatic music playlist continuation. In *Proceedings of the 12th ACM Conference on Recommender Systems*, pages 527–528.

SeungHeon Doh, Keunwoo Choi, Daeyong Kwon, Taesu Kim, and Juhan Nam. 2024. Music discovery dialogue generation using human intent analysis and large language models. *arXiv preprint arXiv:2411.07439*.

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

Elena V Epure, Yashar Deldjoo, Bruno Sguerra, Markus Schedl, and Manuel Moussallam. 2025. Music recommendation with large language models: Challenges, opportunities, and evaluation. *arXiv preprint arXiv:2511.16478*.

Luke Friedman, Sameer Ahuja, David Allen, Zhenning Tan, Hakim Sidahmed, Changbo Long, Jun Xie, Gabriel Schubiner, Ajay Patel, Harsh Lara, et al. 2023. Leveraging large language models in conversational recommender systems. *arXiv preprint arXiv:2305.07961*.

Anna Hausberger, Hannah Strauss, and Markus Schedl. 2025. Exim: Exploring intent of music listening for retrieving user-generated playlists. In *Proceedings of the 2025 ACM SIGIR Conference on Human Information Interaction and Retrieval*, pages 348–357.

Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. 2024. Large language models are zero-shot rankers for recommender systems. In *European Conference on Information Retrieval*, pages 364–381. Springer.

Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, and Li Chen. 2021. A survey on conversational recommender systems. *ACM Computing Surveys (CSUR)*, 54(5):1–36.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Petra Jósár. 2025. Large language models for intent-aware music recommendation/author petra jósár, bsc.

Jianghao Lin, Xinyi Dai, Yunjia Xi, Weiwen Liu, Bo Chen, Hao Zhang, Yong Liu, Chuhan Wu, Xiangyang Li, Chenxu Zhu, et al. 2025. How can recommender systems benefit from large language models: A survey. *ACM Transactions on Information Systems*, 43(2):1–47.

Alvaro Lozano Murciego, Diego M Jiménez-Bravo, Adrian Valera Roman, Juan F De Paz Santana, and María N Moreno-García. 2021. Context-aware recommender systems in the music domain: A systematic literature review. *Electronics*, 10(13):1555.

Thomas Schäfer, Peter Sedlmeier, Christine Städtler, and David Huron. 2013. The psychological functions of music listening. *Frontiers in psychology*, 4:511.

Markus Schedl, Stefan Brandl, Oleg Lesota, Emilia Parada-Cabaleiro, David Penz, and Navid Rekabsaz. 2022. Lfm-2b: A dataset of enriched music listening events for recommender systems research and fairness analysis. In *Proceedings of the 2022 Conference on Human Information Interaction and Retrieval*, pages 337–341.

Qwen Team et al. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2(3).

Henrique Schechter Vera, Sahil Dua, Biao Zhang, Daniel Salz, Ryan Mullins, Sindhu Raghuram Panyam, Sara Smoot, Iftekhar Naim, Joe Zou, Feiyang Chen, et al. 2025. Embeddinggemma: Powerful and lightweight text representations. *arXiv preprint arXiv:2509.20354*.

Shijia Wang, Tianpei Ouyang, Yunfan Zhou, Qiang Xiao, Yintao Ren, Yifei Pan, Fangjian Li, and Chuanjiang Luo. 2025. Enhanced emotion-aware music recommendation via large language models. In *Proceedings of the 31st ACM SIGKDD Conference on*

Knowledge Discovery and Data Mining V. 2, pages 4986–4994.

Sojeong Yun and Youn-kyung Lim. 2025. User experience with llm-powered conversational recommendation systems: A case of music recommendation. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–15.

Junjie Zhang, Ruobing Xie, Yupeng Hou, Xin Zhao, Leyu Lin, and Ji-Rong Wen. 2025. Recommendation as instruction following: A large language model empowered recommendation approach. *ACM Transactions on Information Systems*, 43(5):1–37.

A Prompts

This section presents the prompts used for text generation. In each prompt, the placeholder INTENT_SONGS is replaced with the three songs previously listened to by the user, CANDIDATE_LIST is replaced with the set of candidate songs, and INTENT_NAME is replaced with the corresponding intent label. Depending on the prompt type (explicit, implicit, or both), the prompt provided to the LLM includes the three listened songs (implicit information), the intent name (explicit information), or both.

A.1 System Prompt

"System: You are a music recommender. You recommend music tracks for a user and an intent from a set of music tracks. You get a user listening history, the intent and return a list of music tracks recommended for the intent and user. Recommend the best suitable track first. Return 10 recommendations. Return the track in the format: Recommendations: "artist - track" split with ",". You get the songs to sort in the format "ID": "artist - track", return 10 tracks recommended for the intent and user. Analyze the music tracks given based on how relevant they are for an intent and user and recommend 10 tracks best, most relevant tracks first. Recommend only songs before 2020. Recommend best suited songs first."

A.2 User Prompts

1. 'Intent: "INTENT_NAME"
   User Listened Songs in Intent: "INTENT_SONGS"
   Song Options: "CANDIDATE_LIST"'

2. 'User Listening History: "INTENT_SONGS"
   Intent: "INTENT_NAME"
   Song to rank: "CANDIDATE_LIST"'

3. 'Intent: "INTENT_NAME"
   Songs listened in Intent by User: "INTENT_SONGS"
   Song Options to rank: "CANDIDATE_LIST"'

A.3 Examples

A.3.1 Explicit

Intent: Fitness Motivation
Song Options: 0: Nonpoint - In The Air Tonight, 1: Omar Souleyman - Shift Al Mani, 2: Cavalera Conspiracy - Bloodbrawl, 3: Japandroids - The Nights of Wine and Roses, 4: Danzig - I Don't Mind The Pain, 5: Nine Inch Nails - The Day the World Went Away, 6: Misfits - Horror Hotel, 7: Social Distortion - Highway 101, 8: Airbourne - Raise the Flag, 9: Black Rebel Motorcycle Club - Spread Your Love, 10: Diablo - Icaros, 11: Social Distortion - Making Believe, 12: Winds of Plague - Legions, 13: A Day to Remember - We Got This, 14: We Came As Romans - To Plant a Seed, 15: Alexisonfire - We Are The End, 16: Kiss - Tough Love, 17: Sepultura - Subtraction, 18: Schiller - Solitude, 19: Kiss - 100,000 Years, 20: The Real McKenzies - Drink the Way I Do, 21: Ministry - Jesus Built My Hotrod, 22: Dropkick Murphys - It's A Long Way to The Top (If You Wanna Rock N Roll), 23: Jennifer Lopez - Papi, 24: Mad Sin - U.F.O., 25: Black Light Burns - Animal, 26: Cavalera Conspiracy - Black Ark, 27: Macklemore - Levitate (feat. Otieno Terry), 28: Rancid - 1998, 29: Twenty One Pilots - Heathens, 30: Tiger Army - As The Cold Rain Falls, 31: Linkin Park - Crawling, 32: The Clancy Brothers - Johnson's Motor Car, 33: Big Bad Voodoo Daddy - Minnie The Moocher, 34: The Gaslight Anthem - Handwritten, 35: Trapt - Headstrong, 36: Danzig - She Rides, 37: Sepultura - Desperate Cry, 38: Ramones - Substitute, 39: Karmin - Brokenhearted"

A.3.2 Implicit

'User Listened Songs in Intent: "Jenny Lewis - Pretty Bird", "Feist - Honey Honey", "The Format - Pick Me Up"
Song Options: 0: Love Unlimited Orchestra - Satin Soul, 1: Kent - Sverige, 2: Alkaline Trio - Nose Over Tail, 3: Zeigeist - Humanitarianism, 4: Old Crow Medicine Show - Cocaine Habit, 5: The Submarines - Maybe, 6: Teebs - Clapstick, 7: of Montreal - ErosÉntropic Tundra, 8: Matt Pond PA - I Want To See The Bright Lights Tonight, 9: Foxes - Devil Side, 10: Bhagavan Das - Hanuman Chalisa, 11: The Danse Society - Somewhere, 12: Slum Village - Players, 13: Jeff Beck - Over The Rainbow, 14: Neko Case - Fever, 15: Someone Still Loves You Boris Yeltsin - Yr Broom, 16: Andhim - Bermudachords, 17: Jenny Lewis - The Next Messiah, 18: Grouper - Headache, 19: of Montreal - Chrissy Kiss the Corpse, 20: Air - Missing the Light of the Day, 21: The Knife - Pass This On, 22: Starlight Mints - Brass Digger, 23: Mates of State - What I Could Stand For, 24: Robert Ellis - Perfect Strangers, 25: Grouper - Being Her Shadow, 26: Kesha - Finding You, 27: The Little Ones - Morning Tide, 28: Jenny Lewis - Bad Manś World, 29: Mates of State - Parachutes (Funeral Song), 30: Bright Eyes - Southern State, 31: Boys Noize - Drummer, 32: Paul Kalkbrenner - Torted, 33: Ted Leo and the Pharmacists - The Gold Finch and the Red Oak Tree, 34: Someone Still Loves You Boris Yeltsin - Anna Lee, 35: Regina Spektor - Human of the Year, 36: The Go! Team - Bottle Rocket, 37: Elvis Presley - Heartbreak Hotel, 38: Manowar - Animals, 39: Ani DiFranco - You Had Time'

