# Information Extraction of Music Entities in Conversational Music Queries

**Simon Hachmeier and Robert Jäschke**
School of Library and Information Science
Humboldt-Universität zu Berlin
{simon.hachmeier,robert.jaeschke}@hu-berlin.de

## Abstract

The detection of music entities such as songs or performing artists in natural language queries is an important task when designing conversational music recommendation agents. Previous research has observed the applicability of named entity recognition approaches for this task based on pre-trained encoders like BERT. In recent years, large language models (LLMs) have surpassed these encoders in a variety of downstream tasks. In this paper, we validate the use of LLMs for information extraction of music entities in conversational queries by few-shot prompting. We test different numbers of examples and compare two sampling methods to obtain few-shot examples. Our results indicate that LLM performance can achieve state-of-the-art performance in the task.

## 1 Introduction

Detecting music entities such as songs or musical artists in natural language queries is a key component of conversational music agents (Jannach et al., 2021). In such queries, users request music entities they want to listen to in a conversational way as an alternative to traditional text search.

The task of detecting music entities is typically modeled as named entity recognition (NER) which was earlier addressed by probabilistic approaches (Liljeqvist, 2016; Porcaro and Saggion, 2019). More recently, pre-trained encoders demonstrated strong performance in the NER task in the music domain (Xu and Qi, 2022; Epure and Hennequin, 2023).

With the advent of large language models (LLMs) such as GPT-3.5 for text generation tasks, these are increasingly used for NER and the related task of information extraction (IE) (Wang et al., 2023a; Ashok and Lipton, 2023; Zhang et al., 2023). Several studies found that encoder-only models still outperform LLMs (Wang et al., 2023b; Ma et al., 2023; Sun et al., 2023; Zhou et al., 2024). However, LLMs are usually trained on much larger datasets than encoder-only models which theoretically makes these more likely to capture music knowledge to some extent.

In this paper, we investigate the success of LLMs for IE of music entities in conversational music queries (e.g., *give me some artists like metallica*).

We prompt LLMs to label utterances in the queries with respective labels (*title* and *artist*) and compare these to two strong baseline encoder-only models. We investigate the difference of two few-shot sampling methods and different numbers of few-shot examples. Lastly, we outline some contextual cues captured by the best performing LLM which we request in our prompt to reflect internal reasoning. We release our code publicly.[1]

In the next section, we outline related work in IE and NER using LLMs. In Section 3 we describe our proposed method. In Section 4 we describe our used dataset and baselines before presenting the results in Section 5. Lastly, we close this paper with Section 6.

## 2 Related Work

In this section we outline related work in IE and NER with LLMs. NER is a subtask of IE in which a sequence of labels per token or character is obtained. The more general task of IE refers to the extraction of relevant information in some structured form, but not necessarily a sequence of labels and possibly with additional steps (e.g., normalization). Research towards the use of LLMs for IE comprises the direct use of LLMs for the task (e.g., by instruction tuning) or auxiliary use in combination with encoder-only models (e.g., BERT).

A line of research has validated the usefulness of LLMs for NER. Beside Li et al. (2023) which fine-tune Llama-2, the most prevalent strategy appears to be prompting the LLM (Wang et al., 2023a;

[1] https://github.com/progsi/YTUnCoverLLM

Ashok and Lipton, 2023; Jung et al., 2024). Jung et al. (2024) relies on a single prompt without few-shot examples. Wang et al. (2023a) prompt GPT-3 and provide few-shot examples retrieved by a nearest neighbor search. Their approach achieves a performance close to the state-of-the-art based on BERT (Devlin et al., 2018). Ashok and Lipton (2023) demonstrate that GPT-3.5 and GPT-4 beat other LLMs such as T5XXL in the task. The authors include entity label descriptions and request reasoning for predicted entities. Sun et al. (2023) state that LLMs underperform in comparison to encoder-only models due to reasons like hallucination and context limits in few-shot settings. They provide various means to overcome this issue, such as demonstration retrieval and self-verification.

The mixed performance of LLMs for IE and NER motivates another paradigms which favors the auxiliary use of LLMs together with encoder-only models trained in a supervised fashion.

Zhang et al. (2024) argue that the lack of speciality of LLMs is a major factor. Hence, they propose an approach to combine those with encoder-only models, utilizing the LLM only for relabeling of initially uncertain predictions. Similarly, Ma et al. (2023) state that LLMs are better to use for hard samples than for general use in IE. They suggest to rather use LLM to re-rank of outputs obtained from a pre-trained encoder-only model. Other works include the auxiliary LLM purely for data augmentation Ye et al. (2024) or model distillation (Zhou et al., 2024; Peng et al., 2024). In the following, we propose our approach for IE from music queries using LLMs.

