A Benchmark and Robustness Study of In-Context-Learning with Large Language Models in Music Entity Detection

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

Abstract

Detecting music entities such as song titles or artist names is a useful application to help use cases like processing music search queries or analyzing music consumption on the web. Recent approaches incorporate smaller language models (SLMs) like BERT and achieve high results. However, further research indicates a high influence of entity exposure during pre-training on the performance of the models. With the advent of large language models (LLMs), these outperform SLMs in a variety of downstream tasks. However, researchers are still divided if this is applicable to tasks like entity detection in texts due to issues like hallucination. In this paper, we provide a novel dataset of user-generated metadata and conduct a benchmark and a robustness study using recent LLMs with in-context-learning (ICL). Our results indicate that LLMs in the ICL setting yield higher performance than SLMs. We further uncover the large impact of entity exposure on the best performing LLM in our study.

1 Introduction

The detection of music entities (e.g., song titles and artists) in texts on the web can be of elementary use in various applications such as processing (conversational) search queries (Liljeqvist, 2016; Epure and Hennequin, 2023) or the analyses of music consumption on online video platforms. Within these use cases of named entity recognition (NER) in the music domain, the utterances typically originate from user-generated content (UGC).

The difficulties of NER in UGC have already been identified, for example, by Jijkoun et al. (2008) and Porcaro and Saggion (2019): users can express themselves freely, resulting in potential misspellings or abbreviated utterances of named entities. In the music domain a major challenge arises, which is also common in other creative content domains (e.g., movies, books or video games): Unlike other entity classes, such as names of persons, there is no known regular structure or defined vocabulary from which music entities are composed of (Derczynski et al., 2017; Brasoveanu et al., 2020). This renders utterances of musical entities susceptible to ambiguity. This phenomenon is not limited to cross-domain ambiguity (e.g., the term *Queen* as a band in contrast to the term representing a monarch), but also encompasses class discrimination within the music domain (e.g., the album *Queen* by the singer Nicki Minaj).

The currently proposed state-of-the-art approaches in NER are mostly based on encoder-only models like BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019). Although these have been shown to struggle with the aforementioned difficulties, higher exposure of entities in pre-training consequently leads to significantly higher performance of those in testing (Lin et al., 2020b; Epure and Hennequin, 2022, 2023). As a result, some approaches focus on contextual triggers within the context (Lin et al., 2020a; Ma and Liu, 2021).

Large language models (LLMs), such as GPT-4 (OpenAI et al., 2024) or Llama3 (Dubey et al., 2024), have been shown to master a variety of natural language tasks. In the case of NER, researchers are still divided if LLMs are the preferred choice for the task in contrast to smaller language models (SLMs)[1] like RoBERTa, due to problems like hallucination (Ma et al., 2023; Sun et al., 2023; Zhang et al., 2024). However, due to the usually much larger amount of pre-training data of LLMs in contrast to SLMs, the likelihood of music entity exposure during the pre-training is even higher. This strengthens the question about the performance of LLMs in the task as well as the ability to generalize to unseen entities.

In this paper, we aim to address these ques-

[1]We adopt the terminology to distinguish between SLMs and LLMs from Ma et al. (2023).

Proceedings of the 31st International Conference on Computational Linguistics, pages 9845–9859
January 19–24, 2025.

tions by conducting a benchmark study on a novel dataset of music entities in UGC using multiple recent LLMs with in-context-learning (ICL). In the second step, we conduct a controlled experiment to investigate the robustness of a selected LLM in which we show discover factors harming its performance. In summary, our contributions are twofold:

- We present an annotated dataset *MusicUGC-NER* for NER in the music domain based on user-generated content from the web. We release our dataset publicly[2] consisting of Reddit posts provided by Epure and Hennequin (2023) and YouTube video titles annotated by us. On our proposed dataset we conduct a benchmark comparing fine-tuned SLMs with LLMs using ICL for the task of NER of music entities in UGC.
- From our UGC dataset, we generate clozes to evaluate the robustness of LLMs with regards to unseen music entities and perturbations (e.g., typos and abbreviations) in the entity utterances.

The remainder of this paper is organized as following: in the next section, we outline related work regarding NER in the music domain and the task of information extraction, which is a broader task encompassing NER. In Section 3 we document our dataset creation procedure and present the corresponding descriptive statistics. In Section 4 we describe methodology to quantify exposure of LLMs and our data synthesis to generate new data using our cloze dataset. We describe our experimental design in Section 5 and the respective results in Section 6. We close the paper with the conclusion in Section 7 and reflect limitations of this study in Section 8.

2 Related Work

NER in the music domain Various works proposed NER approaches to detect music entities like musical artists and song or album titles in (user-generated) texts. Before the broad use of pre-trained language models, NER approaches were based on conditional random fields (CRF) (Liljeqvist, 2016; Porcaro and Saggion, 2019) or automated voting approaches (Oramas et al., 2016). Porcaro and Saggion (2019) propose an approach based on long short-term memory networks (LSTMs) together with CRFs for named entity recognition of classical musical entities. The data is also UGC since it is gathered from tweets to a radio channel profile. While this dataset also concerns the music domain, it is noteworthy that music entities in classical music are usually more regular than in western pop music, because they follow a structure (e.g., *Symphony No. 5* or *Symhony No. 9*).