A.3.3 Both

Intent: Nostalgia
User Listened Songs in Intent: "Live - I Alone", "Delerium - Paris", "When In Rome - The Promise"
Song Options: 0: Kim Carnes - Bette Davis Eyes, 1: Kut Klose - Surrender, 2: Moby - Hymn, 3: Britney Spears - E-Mail My Heart, 4: Counting Crows - Rain King, 5: Razed in Black - Am I 2 Blame?, 6: Joni Mitchell - Free Man In Paris, 7: Nice Smooth - Funky For You, 8: Marc Streitenfeld - Earth, 9: Lynyrd Skynyrd - Free Bird, 10: Michael Jackson - I just cant́ stop loving, 11: Orgy - Blue Monday, 12: Joss Stone - Sideway Shuffle, 13: Marvin Gaye - Distant Lover, 14: Depeche Mode - Personal Jesus (Telephone Stomp Mix), 15: Van Halen - Top of the World, 16: Beastie Boys - Triple Trouble, 17: Marc Streitenfeld - Friend From The Past, 18: Marconi Union - Breathing With Assistance, 19: The Temptations - Some Enchanted Evening, 20: Kelly Clarkson - Anytime, 21: Beastie Boys - Pass the Mic, 22: Bob Dylan - Subterranean Homesick Blues, 23: Jens Gad - Cape Blanc, 24: Seabird - Cottonmouth (Jargon), 25: Beastie Boys - Hey Ladies, 26: Lifehouse - First Time, 27: The Beautiful South - Everybodyś Talkin; 28: A Tribe Called Quest - Oh My God, 29: Chingy - Right Thurr, 30: Judas Priest - On the Run, 31: Enigma - Mea Culpa - Part II (Catholic Version), 32: The Cars - Drive, 33: Kelly Clarkson - A Moment Like This, 34: Beastie Boys - Shake Your Rump, 35: Audiomachine - House of Cards, 36: Razed in Black - Blush, 37: The Cranberries - How, 38: Marc Streitenfeld - Hyper Sleep, 39: Louie Culture - No Gal"

Learning When to Personalize: LLM Based Playlist Generation via Query Taxonomy and Classification

Fedor Buzaev[1,2], **Ivan Sukharev**[1], **Rinat Mullahmetov**[1,3], **Roman Bogachev**[1], **Ilya Sedunov**[1], **Oleg Pavlovich**[1], **Daria Pugacheva**[2,4]

[1]Zvuk, [2]HSE University, [3]Innopolis University, [4]Domain-specific NLP Group

Correspondence: {fa.buzaev,dpugacheva}@hse.ru, ivan@sukharev.me

Abstract

Playlist generation based on textual queries using large language models (LLMs) is becoming an important interaction paradigm for music streaming platforms. User queries span a wide spectrum from highly personalized intent to essentially catalog-style requests. Existing systems typically rely on non-personalized retrieval/ranking or apply a fixed level of preference conditioning to every query, which can overfit catalog queries to a single user or underpersonalize explicitly listener-dependent requests. We present an industrial-scale LLM-based playlist generation system with dynamic personalization that adapts the personalization strength to the query type. We define a query taxonomy, train a query-type classifier on 5,000 manually labeled queries, and use its predicted probability to modulate the mixture of LLM-based semantic scoring and personalized evaluation. In a blind user study with pairwise comparisons and ELO aggregation, this approach consistently outperforms both non-personalized and fixed-personalization baselines.

1 Introduction

Large language models (LLMs) are progressively being employed to power natural language interfaces for recommender systems (Wu et al., 2024). In the context of music streaming, users are able to formulate playlist requests using free-form text, such as 'upbeat indie rock for a Friday night with friends' or 'calm instrumental music for deep work'. LLMs effectively parse such rich queries and retrieve items that are semantically aligned with the articulated user intent (Palumbo et al., 2025). At the same time, music recommendation is inherently personal. Different users may submit identical queries yet expect vastly different playlists, influenced by their long-term preferences, listening histories, and current context. Consequently, a practical LLM-based playlist generator must integrate two key signals: semantic relevance to the query and personalization to the user. Modern platforms typically utilize collaborative or sequence-based recommender systems to capture these individual preferences (Zeng et al., 2024; Deldjoo et al., 2024). However, not all queries should be personalized to the same degree. Some queries are explicitly dependent on the preferences of an individual user (e.g., 'my favorite pop hits of the 2010s'), others are primarily constrained by external limitations and specific descriptions (e.g., 'best disco hits of the 1980s'). Applying a strong collaborative approach to the latter may degrade the perceived quality by replacing canonical tracks with niche songs that appeal only to a particular user. Conversely, weak personalization for taste-dependent queries may result in the creation of generic playlists that feel detached from the listening history of the user.

In this work, we argue that the degree of personalization should be adapted to the query type. We analyze real user queries from a large-scale music streaming platform and construct a taxonomy of 19 categories, grouped into queries that warrant strong personalization and queries that should primarily adhere to the textual description. Building on this taxonomy, we train a query-type classifier on 5,000 manually labeled queries sourced from the *Zvuk.com* music streaming platform, and use its predicted probability as a continuous personalization strength.

Our system starts from an LLM-based embedder that retrieves candidate tracks by semantic similarity to the text query. A collaborative recommender scores the same candidates based on the listener's historical behavior. For each query, the classifier predicts the probability that this query benefits from strong personalization. Then the semantic and collaborative scores are fused.

We evaluate several model variants in a blind

Proceedings of the 4th Workshop on NLP for Music and Audio (NLP4MusA 2026), pages 51–57

user study with pairwise comparisons between systems. Participants see two anonymized playlists for the same query and indicate which one better matches their expectations, or that they are equivalent. We aggregate outcomes with an ELO-style rating system and complement this with non-parametric significance tests. The dynamic personalization variant achieves the highest ELO rating and win rate.

Our contributions are:

- A taxonomy of text-to-playlist queries from a commercial music streaming platform, highlighting when strong personalization is desirable or harmful.
- A method for dynamically fusing LLM-based and collaborative scores based on the type of a query.
- A blind user study with pairwise comparisons and ELO aggregation, showing that query-type-aware dynamic personalization improves perceived playlist quality over non-personalized and fixed-personalization baselines.

2 Related Work

Large language models (LLMs) enable text-to-playlist generation by mapping natural language queries to semantically relevant tracks. Prototypes like LLMusic integrate LLMs with Spotify APIs for prompt-based playlist creation without fine-tuning (Preda, 2023). Industrial systems like Text2Tracks extend this with generative retrieval, which generates track IDs directly from a textual query (Palumbo et al., 2025).

Traditional music recommenders fuse semantic relevance with collaborative filtering (Zeng and Umrawal, 2025), but fixed-weight hybrids often over-personalize catalog queries or underfit taste-dependent ones. Sequential models incorporate personalized popularity awareness to balance long-term preferences and novelty in next-track prediction (Abbattista et al., 2024). Dynamic user interest models adapt recommendations over time by modeling evolving preferences, yet lack query-specific modulation (Mao et al., 2024).

3 Methodology

In this section, we formalize the text-to-playlist task, outline the semantic retrieval and query expansion approach based on LLMs, present the collaborative re-ranking component, and introduce the dynamic personalization system guided by the query-type classifier.

3.1 Problem Setting and Notation

We consider a platform with a set of users $\mathcal{U}$ and a catalog of tracks $\mathcal{T}$. A user $u \in \mathcal{U}$ issues a free-form natural language query x. Given (u, x), the system must produce an ordered list of tracks

$$\pi(u, x) = (t_1, t_2, \dots, t_K),$$

where $t_k \in \mathcal{T}$ and K is the playlist length.

The pipeline operates in two main stages:

1. **Semantic candidate generation** with an LLM-based embedder.
2. **Personalized re-ranking** with a collaborative recommender, optionally modulated by a query-type classifier.

We then denote the semantic (Embedder) score by $s_{\text{llm}}(x, t)$ and the collaborative (RecSys) score by $s_{\text{cf}}(u, t)$. Both scores are normalized to $(0, 1]$ via monotonic transformations so that multiplicative fusion is well-defined.

3.2 LLM-Based Semantic Retrieval

The first stage uses an LLM-based text encoder to map the query x to a dense vector representation $\mathbf{v}_{\text{query}}(x) \in \mathbb{R}^d$. Each track $t \in \mathcal{T}$ is associated with one or more textual descriptions (e.g., title, artist names, editorial tags), which are encoded into vectors $\mathbf{v}_t \in \mathbb{R}^d$ using the same or a compatible encoder. The LLM-based text encoder is fine-tuned using the InfoNCE contrastive objective on triplets of anchor, positive, and negative examples to map queries and tracks into a shared semantic embedding space that enables meaning-based retrieval.