## 3 Music Entity Extraction with LLMs

The goal of our approach is the extraction of music entities from natural language queries. When users request music recommendations, the query can refer to various aspects such as genres, moods, titles (e.g., album or song titles) or performing artists (performers). We only focus on musical entities such as songs and albums (represented by their title) and performing artists (represented by their name). For each query, we aim to extract all the entities of this type and permit the possibility of no relevant entities being contained. Thus, the query *recommend me some rock songs* should yield no result, while the query *something similar to metallica st anger* should yield the utterance *metallica* with the label *performer* and the utterance *st anger* with the label *title* which refers to the American band *Metallica* and their album *St. Anger*. We model this task as an IE problem as we explain in the following.

**Instruction**
From the following text, which contains a user request for music suggestions, extract all the relevant entities that you find.

**Entity Attributes**
- **utterance**: The utterance of the entity in the text. For example "the beatles" in "recommend me music like the beatles". An utterance can only be of a type for which labels are defined.
- **label**: The label of the entity. It can either be 'TITLE' (if the utterance refers to a song or album name), 'PERFORMER' (if the utterance refers to a performing artist) or 'OTHER' for any other entity type.
- **cue**: The contextual cue which indicates the entity (e.g., "music like" in "recommend me music like the beatles" indicating "the beatles")

**Examples**
Input: stuff like flylo
({'utterance': 'flylo', 'label': 'performer', 'cue': ''})
Input: dré anthony brand new
...
**Output Schema**

```
from pydantic import BaseModel

class MusicEntity(BaseModel):
    """
    Data model of a music entity
    """
    utterance: str
    label: str
    cue: str
```

**Input**
songs similar to black bird by alter bridge

Figure 1: Prompt with few-shot examples and input text.

**Instruction** To obtain a structured output from an LLM, we define a Pydantic (Colvin et al., 2023) output schema and detailed instruction (cf. Figure 1). Previous approaches for IE with LLMs have discovered the importance of detailed attribute explanations with examples (Wang et al., 2023a; Ashok and Lipton, 2023; Zhang et al., 2023). Thus, we include explanations for each of the attributes. We also inclue a wildcard label *other* which we found helpful to improve the precision of LLMs (see Section 4). Beside utterance and label attributes, we request contextual cues.

**Contextual Cues** In theory, one can identify music entities in text by two ways. First, one can

simply rely on world knowledge. This way, even in difficult cases, one can identify a music entity with a correct label. For instance, the query containing *metallica st anger* which we showed earlier does not contain any cue clarifying that these are two entities and more precisely that *metallica* refers to a performer and *st anger* to an album. In contrast, some queries indicate the entity labels more clearly. For example *songs like nothing else matters by metallica* contains the cues *songs* and *by* which indicate the relationship *[song] by [performer]*. To gather more insights behind the internal reasoning of LLMs, we include an attribute *cue* in the structured output which should capture the contexts from the queries. This idea resembles the explanations requested in the prompt by Ashok and Lipton (2023).

**Few-Shot** Additionally to the zero-shot approach we described, we experiment with few-shot settings. We construct a few-shot example dataset which is the same dataset as the training dataset of the baseline models (cf. Section 4). We experiment with different numbers of $k$, corresponding to the amount of sampled examples at each iteration. Since the annotated dataset does not include contextual cues, we omit those in the few-shot examples. Beside random sampling, we experiment with a sampling approach that relies on the most similar $k$ items from the example dataset, similar to the nearest neighbor approach by Wang et al. (2023a), but we use term frequency inverse document frequency (tf-idf) vectors and the Cosine similarity as a metric. To not let the actual title and performer strings impact the similarity, we replace them in the examples by respective masks. For instance, in a example query *songs like nothing else matters by metallica* we obtain *songs like [song] by [performer]*.

## 4 Experimental Design

### 4.1 Implementation Details

We test different parameter values of $k \in \{0, 5, 15, 25, 35, 45\}$. We tested different LLMs for their capability to output structured content reliably. For example, we tested Llama-3-8B[2] but it failed too often to conform to the output structure. For our experiments, we use the following three LLMs:

**GPT-3.5-Turbo:** An LLM that supports function calling and is well suited for structured output.[3] We use *gpt-3.5-turbo-0125*.[4]

**Mistral-7B:** An open-source LLM by Jiang et al. (2023) suitable for structured output without function calling.

**Mixtral-8x7B:** An open-source LLM following the mixture of experts (MoE) paradigma (Jiang et al., 2024).

We also experimented with the use of the label *other* and compared the precision for Artist and WoA respectively. While Artist precision was relatively stable, we observed a decrease of 0.27 for WoA precision. That is, because a lot of more generic utterances like genres or moods were detected as WoAs. Thus, we decided to use the *other* label for all for all further experimental runs and we simply ignore the respective outputs to compute the WoA and Artist evaluation metrics.