We perform nearest-neighbor search in the track embedding index to find the set $\mathcal{C}(x)$ of top-N candidates. We define the semantic relevance score as

$$s_{\text{llm}}(x, t) = \text{sim}(\mathbf{v}_{\text{query}}(x), \mathbf{v}_t), \tag{1}$$

where $\text{sim}(\cdot, \cdot)$ is the cosine similarity (Manning et al., 2008).

This stage ensures that all candidates are thematically connected to the textual description, even for users with sparse histories or for niche queries that do not align well with collaborative signals.

3.3 Collaborative Re-Ranking

The second stage uses a production collaborative recommender to personalize the candidate set $\mathcal{C}(x)$ for user u. The model takes as input the user identifier and track features (including historical interactions such as listens, skips, and likes) and outputs a score $s_{\text{cf}}(u,t) \in (0,1]$, which reflects how well track t matches long-term preferences of the user.

For this purpose, the expanded vector $\mathbf{v}_{\text{exp}}(u,x)$ (see the exact definition in the Appendix A) replaces $\mathbf{v}_{\text{query}}(x)$ in the similarity computation (1), yielding a score that reflects both the textual description and the explicit feedback of the user.

A simple fixed-personalization baseline fuses the two scores using a global exponent $\alpha_{\text{fixed}} \geq 0$:

$$s_{\text{final}}^{\text{fixed}}(u,x,t) = s_{\text{llm}}(x,t) \cdot \left(s_{\text{cf}}(u,t)\right)^{\alpha_{\text{fixed}}}. \quad (2)$$

When $\alpha_{\text{fixed}} = 0$, the ranking is purely semantic. As α_{fixed} increases, the model becomes more personalized and elevates preferred tracks, even if their connection to the query is weak.

However, a single global α_{fixed} cannot capture the diversity of query types. A fixed value may be both insufficient for queries that are highly dependent on personal preferences or excessive for descriptive or catalog-oriented queries. To address this issue, we introduce a query-type classifier and define α as a query-dependent function.

3.4 Dynamic Personalization

Given that queries fall into two types, one strictly personalized and the other corresponding to non-personalized catalog queries, let us introduce a binary variable $y \in \{0,1\}$. Here, $y = 1$ if a query falls into the first type and $y = 0$ if it belongs to the second. Thus, a binary classifier trained on the labeled data returns the probability that a query x is strictly personalized, and the coefficient α is calculated as

$$\alpha(x) = p_\theta(y = 1 \mid x) \in [0,1]. \quad (3)$$

Given $\alpha(x)$, we define the dynamically personalized fusion score as

$$s_{\text{final}}^{\text{dyn}}(u,x,t) = s_{\text{llm}}(x,t) \cdot \left(s_{\text{cf}}(u,t)\right)^{\alpha(x)}. \quad (4)$$

When $\alpha(x) \to 0$, the collaborative factor tends to 1 and the final score is dominated by the LLM component:

$$\lim_{\alpha(x)\to 0} s_{\text{final}}^{\text{dyn}}(u,x,t) = s_{\text{llm}}(x,t). \quad (5)$$

When $\alpha(x)$ is close to 1, the collaborative score retains its full strength and strongly re-orders the semantic candidates. In intermediate cases, the exponent provides a smooth interpolation between semantic and collaborative ranking, driven by the query-type classifier.

4 Taxonomy and Labels

To determine when strong personalization is appropriate, we manually examine user queries and classify them into 19 categories (see the Appendix B). We further differentiate between categories for which the playlist should depend on preferences of the user and those for which it should primarily reflect external constraints. Data were sourced from the *Zvuk.com* music streaming platform. We sampled no more than two queries per user and processed all logs in anonymized form.

Nine categories (e.g., mood, activity and etc.) are treated as *personalized*, while the remaining ten categories (e.g., historical or cultural events, explicit artists and etc) are treated as *non-personalized* (see Table 3). Then, we annotate 5,000 real user queries with this taxonomy. After annotation, the query distribution comprised 42% personalized requests, while 58% did not require personalization.

For dynamic personalization we collapse the taxonomy into a binary label $y \in \{0,1\}$, where $y = 1$ denotes that the query should be strongly personalized (belongs to one of the personalized categories) and $y = 0$ denotes that the query should mostly follow the textual description.

5 Experimental Setup and Results

We evaluate the proposed approach in an internal blind user study (please refer to the Appendix C for details) and compares different model variants.

5.1 Model Variants

We instantiate the general architecture in several concrete model variants, all of which are built on a `Qwen2.5-7B`–based (Qwen, 2024) semantic retriever.

In the first variant, final ranking are performed exclusively by the LLM retriever, such that the ordering is determined solely by $s_{\text{llm}}(x,t)$. The second variant retains the same retriever but uses it to generate a candidate set of size $N = 500$, which is subsequently re-ranked in a collaborative manner using $s_{cf}(u,t)$; the two signals are combined via

Model	#, M	Acc	Macro-F1	ROC-AUC
RuBERT	178	83.1	81.2	88.31
RuRoBERTa	355	84.13	82.2	89.5
SBERT	400	86.2	85.02	91.33
RuModernBERT	150	**87.55**	**86.7**	**92.67**

Table 1: Performance of query-type classifiers on the test set.

the fixed-exponent fusion defined in Eq.(2). The third variant follows the same two-stage candidate generation with $N = 500$ and collaborative scoring, but additionally incorporates a query-type classifier that predicts $\alpha(x)$, and the final ranking is obtained using the query-dependent fusion in Eq.(4).

These variants enable an ablation-style analysis of the contributions of collaborative re-ranking, and query-type-aware dynamic personalization within a unified retrieval framework.

5.2 Query-type classification.

We fine-tune four Russiaan-language transformer encoders as query-type classifiers, namely `RuBERT-base`, `RuRoBERTa-large` (Zmitrovich et al., 2024), `sbert_large_nlu_ru` (Abramov et al., 2024) and `RuModernBERT-base` (Spirin et al., 2025), since Russian is the primary language of user queries on the streaming platform.

Each model takes the tokenized query x as input, encodes it into a contextual representation, and applies a linear classification head with a softmax over the two classes. We split the 5,000 labeled queries into nto training, validation, and test sets in a 70%, 20%, and 10% ratio, respectively, and train with cross-entropy loss. Training details (batch size 200, learning rate $5e$-4, number of epochs $2e3$) follow standard fine-tuning practice.

Table 1 summarizes the performance of the four models on the held-out test set. Due to architectural modifications and the pretraining design, `RuModernBERT-base` is expected to outperform the other models (Spirin et al., 2025). Consequently, we select the `RuModernBERT-base` to compute $\alpha(x)$ for the internal blind user study stage.

5.3 User study results

In total, 20 unique users participated in the study and submitted 254 pairwise judgments.

Each comparison corresponds to a single query and a single model pair. We aggregate the pairwise outcomes using an ELO-style rating system (Olesker-Taylor and Zanetti, 2024) and complement this with non-parametric significance tests (Mann–Whitney) (Groggel, 2000) where appropriate. Table 2 reports the number of wins and the resulting ELO ratings. Among the results, there were 19 cases in which users judged the two playlists as equivalent.

Model	Wins	ELO
Retriever	37	1382.10
Retriever + CR	73	1525.54
Retriever + CR + DP	125	1592.37

Table 2: Number of wins and ELO ratings for the non-personalized baseline (Retriever), the fully personalized setting applied to all queries (Retriever + CR), and the dynamic-personalization approach (Retriever + CR + DP).

The dynamically personalized variant wins 125 comparisons (53.2%) versus 37 wins (15.7%) for the retriever-only solution and 73 wins (31.1%) for the the fully personalized setting. The difference in ELO ratings of more than 143.44 points indicates a substantial advantage of personalization over the non-personalized baseline. Query-type-aware dynamic fusion delivers additional benefits, outperforming both the non-personalized baseline and the always-personalized configuration. Non-parametric tests confirm that the difference in pairwise preferences is statistically significant ($p < 0.05$).

Conclusion

We presented a query-type-aware dynamic personalization mechanism for LLM-based playlist generation. By training a classifier on a manually constructed taxonomy of user queries and using its output probability as an exponent for the collaborative score, the system can smoothly interpolate between purely semantic and strongly personalized rankings. Combined with query expansion from likes and dislikes, this approach improves user preferences in a blind pairwise study compared to non-personalized and fixed-personalization baselines. Future work includes richer multi-label query understanding, user-controllable personalization sliders, and automatic learning of personalization schedules from interaction data.

Discussion and Limitations

Any personalized recommender may reinforce existing preferences and reduce diversity, thereby limiting exposure to novel content. Dynamic personalization partially mitigates this effect by attenuating the contribution of the collaborative component for many non-personalized queries. In designing and training the pipeline, we focused on a single language that is predominant on the streaming platform. However, extending the approach to multilingual settings may improve robustness and reliability (Pletenev et al., 2025).

Ethics

Our study relies on user testing and crowdsourced data annotation conducted by paid contributors through the streaming platform's internal service. Participation in the user testing was strictly voluntary and anonymized. Contributors were compensated at rates above the average level to ensure fair remuneration for their time and effort. This approach reflects our commitment to labor ethics and our respect for the value of human contributions to AI research.

References

Davide Abbattista, Vito Walter Anelli, Tommaso Di Noia, Craig MacDonald, and Aleksandr Vladimirovich Petrov. 2024. Enhancing sequential music recommendation with personalized popularity awareness. In *Proceedings of the 18th ACM Conference on Recommender Systems, RecSys 2024, Bari, Italy, October 14-18, 2024*, pages 1168–1173. ACM.

Aleksandr Abramov, Denis Antykhov, and Ibragim Badertdinov. 2024. Bert large model (uncased) for sentence embeddings in russian language.

Yashar Deldjoo, Markus Schedl, and Peter Knees. 2024. Content-driven music recommendation: Evolution, state of the art, and challenges. *Comput. Sci. Rev.*, 51:100618.

David J. Groggel. 2000. Practical nonparametric statistics. *Technometrics*, 42(3):317–318.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to information retrieval*. Cambridge University Press.

Chunyan Mao, Shuaishuai Huang, Mingxiu Sui, Haowei Yang, and Xueshe Wang. 2024. Analysis and design of a personalized recommendation system based on a dynamic user interest model. *CoRR*, abs/2410.09923.

Sam Olesker-Taylor and Luca Zanetti. 2024. An analysis of elo rating systems via markov chains. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Enrico Palumbo, Gustavo Penha, Andreas Damianou, José Luis Redondo García, Timothy Christopher Heath, Alice Wang, Hugues Bouchard, and Mounia Lalmas. 2025. Text2tracks: Prompt-based music recommendation via generative retrieval. *CoRR*, abs/2503.24193.

Sergey Pletenev, Maria Marina, Nikolay Ivanov, Daria Galimzianova, Nikita Krayko, Mikhail Salnikov, Vasily Konovalov, Alexander Panchenko, and Viktor Moskvoretskii. 2025. Will it still be true tomorrow? multilingual evergreen question classification to improve trustworthy QA. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 8614–8631, Suzhou, China. Association for Computational Linguistics.

David Preda. 2023. Llmusic: An llm-based playlist generator connected to spotify.

Team Qwen. 2024. Qwen2.5: A party of foundation models.

Egor Spirin, Boris Malashenko, and Sokolov Andrey. 2025. Rumodernbert: Modernized bert for russian.

Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, Hui Xiong, and Enhong Chen. 2024. A survey on large language models for recommendation. *World Wide Web (WWW)*, 27(5):60.

Terence Zeng and Abhishek K. Umrawal. 2025. Content filtering methods for music recommendation: A review. *CoRR*, abs/2507.02282.

Zhaodian Zeng, Yining Wang, Yanbin Zhao, and Wenxuan Shi. 2024. A survey of music recommendation systems. In *Proceedings of the 5th International Conference on Computer Information and Big Data Applications*, CIBDA '24, page 507–519, New York, NY, USA. Association for Computing Machinery.

Dmitry Zmitrovich, Aleksandr Abramov, Andrey Kalmykov, Vitaly Kadulin, Maria Tikhonova, Ekaterina Taktasheva, Danil Astafurov, Mark Baushenko, Artem Snegirev, Tatiana Shavrina, Sergei S. Markov, Vladislav Mikhailov, and Alena Fenogenova. 2024. A family of pretrained transformer language models for russian. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 507–524. ELRA and ICCL.

A Query Expansion from Likes and Dislikes

To better align the semantic search with long-term preferences of the user, we expand the query representation using embeddings of tracks that the user has explicitly liked or disliked.

Let $\mathbf{v}_{\text{pos}}(u)$ be the average embedding of tracks that user u has liked, and $\mathbf{v}_{\text{neg}}(u)$ the average embedding of tracks that the user has disliked or hidden. We introduce three non-negative scalar weights:

- q — weight of the original query embedding;
- p — weight of positive history (likes);
- n — weight of negative history (dislikes).

These weights are controlled in the range of $[0, 1]$.

We form an expanded query vector

$$\mathbf{v}_{\text{exp}}(u, x) = \frac{q\,\mathbf{v}_{\text{query}}(x) + p\,\mathbf{v}_{\text{pos}}(u) - n\,\mathbf{v}_{\text{neg}}(u)}{q + p - n}, \quad (6)$$

assuming $q + p - n > 0$. Here the negative sign in front of $n \cdot \mathbf{v}_{\text{neg}}(u)$ explicitly pushes the expanded query away from regions of the space corresponding to disliked content.

B Queries Taxonomy

Table 3 presents the query taxonomy used in our analysis, separating user requests into personalized and non-personalized types. Personalized queries capture situations where the appropriate response is expected to depend on the individual listener (e.g., mood, activities, life events, travel or romantic context, and explicit taste-dependent requests). Non-personalized queries cover requests that can be satisfied largely independently of user identity, including cultural references, known artists/genres, trends and charts, "in the style of" prompts, children's content, media soundtracks, concrete musical attributes, references to third parties or public venues, and noisy or malformed inputs. Each category is illustrated with a representative example query.

C User Study Interface

The study uses a web-based interface integrated into the music streaming platform. Each participant provides their platform identifier (u) and enters a free-form text query x in their native language. For each evaluation trial, the system samples a pair of models from a predefined comparison set and generates two anonymized playlists:

- Playlist A: output of model M_a for (u, x).
- Playlist B: output of model M_b for (u, x).

The interface does not reveal which underlying model produced which playlist. Participants see only the track lists and can interact with them (e.g., by inspecting track titles and artists). For each pair they are asked to choose one of three options:

1. Playlist A is better.
2. Playlist B is better.
3. Both playlists are equivalent.

Each response is recorded as a pairwise outcome for the corresponding models (M_a, M_b). A single user can submit multiple queries; queries and model pairs are randomized across trials.

Type	Category	Example
Personalized	Emotional state	*for a melancholic mood*
	Sports and physical activity	*for running*
	Home leisure	*for a cozy evening at home*
	Work and study	*for concentration at work*
	Life situation	*for exam preparation*
	Travel	*for a night drive through the city*
	Fantasy situation	*to imagine yourself as Marilyn Monroe*
	Romantic context	*for a romantic mood*
	Explicit taste-dependent requests	*punk rock for an evening mosh that I will like*
Non-personalized	Cultural or historical context	*Victory Day*
	Existing genres, styles, artists, or tracks	*the most popular song of Louis Armstrong*
	Existing trends and charts	*TikTok trends 2025*
	"In the style of X" requests	*songs in the style of the Beatles*
	Children's content	*for a children's morning party*
	Soundtracks and media	*songs from the cartoon N*
	Concrete musical characteristics	*guitar in djent style*
	Queries referencing other people or groups	*for a party with my father-in-law*
	Public places	*for a coffee shop*
	Noise or random strings	*fjdhk.!!668!*

Table 3: Categories of personalized and non-personalized queries with examples.

HumMusQA: A Human-written Music Understanding QA Benchmark Dataset

Benno Weck
Universitat Pompeu Fabra
benno.weck01@estudiant.upf.edu

Pablo Puentes
Universitat Autònoma de Barcelona

Andrea Poltronieri
Universitat Pompeu Fabra

Satyajeet Prabhu
Universitat Pompeu Fabra

Dmitry Bogdanov
Universitat Pompeu Fabra
dmitry.bogdanov@upf.edu

Abstract

The evaluation of music understanding in Large Audio-Language Models (LALMs) requires a rigorously defined benchmark that truly tests whether models can perceive and interpret music, a standard that current data methodologies frequently fail to meet. This paper introduces a meticulously structured approach to music evaluation, proposing a new dataset of 320 hand-written questions curated and validated by experts with musical training, arguing that such focused, manual curation is superior for probing complex audio comprehension. To demonstrate the use of the dataset, we benchmark six state-of-the-art LALMs and additionally test their robustness to uni-modal shortcuts.

1 Introduction

The rapid progress of Large Language Models (LLMs) has catalysed the development of Large Audio-Language Models (LALMs), such as Audio Flamingo (Ghosh et al., 2025b; Goel et al., 2025) and Qwen-Audio (Chu et al., 2023). These multi-modal systems integrate an audio encoder with a large language model, allowing them to process audio input and generate textual responses conditioned on what they hear. This sets them apart from earlier self-supervised audio representation models (Schneider et al., 2019; Alonso-Jiménez et al., 2025), which learn acoustic features without language generation, and from uni-modal text-only approaches. To achieve comprehensive audio understanding, LALMs must go beyond speech recognition and encompass all audio domains (Iyer, 2025), with music being one of the most challenging – requiring a model to listen to an audio clip, process a text-based question, and produce an answer grounded in auditory perception.