### 4.2 Dataset & Baselines

We use the MusicRecoNER (Epure and Hennequin, 2023) dataset which is based on a subreddit[5] in which users request music suggestions by mentioning reference entities of the type performing artists or other entities such as song titles or music albums (labeled *title*).[6] The dataset is split into four subsets, three with 600 and one with 751 queries. On average, each query has two entity mentions but around 56% queries do not have any entity mention. We fine-tune two strong baselines and report the results using 4-fold cross validation as done by Epure and Hennequin (2023):

**BERT (Devlin et al., 2018):** A bi-directional encoder pre-trained by cloze tasks such as masked language modeling. It achieves comparable performance to MPNet (Song et al., 2020) for the task.

**RoBERTa (Liu et al., 2019):** This encoder has the same architecture as BERT but was pre-trained using a different training scheme. The model surpasses vanilla BERT on a variety of downstream tasks.

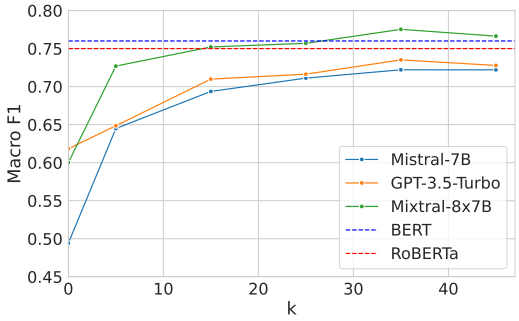
[2] see https://llama.meta.com/llama3/

[3] see OpenAI Function Calling Guide

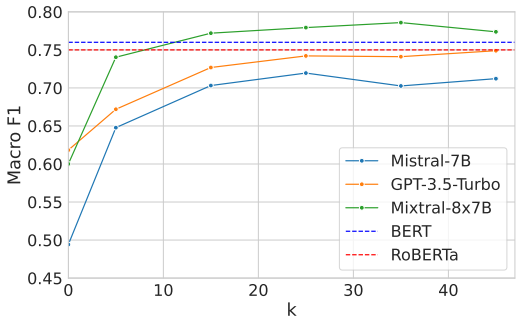
[4] see OpenAI Models

[5] www.reddit.com/r/musicsuggestions/

[6] Please note that we renamed the label in our prompts, since we found that the LLMs performance increased when using *title* instead of the original name *Work of Art* or *WoA*.




(a) F1 Scores for random few-shot sampling.




(b) F1 Scores tf-idf few-shot sampling.

Figure 2: Comparison of F1 Scores under strict evaluation scheme for different methods.

## 5 Results

In this section, we present the results of our previously presented experiments. All of the results are obtained with 4-fold cross validation using the split from Epure and Hennequin (2023).

In Figure 2 we show F1 scores as a function of $k$ for both sampling methods of few-shot examples. The effect of tf-idf sampling as opposed to random sampling seems to have no positive effect on the performance of Mistral-7B and just a minor effect on GPT-3.5-Turbo. Mixtral-8x7B is the best performing LLM and achieves higher F1 scores than the baselines for $k = 35$ and random sampling. Using tf-idf sampling, it exceeds the baseline for smaller values of $k = 15$ and $k = 25$ and it achieves the highest F1 score in the experiment at close to 0.80. However, the performance for $k = 45$ decreases which is also the case for the other models at tf-idf sampling. At random sampling, the performance appears to stagnate for $k \geq 35$ as well, but experiments with even higher values are necessary to fully exploit the potential of even more examples.

To gather more detailed insights in LLM performance against the baselines, we report the precision and recall of both entity labels in Table 1. While BERT has the highest recall for both labels, it has a lower precision by a substantial margin compared to GPT-3.5-Turbo and Mixtral-8x7B for performers. Apparently, the recognition of titles in the queries is a more difficult task, since all the models undershoot both metrics compared to performers.

| | **Perf.** | | **Title** | |
|---|---|---|---|---|
| | Pr | Re | Pr | Re |
| BERT | 0.81 | **0.82** | 0.72 | **0.77** |
| RoBERTa | 0.78 | 0.78 | 0.72 | 0.75 |
| GPT-3.5-Turbo | **0.91** | 0.78 | 0.64 | 0.65 |
| Mistral-7B | 0.78 | 0.71 | 0.74 | 0.61 |
| Mixtral-8x7B | 0.89 | 0.78 | **0.77** | 0.72 |

Table 1: Precision (Pr) and recall (Re) per labels performer and title for LLMs with $k = 35$ examples against the baselines.