Music understanding presents persistent challenges for LALMs due to music's dynamic, layered, and information-dense nature. This includes both perceptual and analytical capabilities, recognizing musical features like instrumentation, key, and structure, as well as cultural and contextual knowledge about genre and mood. Evaluating music understanding in LALMs is particularly difficult because musical concepts are often complex and open-ended, making conventional lexical metrics like BLEU (Papineni et al., 2002) inadequate for assessing the diverse language responses.

To establish a comprehensive and objective measure of auditory intelligence, the field has coalesced around Question Answering (QA) frameworks (e.g., Weck et al., 2024; Sakshi et al., 2025; Wang et al., 2025; Yang et al., 2024), which structure evaluation through multiple-choice classification, constrained reasoning, or open-ended questions that are better suited to assessing complex music capabilities. Despite the growth of Music-QA datasets, the field has historically prioritized scale over quality. Early benchmarks like MusicQA (Liu et al., 2024) and MusicInstruct (Deng et al., 2024) were constructed by using LLMs to automatically augment existing captions from datasets like MusicCaps (Agostinelli et al., 2023) or tags from MagnaTagATune (Law et al., 2009). This reliance on automated sourcing often compromises evaluation integrity: text-only LLMs lacking audio perception can achieve high accuracy by exploiting language priors and "world knowledge" embedded in the question text alone. This "perception gap" suggests that many current benchmarks primarily measure a model's reasoning ability rather than genuine audio perception (Weck et al., 2024; Zang et al., 2025).

Automatically deriving questions from short, surface-level captions or tags inherently limits question depth and scope, preventing the formulation of challenging, multi-hop inquiries necessary for testing expert-level musical understanding.

Recent work has begun shifting toward expert-annotated benchmarks that demand more than surface-level recognition. A significant milestone

Proceedings of the 4th Workshop on NLP for Music and Audio (NLP4MusA 2026), pages 58–67
March 28, 2026 ©2026 Association for Computational Linguistics

in this direction is MMAU-Pro (Kumar et al., 2025), a comprehensive benchmark that utilizes expert-written and validated question-answer pairs to evaluate holistic auditory intelligence. Notably, music forms a substantial portion of this dataset, with 1,618 questions dedicated to musical understanding. While MMAU-Pro sets a high standard for expert curation, it highlights a remaining trade-off in benchmark construction regarding data provenance. To avoid data leakage from existing training sets, MMAU-Pro sources its audio "from the wild" and through various online repositories. This approach, while robust against leakage, often relies on disparate sources with potentially variable audio quality and metadata reliability. Furthermore, other expert-curated efforts like MusicTheoryBench (MTB) (Yuan et al., 2024) offer high expert-driven quality but remain limited to the symbolic domain (ABC notation), failing to test direct perceptual grounding in audio.

We argue that evaluating the full depth of music understanding requires a specialized, perceptually rigorous approach that combines expert curation with high-fidelity source material. We introduce a novel evaluation dataset containing 320 handwritten questions, curated and validated by experts with advanced musical training. Manual authorship enables broader topic coverage and more sophisticated multi-layered reasoning than automated generation can achieve. Crucially, our design minimizes language shortcuts: questions require genuine musical perception and analysis across structural, harmonic, perceptual, and cultural dimensions. All audio materials are sourced from Creative Commons-licensed recordings, ensuring the benchmark can be openly distributed.

2 Methodology

The goal of this study is to create a human-authored benchmark for evaluating large audio-language models on music understanding tasks. The benchmark consists of 320 expert-written questions paired with freely licensed musical recordings, designed to assess model performance across diverse aspects of musical knowledge and reasoning.

All audio tracks were sourced from Jamendo[1], a platform hosting Creative Commons-licensed music. We selected 108 tracks spanning multiple genres, instrumentation types, and production styles to ensure comprehensive coverage of musical characteristics. Each question refers to a specific excerpt from a track, ranging from 30 to 90 seconds in duration, with the exact time window determined by the question authors based on the musical content being assessed. The use of openly licensed material ensures the benchmark can be freely distributed and reproduced without legal restrictions (Bogdanov et al., 2019; Manco et al., 2023), addressing a significant barrier to reproducibility in music AI research (Batlle-Roca et al., 2025). The complete dataset, including questions, and metadata statistics is made publicly available under a Creative Commons licenses on Zenodo.[2]

2.1 Question Design

Question design was informed by two established music education standards: the Associated Board of the Royal Schools of Music (ABRSM)[3] syllabi and the General Certificate of Secondary Education (GCSE) music curriculum[4]. We additionally informed our approach by existing music understanding benchmarks, such as MuChoMusic (Weck et al., 2024) and MMAU (Sakshi et al., 2025). Drawing on these sources, we designed questions spanning a broad spectrum of music understanding: from foundational perceptual tasks and world-knowledge aspects (e.g., cultural context, lyrical content) accessible to music beginners, to sophisticated analytical reasoning requiring music theory knowledge.

Three music theory experts (mean professional experience in music = 15 years; all holding advanced academic qualifications in music theory) each authored approximately one-third of the questions. Authors were instructed to design questions that: i) reflect authentic educational objectives from ABRSM/GCSE curricula; ii) span diverse cognitive demands including perceptual identification, analytical reasoning, and interpretive assessment; iii) require careful listening and musical knowledge to answer correctly; and iv) admit exactly one clear, unambiguous correct answer. All questions were designed in multiple-choice format with four options (one correct, three distractors) to facilitate automated evaluation. Questions range from those accessible to casual listeners (e.g., *"What emotion is mainly conveyed in this song?"* with options: joy, sadness, anger, disgust) to those requiring music theory knowledge (e.g., *"What intervals create dissonance in the background guitar?"* with options:

[1] https://www.jamendo.com/

[2] https://doi.org/10.5281/zenodo.18462524

[3] https://www.abrsm.org/

[4] https://www.gov.uk/education

4ths, fifths, octaves, unison). More examples are provided in Appendix 6.

While LLMs could potentially generate questions of this kind, they lack the ability to ground questions in genuine audio perception. Expert authorship ensures that the proposed questions reflect authentic musical reasoning, requiring engagement with both the audio and the textual content.

Experts played a dual role in this process: not only generating the questions, but also validating each other's work through iterative peer review. Each expert was asked to blindly answer questions authored by the others without prior knowledge of the intended correct answer, ensuring that questions could be consistently and unambiguously resolved. During this blind review, annotators flagged disagreements regarding the most likely answer option and provided written comments identifying potential issues. Common failure modes included questions deemed too subjective (e.g., relying on personal interpretation rather than objective musical features), distractors that were not equally plausible (e.g., one option being trivially eliminable), and incorrect or imprecise labeling of answer options. Authors then revised their questions based on this feedback, addressing flagged issues and clarifying ambiguities. This iterative cycle continued until no further comments or disagreements were raised, at which point the question was considered validated and included in the final benchmark.

2.2 Question Labelling

To enable systematic analysis of model performance across different aspects of musical understanding, we classified questions according to two dimensions: *musical category* and *level of musical knowledge required*.

Each question was assigned one or more categories from an adapted version of the MuChoMusic (Weck et al., 2024) taxonomy, comprising 13 musical dimensions: Melody, Harmony, Metre and Rhythm, Instrumentation, Musical Texture, Sound Texture, Performance, Structure, Mood and Expression, Lyrics, Genre and Style, Historical and Cultural Context, and Functional Context. Each question received one primary category reflecting its main analytical focus and zero or more secondary categories if addressing multiple aspects.

We additionally classified questions according to the level of musical knowledge required to answer them correctly based on listening alone. Questions were assigned to one of three levels: Low (answerable by casual listeners with no formal training), Medium (requiring some musical training or active listening experience), or High (requiring formal music education or specialized knowledge).

Both classifications were performed using GPT-5 (`gpt-5-2025-08-07`) (OpenAI, 2025a) with structured prompts providing category definitions and detailed examples of each dimension. We used LLM-assisted annotation to ensure consistency across all 320 questions and 13 categories, reducing subjective interpretation of category boundaries. Two domain experts independently validated all automated assignments and disagreements were resolved through discussion.

The final benchmark comprises questions distributed across all 13 musical categories, with the most frequent being Instrumentation (19.7%), Harmony (11.3%), and Melody (10.6%), while Musical Texture, Structure, and Lyrics each represent less than 3.5% of questions. Regarding difficulty, 44.4% of questions were classified as *low*, 38.4% as *medium*, and 17.2% as *high*.

3 Experiments

To demonstrate the utility of our benchmark, we test several state-of-the-art LALMs, selecting models that span different design paradigms: general-purpose multi-modal LLMs (gemini-2.5-flash, gpt-audio), audio-specialized LALMs (audio-flamingo-3, qwen2.5-omni-7b, audsemthinker), and one model explicitly designed for music understanding (music-flamingo). Since all models have been designed and fine-tuned on question-answering tasks, the QA format should be familiar, enabling performance to serve as a direct measure of music understanding rather than task format comprehension. Furthermore, following prior work (Zang et al., 2025), we assess whether questions can be answered using text alone, without access to audio.

3.1 Evaluation strategy

Models are evaluated by prompting them with audio snippets and corresponding multiple-choice questions. Previous studies have shown that both large audio-language models and text-only large language models are highly sensitive to the ordering of multiple-choice options, with answer position alone inducing substantial performance variance and unstable model rankings (Lin et al., 2025; Zheng et al., 2024; Pezeshkpour and Hruschka, 2024). To address this issue, and following estab-

Model	Accuracy with SD (%)				Consistency (%)			
	All	Low	Med.	High	All	Low	Med.	High
qwen2.5-omni-7b (Xu et al., 2025)	**64.3** ± 2.9	**73.9** ± 3.1	**58.1** ± 2.5	53.2 ± 3.9	66.6	70.5	61.3	67.9
music-flamingo (Ghosh et al., 2025a)	58.5 ± 1.6	60.9 ± 2.1	**58.1** ± 1.8	53.2 ± 4.1	39.1	41.7	38.7	30.2
audio-flamingo-3 (Goel et al., 2025)	58.1 ± 3.2	62.7 ± 3.3	54.7 ± 3.8	**54.1** ± 1.5	55.0	55.4	56.5	50.9
gpt-audio (OpenAI, 2025b)	57.7 ± 1.5	65.3 ± 3.9	55.1 ± 1.9	43.6 ± 1.8	45.6	45.3	43.5	47.2
gemini-2.5-flash (Google, 2025)	55.9 ± 1.2	64.4 ± 1.5	48.4 ± 3.6	50.5 ± 2.7	35.0	40.3	31.5	28.3
audsemthinker (Wijngaard et al., 2025)	54.8 ± 2.2	61.3 ± 1.5	49.0 ± 3.7	51.4 ± 3.5	42.5	44.6	41.1	39.6
gemini-2.5-flash *(text-only)*	42.7 ± 0.9	44.4 ± 0.5	37.6 ± 1.8	50.0 ± 2.0	48.8	54.0	46.0	43.4

Table 1: Accuracy & consistency scores for systems across all benchmark questions, overall and by difficulty level. Accuracy is averaged over four runs with randomized answer orderings (standard deviation shown). Consistency measures the percentage of questions where the model produced identical answers across all four runs, indicating robustness to answer position bias.

lished practices in recent audio and music understanding benchmarks (Weck et al., 2024; Lin et al., 2025), we evaluate each model under multiple randomized answer orderings. Specifically, for each question, we perform four independent evaluation runs, where the answer options are randomly shuffled in each run. Final performance metrics are computed by averaging results across these runs.

The output provided by the model is automatically parsed by an LLM (`gemini-2.5-flash`) prompted to match the response with the given options. This ensures consistent analysis of model outputs of different lengths, particularly when responses are long. From this matching, we calculate a simple accuracy scores which are presented in Table 1.

From the results, we observe that Qwen2.5-Omni-7B (Xu et al., 2025) attains the best performance overall, as well as within the *low* and *medium* difficulty categories. Notably, this model demonstrates remarkable consistency, producing the same answers across multiple runs despite variations in answer option ordering. In contrast, most models exhibit strong sensitivity to answer shuffling, with performance varying substantially across runs – suggesting vulnerability to prompt formulation rather than robust understanding.

Performance decreases consistently with increasing difficulty levels, validating our difficulty labeling scheme. Figure 1 reveals distinct patterns across question categories: questions requiring general musical knowledge (e.g., genre and style, mood and expression, functional context) achieve higher scores, while music theory-grounded questions (e.g., harmony, melody, performance) yield substantially lower performance. This suggests that models are stronger at cultural and contextual reasoning than at formal analytical tasks requiring music theory expertise.

3.2 Testing robustness to uni-modal shortcuts

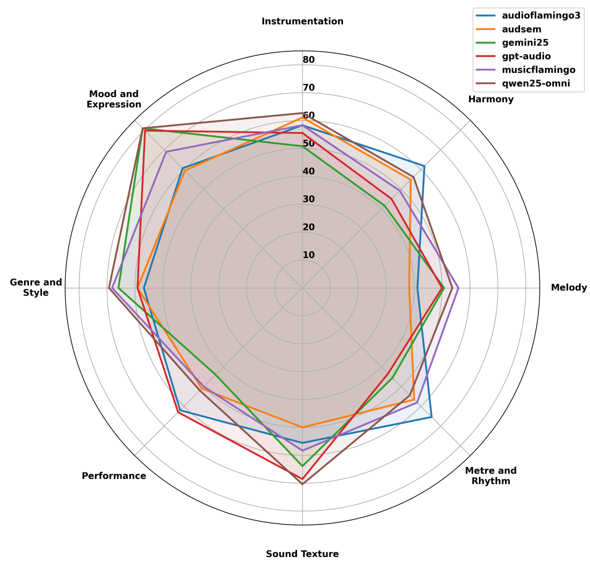

Figure 1: Accuracy across different categories in the benchmark. Categories accounting for $<= 5\%$ of questions in the dataset are excluded from the chart. Specifically, these are *Historical and Cultural Context*, *Musical Texture*, *Lyrics*, *Structure*, and *Functional Context*.

Additional experiments are conducted to test robustness by replacing real audio inputs with fake audio, following prior work (Zang et al., 2025; Weck et al., 2024; Kumar et al., 2025). The generally accepted hypothesis is that, without correct audio context, question-answering accuracy should not exceed random chance (25%). Therefore, we evaluate whether average model performance drops under these fake-audio conditions (see Table 2).

We also evaluate gemini-2.5-flash in a text-only setting, prompting it to respond using theoretical

	Accuracy with SD (%)		
Model	Real	Noise	Silence
qwen2.5-omni-7b	64.3 ± 2.9	46.2 ± 1.3	47.0 ± 2.0
musicflamingo	58.5 ± 1.6	42.9 ± 3.6	42.2 ± 3.0
gemini-2.5-flash	55.9 ± 1.2	37.8 ± 1.5	39.8 ± 1.6

Table 2: Accuracy scores with true audio compared to fake audio (Gaussian noise or silence).

knowledge. We further test using prompt variations and prompt optimisation strategies such as DSPy (Khattab et al., 2024) without significant differences in results. Results show that, while model performance is indeed degraded in comparison to the true audio setting, it exceeds pure random selection. This is noteworthy since in our question writing we strive for equally likely options.

Analysis of responses reveals that the model exploits cues in the questions and employs its knowledge of common practices and methods like statistical likelihood of an option and elimination of outliers. For example, in the question, "The guitar is a typical accompaniment from a specific country. Which country is it?", the model might deduce Brazil (Bossa Nova) by assuming "typical" means "most globally recognized/distinctive," using theoretical and historical knowledge, and statistical prevalence of specific guitar styles. Other options, Argentina, Venezuela and Cuba, while having "typical" guitar styles, are seen as less singularly iconic. In another example, "During the chorus, we can hear a very popular type of synthesizer sound. Can you guess its name?", the phrase "very popular type of synthesizer sound" strongly cues "supersaw" over the other options (square, triangle, sine) because it's a named, popular sound, while the others are building blocks. The options themselves create a categorical distinction. These findings reveal that human-written questions are vulnerable to being answered due to factors such as weak question phrasing and distractors. We leave a more detailed analysis of what makes human-authored questions solvable from text alone to future work. In particular, follow-up studies could analyse why items labelled as high difficulty are still answered correctly by a text-only model in roughly 50% of cases. This could inform how annotation procedures might be refined to help annotators design genuinely high-difficulty questions (typically involving very specific conceptual content) such that all answer choices are roughly equally plausible based on the text alone.

4 Conclusion

We presented a music question-answering benchmark with 320 expert-authored questions and evaluated six state-of-the-art LALMs. Results show that while models achieve moderate performance overall, they exhibit systematic weaknesses on music theory questions requiring analytical reasoning, with performance decreasing with difficulty. The dataset is situated within the broader ecosystem of existing benchmark datasets and is intended to be used either in conjunction with or as a complement to them. Furthermore, it enables reproducible evaluation under clear licensing constraints, since both the audio materials and the expert-authored question text are provided under Creative Commons licenses.

5 Acknowledgements

We thank Armando Cedillo Martínez for the contributions and discussions. This work is supported by "IA y Música: Cátedra en Inteligencia Artificial y Música" (TSI-100929-2023-1) funded by the Secretaría de Estado de Digitalización e Inteligencia Artificial and the European Union-Next Generation EU, under the program Cátedras ENIA.

References

Andrea Agostinelli, Timo I. Denk, Zalán 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*.

Pablo Alonso-Jiménez, Pedro Ramoneda, Recep Oguz Araz, Andrea Poltronieri, and Dmitry Bogdanov. 2025. OMAR-RQ: open music audio representation model trained with multi-feature masked token prediction. In *Proceedings of the 33rd ACM International Conference on Multimedia, MM 2025, Dublin, Ireland, October 27-31, 2025*, pages 13640–13643. ACM.