Lastly, we investigate the contextual cues returned by the two best models: Mixtral-8x7B and GPT-3.5-Turbo with $k = 35$. We observe that cues indicating WoAs are less effective, resulting in 0.54 and 0.43 of WoA precision respectively. In contrast, the respective Artist precision is higher with 0.71 and 0.78. Frequent successful cues of Mixtral-8x7B are *ft*, *featuring* and *remix*. It is noteworthy, that Mixtral-8x7B and GPT-3.5-Turbo only returned cues in around 15% of cases, which might be due to the absence of cues in the few-shot examples.

## 6 Conclusion and Limitations

In this paper, we performed IE of music entities in conversational music queries with three LLMs as an alternative to the previously suggested NER with encoder-only models like BERT. We showed that tf-idf sampling to obtain similar few-shot examples to the query text can enhance the LLM performance, especially in case of the best performing model Mixtral-8x7B. The observed increase in F1 is mostly achieved by improved precision which leads to an overall improvement against the baselines. In future work, the inclusion of annotations for contextual cues could be helpful to encourage the LLMs to return those more frequently and possibly encourage better reasoning. Further, our study motivates experiments with even more capable LLMs such as Llama-3-70B.

## References


Dhananjay Ashok and Zachary C Lipton. 2023. Promptner: Prompting for named entity recognition. *arXiv preprint arXiv:2305.15444*.

Samuel Colvin, Eric Jolibois, Hasan Ramezani, Adrian Garcia Badaracco, Terrence Dorsey, David Montague, Serge Matveenko, Marcelo Trylesinski, Sydney Runkle, David Hewitt, and Alex Hall. 2023. Pydantic. If you use this software, please cite it as below.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

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.

Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, and Li Chen. 2021. A survey on conversational recommender systems. *ACM Comput. Surv.*, 54(5).

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.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts. *Preprint*, arXiv:2401.04088.

Sung Jae Jung, Hajung Kim, and Kyoung Sang Jang. 2024. Llm based biological named entity recognition from scientific literature. In *2024 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 433–435. IEEE.

Zongxi Li, Xianming Li, Yuzhang Liu, Haoran Xie, Jing Li, Fu-lee Wang, Qing Li, and Xiaoqin Zhong. 2023. Label supervised llama finetuning. *arXiv preprint arXiv:2310.01208*.

Sandra Liljeqvist. 2016. Named entity recognition for search queries in the music domain.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *Preprint*, arXiv:1907.11692.

Yubo Ma, Yixin Cao, Yong Hong, and Aixin Sun. 2023. Large language model is not a good few-shot information extractor, but a good reranker for hard samples! In *Findings of the Association for Computational Linguistics: EMNLP 2023*. Association for Computational Linguistics.

Letian Peng, Zilong Wang, Feng Yao, Zihan Wang, and Jingbo Shang. 2024. Metaie: Distilling a meta model from llm for all kinds of information extraction tasks. *arXiv preprint arXiv:2404.00457*.

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

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Preprint*, arXiv:2004.09297.

Xiaofei Sun, Linfeng Dong, Xiaoya Li, Zhen Wan, Shuhe Wang, Tianwei Zhang, Jiwei Li, Fei Cheng, Lingjuan Lyu, Fei Wu, and Guoyin Wang. 2023. Pushing the limits of chatgpt on nlp tasks. *Preprint*, arXiv:2306.09719.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023a. Gpt-ner: Named entity recognition via large language models. *Preprint*, arXiv:2304.10428.

Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, et al. 2023b. Instructuie: multi-task instruction tuning for unified information extraction. *arXiv preprint arXiv:2304.08085*.

Wenjia Xu and Yangyang Qi. 2022. Gazetteer enhanced named entity recognition for musical user-generated content. In *2022 3rd International Conference on Computer Science and Management Technology (ICCSMT)*, pages 40–43. IEEE.

Junjie Ye, Nuo Xu, Yikun Wang, Jie Zhou, Qi Zhang, Tao Gui, and Xuanjing Huang. 2024. Llm-da: Data augmentation via large language models for few-shot named entity recognition. *arXiv preprint arXiv:2402.14568*.

Mozhi Zhang, Hang Yan, Yaqian Zhou, and Xipeng Qiu. 2023. Promptner: A prompting method for few-shot named entity recognition via k nearest neighbor search. *arXiv preprint arXiv:2305.12217*.

Zhen Zhang, Yuhua Zhao, Hang Gao, and Mengting Hu. 2024. Linkner: Linking local named entity recognition models to large language models using uncertainty. In *Proceedings of the ACM on Web Conference 2024*, WWW '24, page 4047–4058, New York, NY, USA. Association for Computing Machinery.

Wenxuan Zhou, Sheng Zhang, Yu Gu, Muhao Chen, and Hoifung Poon. 2024. Universalner: Targeted distillation from large language models for open named entity recognition. *Preprint*, arXiv:2308.03279.