Roser Batlle-Roca, Laura Ibáñez-Martínez, Xavier Serra, Emilia Gómez, and Martín Rocamora. 2025. MusGO: A Community-Driven Framework for Assessing Openness in Music-Generative AI. In *Proceedings of the 26th International Society for Music Information Retrieval Conference, ISMIR 2025, Daejeon, South Korea, September 21-25, 2025*, pages 727–738.

Dmitry Bogdanov, Minz Won, Philip Tovstogan, Alastair Porter, and Xavier Serra. 2019. The MTG-Jamendo Dataset for Automatic Music Tagging. In

Machine Learning for Music Discovery Workshop, International Conference on Machine Learning (ICML 2019), Long Beach, CA, United States.

Yunfei Chu, Jin Xu, Xiaohuan Zhou, Qian Yang, Shiliang Zhang, Zhijie Yan, Chang Zhou, and Jingren Zhou. 2023. Qwen-Audio: Advancing Universal Audio Understanding via Unified Large-Scale Audio-Language Models. *arXiv preprint*. ArXiv:2311.07919 [cs, eess].

Zihao Deng, Yinghao Ma, Yudong Liu, Rongchen Guo, Ge Zhang, Wenhu Chen, Wenhao Huang, and Emmanouil Benetos. 2024. MusiLingo: Bridging Music and Text with Pre-trained Language Models for Music Captioning and Query Response. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3643–3655, Mexico City, Mexico. Association for Computational Linguistics.

Sreyan Ghosh, Arushi Goel, Lasha Koroshinadze, Sang-gil Lee, Zhifeng Kong, Joao Felipe Santos, Ramani Duraiswami, Dinesh Manocha, Wei Ping, Mohammad Shoeybi, and Bryan Catanzaro. 2025a. Music Flamingo: Scaling Music Understanding in Audio Language Models. *arXiv preprint*. Version Number: 1.

Sreyan Ghosh, Zhifeng Kong, Sonal Kumar, S Sakshi, Jaehyeon Kim, Wei Ping, Rafael Valle, Dinesh Manocha, and Bryan Catanzaro. 2025b. Audio Flamingo 2: An Audio-Language Model with Long-Audio Understanding and Expert Reasoning Abilities. In *Forty-second International Conference on Machine Learning*.

Arushi Goel, Sreyan Ghosh, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang-gil Lee, Chao-Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael Valle, and Bryan Catanzaro. 2025. Audio Flamingo 3: Advancing Audio Intelligence with Fully Open Large Audio Language Models. *arXiv preprint arXiv*.

Gemini Team Google. 2025. Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities. *arXiv preprint*. Version Number: 5.

Laya Balaji Iyer. 2025. Analyzing Audio Understanding in Multimodal LLMs: A Benchmark Grounded in Assistive and Industrial Use Cases.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. 2024. DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines. In *The Twelfth International Conference on Learning Representations*.

Sonal Kumar, Simon Sedlácek, Vaibhavi Lokegaonkar, Fernando López, Wenyi Yu, Nishit Anand, Hyeonggon Ryu, Lichang Chen, Maxim Plicka, Miroslav Hlavácek, William Fineas Ellingwood, Sathvik Udupa, Siyuan Hou, Allison Ferner, Sara Barahona, Cecilia Bolaños, Satish Rahi, Laura Herrera-Alarcón, Satvik Dixit, Rupali S. Patil, Soham Deshmukh, Lasha Koroshinadze, Yao Liu, Leibny Paola Garcia Perera, Eleni Zanou, Themos Stafylakis, Joon Son Chung, David Harwath, Chao Zhang, Dinesh Manocha, Alicia Lozano-Diez, Santosh Kesiraju, Sreyan Ghosh, and Ramani Duraiswami. 2025. MMAU-Pro: A Challenging and Comprehensive Benchmark for Holistic Evaluation of Audio General Intelligence. *CoRR*, abs/2508.13992. ArXiv: 2508.13992.

Edith Law, Kris West, Michael I. Mandel, Mert Bay, and J. Stephen Downie. 2009. Evaluation of Algorithms Using Games: The Case of Music Tagging. In *Proceedings of the 10th International Society for Music Information Retrieval Conference, ISMIR 2009*, pages 387–392, Kobe International Conference Center, Kobe, Japan. International Society for Music Information Retrieval.

Yu-Xiang Lin, Chen-An Li, Sheng-Lun Wei, Po-Chun Chen, Hsin-Hsi Chen, and Hung-yi Lee. 2025. Hearing the Order: Investigating Selection Bias in Large Audio-Language Models. *CoRR*, abs/2510.00628. ArXiv: 2510.00628.

Shansong Liu, Atin Sakkeer Hussain, Chenshuo Sun, and Ying Shan. 2024. Music Understanding LLaMA: Advancing Text-to-Music Generation with Question Answering and Captioning. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 286–290, Seoul, Korea, Republic of. IEEE.

Ilaria Manco, Benno Weck, Seungheon Doh, Minz Won, Yixiao Zhang, Dmitry Bogdanov, Yusong Wu, Ke Chen, Philip Tovstogan, Emmanouil Benetos, Elio Quinton, György Fazekas, and Juhan Nam. 2023. The Song Describer Dataset: a Corpus of Audio Captions for Music-and-Language Evaluation. In *Machine Learning for Audio Workshop at NeurIPS 2023*.

OpenAI. 2025a. GPT-5 System Card.

OpenAI. 2025b. gpt-audio Model | OpenAI API.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Pouya Pezeshkpour and Estevam Hruschka. 2024. Large Language Models Sensitivity to The Order of Options in Multiple-Choice Questions. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2006–2017, Mexico City, Mexico. Association for Computational Linguistics.

S. Sakshi, Utkarsh Tyagi, Sonal Kumar, Ashish Seth, Ramaneswaran Selvakumar, Oriol Nieto, Ramani Duraiswami, Sreyan Ghosh, and Dinesh Manocha. 2025.

MMAU: A Massive Multi-Task Audio Understanding and Reasoning Benchmark. In *The Thirteenth International Conference on Learning Representations*.

Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. 2019. wav2vec: Unsupervised pre-training for speech recognition. In *20th Annual Conference of the International Speech Communication Association, Interspeech 2019, Graz, Austria, September 15-19, 2019*, pages 3465–3469. ISCA.

Bin Wang, Xunlong Zou, Geyu Lin, Shuo Sun, Zhuohan Liu, Wenyu Zhang, Zhengyuan Liu, AiTi Aw, and Nancy F. Chen. 2025. AudioBench: A Universal Benchmark for Audio Large Language Models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2025 - Volume 1: Long Papers, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*, pages 4297–4316. Association for Computational Linguistics.

Benno Weck, Ilaria Manco, Emmanouil Benetos, Elio Quinton, George Fazekas, and Dmitry Bogdanov. 2024. MuChoMusic: Evaluating Music Understanding in Multimodal Audio-Language Models. In *Proceedings of the 25th International Society for Music Information Retrieval Conference, ISMIR 2024*, pages 825–833, San Francisco, California, USA and Online.

Gijs Wijngaard, Elia Formisano, Michele Esposito, and Michel Dumontier. 2025. AudSemThinker: Enhancing Audio-Language Models Through Reasoning over Semantics of Sound.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. 2025. Qwen2.5-Omni Technical Report. *arXiv preprint*. Version Number: 1.

Qian Yang, Jin Xu, Wenrui Liu, Yunfei Chu, Ziyue Jiang, Xiaohuan Zhou, Yichong Leng, Yuanjun Lv, Zhou Zhao, Chang Zhou, and Jingren Zhou. 2024. AIR-Bench: Benchmarking Large Audio-Language Models via Generative Comprehension. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1979–1998, 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, Liumeng Xue, Ziyang Ma, Qin Liu, Tianyu Zheng, Yizhi Li, Yinghao Ma, Yiming Liang, Xiaowei Chi, Ruibo Liu, Zili Wang, Chenghua Lin, Qifeng Liu, Tao Jiang, Wenhao Huang, Wenhu Chen, Jie Fu, Emmanouil Benetos, Gus Xia, Roger Dannenberg, Wei Xue, Shiyin Kang, and Yike Guo. 2024. 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.

Yongyi Zang, Sean O'Brien, Taylor Berg-Kirkpatrick, Julian J. McAuley, and Zachary Novack. 2025. Are You Really Listening? Boosting Perceptual Awareness in Music-QA Benchmarks. In *Proceedings of the 26th International Society for Music Information Retrieval Conference, ISMIR 2025, Daejeon, South Korea, September 21-25, 2025*, pages 247–261.

Chujie Zheng, Hao Zhou, Fandong Meng, Jie Zhou, and Minlie Huang. 2024. Large Language Models Are Not Robust Multiple Choice Selectors. In *The Twelfth International Conference on Learning Representations*.

6 Appendix

Category	Difficulty	Question	Answer options (correct first)
Functional Context	Low	Where would you imagine this song being played?	A: In a night club, B: At a children's party, C: During a romantic dinner, D: After exercising
	Medium	Which of these options is most related to this song?	A: bedroom producer, B: racial protests, C: avant-garde music, D: futurism
Genre and Style	Low	Which guitar comping style can we here in the tune?	A: bossa nova, B: manouche guitar, C: flamenco guitar, D: hard bop
	Medium	This song could have been written recently. However, its style sounds like it was written...	A: Before bebop, B: After bebop, C: Before ragtime, D: After jazz fusion
	High	Right before the first verse, there are chords that are reminiscent of a particular jazz style. Do you know which one?	A: Modal jazz, B: Be bop, C: Hard bop, D: Fusion jazz
Harmony	Low	This verse of the tune is written...	A: In a major key, B: In a minor key, C: Neither of the two, D: It is not a tonal tune
	Medium	How would you define the harmonic rhythm of this tune?	A: Steady, B: unstable, C: hemiolic, D: chaotic
	High	What surprises you about the harmony in this song?	A: It has many second dominants, B: It uses chords with tensions, C: It uses polychords, D: Nothing is surprising
Historical and Cultural Context	Low	In what decade could this song be created?	A: 2010, B: 1990, C: 1980, D: 1970
	Medium	Where could the roots of this song be?	A: Andalucia, B: Mexico, C: Cuba, D: Barcelona
Instrumentation	Low	What instrument plays the lead sound in the introduction?	A: Synthesizer, B: Bass guitar, C: Piano, D: Ukulele
	Medium	The guitar is a typical accompaniment from a specific country. Which country is it?	A: Brazil, B: Cuba, C: Argentina, D: Venezuela
	High	Which instrument plays the lead sound in the introduction?	A: Muted trombone, B: Trumpet, C: Tuba, D: Saxophone
Lyrics	Low	What type of language is used in the lyrics of the song?	A: Metaphoric and symbolic, B: Realistic, C: Narrative, D: Essay
Melody	Low	Which of the following best fits the guitar pattern heard at the beginning of this excerpt?	A: riff, B: glissando, C: Alberti, D: walking bass
	Medium	During the introduction, the bass and the lead play their melodies in a...	A: parallel motion, B: contrary motion, C: oblique, D: random
	High	In the verse there is a dissonance that doesn't belong the key and mode of the tune. Do you know which one is it?	A: Ab, B: G, C: F, D: B
Metre and Rhythm	Low	What type of beat do you hear in the drums?	A: rock, B: funk, C: soul, D: jazz
	Medium	During the introduction, the kick is played twice in each bar. On which beats?	A: one and between the second and the third beat, B: two and four, C: one and three, D: two and three
	High	When the drums start playing, we can hear something known as...	A: metric modulation, B: time signature change, C: swing, D: double-time
Mood and Expression	Low	What is the singer pursuing with her tone?	A: Sensuality, B: Power, C: Loudness, D: Detachment
	Medium	What does not contribute to creating this mood?	A: Tremolo, B: Reverb, C: The sound of the guitar, D: A clear and consonant harmony
Musical Texture	Low	What is the intention behind adding new voices in the last section?	A: Increase energy, B: Decrease the energy, C: Maintain the same energy level, D: Introduce new energy
	Medium	What compositional technique do the piano and the drums use in the introduction?	A: call and response (also known as question and answer), B: homophonically, C: There is only a piano in this excerpt., D: counterpoint
Performance	Low	What type of articulation best defines the melody of this song?	A: legato, B: marcato, C: sforzando, D: staccato
	Medium	How does the drummer create anticipation for the final verse?	A: Toms and snare fill, B: Cymbal fill, C: Hi Hat fill, D: Kick drum fill
	High	What extended technique is used by the flutist during the solo?	A: Flutter tonguing, B: Shake, C: Spitting, D: Jet whistle
Sound Texture	Low	What effect is applied to the vocals in this track?	A: pitch correction, B: Bitcrush, C: Ring modulation, D: gated reverb
	Medium	During the chorus, we can hear a very popular type of synthesizer sound. Can you guess its name?	A: supersaw, B: square wave, C: triangle wave, D: sine wave
	High	What would you add to the drums to make them more present?	A: A compressor, B: An equalizer, C: I would duplicate the track, D: I would play it in mono
Structure	Medium	What is the longest section of the tune?	A: Solos, B: Verse, C: Chorus, D: Introduction
	High	Which section of the song follows this harmonic progression: vi, IV, I, V?	A: chorus, B: verse, C: bridge, D: outro

Table 3: Representative questions by category and difficulty

Model	Backbone & Architecture	Training data	Reasoning Mechanism & Variants	Context Length	Max Audio	Capabilities	Openness
audio-flamingo-3	Qwen-2.5-7B with AF-Whisper unified audio encoder	AudioSkills-XL, LongAudio-XL, AF-Think, AF-Chat (∼9.5M samples).	Optional short chain-of-thought (on-demand, not default) • **AF3 (Base)** - Single-turn Inference • **AF3 + Think** - Inference with short CoT prefixes • **AF3-Chat** - multi-turn, multi-audio, voice-to-voice interaction • **AF3 (Long-Audio reasoning)** - long-audio reasoning	8,192 tokens	Up to 10 minutes	automatic speech recognition, audio question answering, audio reasoning, sound event recognition, music understanding, long-audio understanding, multi-turn audio dialogue, voice-to-voice interaction	Fully open (research)
audsemthinker	Qwen2.5-Omni-7B fine-tuned	AUDSEM dataset (∼900k instances) from YouTube.	Explicit structured semantic reasoning over sound • **Think** - <think> <answer> • **Think + Semantic** - <think> <semantic_elements> <answer>	32,768 tokens (inherited from Qwen2.5-Omni)	Variable-length audio segments (>3 seconds, context-limited)	semantic audio reasoning, sound event understanding, audio question answering, audio captioning, audio-based multiple-choice reasoning, creative audio-conditioned text generation	Open
gemini-2.5-flash	Sparse Mixture-of-Experts (MoE) multimodal transformer	Proprietary Data	Dynamic internal reasoning with user-controllable thinking budget	∼1,000,000 tokens	Up to ∼3 hours of audio/video	automatic speech recognition, speech translation, audio question answering, audio summarization, audio-visual reasoning, long-context audio understanding, native audio dialogue, text-to-speech	Closed / Proprietary
gpt-audio-2025-08-28	Not available	Not available	Not available	128,000 tokens	Not available	general audio model; accepts audio input and output;	Closed
music-flamingo-hf	Enhanced AF3 backbone with time-aware embeddings (RoTE)	MF-Skills (4M+) and MF-Think (300k) music datasets.	Explicit music-theory chain-of-thought (optional, structured)	∼24,000 tokens	Up to ∼20 minutes	music captioning, music question answering, music theory reasoning, harmonic analysis, structural music analysis, lyric grounding, cultural music analysis, long-form song understanding	Fully open (research)
qwen2.5-omni-7b	Qwen-2.5-7B Thinker-Talker end-to-end multimodal architecture	Large-scale multimodal pretraining (∼1.2T tokens) from open sources.	Implicit internal reasoning	32,768 tokens	Long streaming audio (context-limited, no fixed minute cap)	automatic speech recognition, speech-to-text translation, audio question answering, audio reasoning, sound classification, music classification, audio-visual reasoning, streaming speech interaction	Open weights

Table 4: Comparison of models used in the study

Summary of model-wise prompts and settings

This section details the salient configuration parameters and representative prompt templates employed for each model evaluated in this study.

1. **audio-flamingo-3**

```
model variant = 'Single-Turn Inference'
prompt = f"{question}\n{answer_str}"
```

2. **audsemthinker**

```
model variant = 'Think + Semantics'
prompt = f"{question}\n{answer_str}"
```

3. **gemini-2.5-flash**

```
prompt = f"""**Task:** You are an expert musicologist with perfect pitch and extensive knowledge
of music theory, instrumentation, and performance techniques. Your goal is to analyze the provided
audio excerpt and answer the multiple-choice question with high precision. If audio is missing,
use theoretical knowledge to deduce the answer.
{question}
{answer_str}
**Final Answer:** Return ONLY the single letter: A, B, C, or D"""
```

4. **gpt-audio**

```
prompt = f"{question}\n{answer_str}"
```

5. **music-flamingo**

```
prompt = f"{question}\n{answer_str}"
```

6. **qwen2.5-omni-7b**

Using a direct question-answer prompt did not yield optimal results. The model returns responses such as:

```
"I'm not sure which direction the low - pass filter is shifting. It could be up or down, or even up - down.
You might need to check the audio more closely or have some technical knowledge about filters to figure it
out. Why are you interested in this low - pass filter?"
```

We use a system prompt based on the audio understanding cookbook provided by the authors.
https://github.com/QwenLM/Qwen2.5-Omni/blob/main/cookbooks/universal_audio_understanding.ipynb
We further guide the model towards selecting one of the provided options using the following prompt template:

```
prompt = f"""You are a music audio understanding model.

Listen carefully to the provided audio clip. Answer the following multiple-choice
question based on what you hear.

Question:
{question}

Options:
{answer_str}

Respond with ONLY the letter of the correct option (A, B, C, or D).
Do not include any explanation or additional text."""
```

Author Index