With the rise of pre-trained SLMs, these were widely adapted for NER generally and in the music domain. (Xu and Qi, 2022) combine BERT (Devlin et al., 2018) in a mixture-of-experts approach with convolutional neural networks (CNNs) and LSTMs to improve upon the dataset of (Porcaro and Saggion, 2019). Liu et al. (2021) proposed pre-training and fine-tuning approaches with BERT to address cross-domain NER. Their created dataset comprises the music domain and four other domains and is based on Wikipedia articles. Another dataset by Epure and Hennequin (2023) focuses on the use case of conversational music recommendation. The dataset contains user requests for music suggestions on Reddit. We use this dataset and provide a joint dataset together with our annotated data as described in the following section.

IE with LLMs The task of IE deals with the automatic extraction of relevant structured information from unstructured text. Thus, it is a broader task which encompasses NER, but also other tasks such as relation extraction. Recently, LLMs are applied to a range of different IE problems. The strategies of LLM use incorporate zero- or in-context-learning (ICL) (Wang et al., 2023; Ashok and Lipton, 2023; Jung et al., 2024; Ma et al., 2024; Hachmeier and Jäschke, 2024), auxiliary use in combination with SLMs (Ma et al., 2023; Peng et al., 2024; Ye et al., 2024; Zhang et al., 2024; Zhou et al., 2024), fine-tuning (Li et al., 2023), or reinforcement learning (Huang et al., 2023; Ding et al., 2024).

In this paper, we employ a tf-idf-based few-shot prompting based on Hachmeier and Jäschke (2024), which has shown to be successful for music entities. The retrieval of similar few-shot examples to the items in the inference stage was used by other authors as well. Wang et al. (2023) achieve near state-of-the-art performance in NER with GPT-3 and a few-shot prompting approach where the closest examples for each unseen sample are retrieved with nearest neighbor search. Similarly, Ashok and Lipton (2023) achieve high NER performance using GPT-3.5 and GPT-4. It is noteworthy that some

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

authors state that LLMs are still not outperforming SLMs in the task due to the increased output space and problems such as hallucination (Ma et al., 2023; Sun et al., 2023; Zhang et al., 2024).

Sun et al. (2023) propose various ideas to mitigate this, such as self-verification and a few-shot demonstration retrieval. Other authors favor the auxiliary use of LLMs together with SLMs. Ma et al. (2023) propose to use LLMs only for re-ranking the SLM outputs; since they claim that LLMs are of better use for hard samples than easy ones. Similarly, Zhang et al. (2024) only utilize LLMs to re-label uncertain SLM predictions. Other techniques include the use of LLMs for data augmentation Ye et al. (2024) or model distillation (Zhou et al., 2024; Peng et al., 2024).

3 Data

Our goal is to benchmark LLMs for music entity detection in UGC. In this section, we describe how we created a novel dataset of YouTube video titles containing music entities. Our dataset is provided in the inside-outside-beginning (IOB) format (Ramshaw and Marcus, 1995) where each text represents a YouTube video title and the tags are referring to entity mentions of either a work of art, such as song titles and albums titles (WoA), or performing artists (Artist). We later join our dataset with MusicRecoNER (Epure and Hennequin, 2023) to cover two different types of UGC, namely online video metadata and Reddit posts.

3.1 Dataset Creation

Data Sources Our dataset contains a subset of works from SHS100K (Xu et al., 2018) which is a large collection of cover songs of (mostly western) popular music. A key advantage for using SHS100K is its carefully curated metadata on the platform Secondhandsongs (SHS),[3] provided by community volunteers. Each cover song is represented by rich information, such as a song title (WoA), an artist name, a composer, a release year, and a link to a YouTube video containing a performance of the respective song. This knowledge base serves to test LLMs for factual knowledge in Section 4.1.

To obtain the UGC utterances, we crawl the corresponding video titles from YouTube under consideration of the fair use policy.[4] From the original SHS100K dataset we retain 89,763 representations of videos which are still available on YouTube at the time of dataset creation. 76% of the representations are in the training set from the initial split from Xu et al. (2018) and the remaining 24% account for approximately half of the initial validation and test set. We annotate an approximately equal amount of the initial train, validation and test subsets. Since we can make use of the song-level metadata from SHS, we decided to apply an automatic matching to make the annotation process more efficient. We provide details about the respective pre-processing and matching steps in Appendix A.1.

Human Annotation We further obtain annotations by two annotators from our organization and one author. In our annotation tool (see Appendix A.2.5) we show the WoA and Artist variations obtained in our pre-processing step in Section A.1 from SHS. The annotators can then select the respective IOB tags per token in a drop-down menu. We provide more details on the annotation protocol in Appendix A.2.

In total, we obtain 609 annotated items, each with two annotators which yield very high agreement (Cohen's Kappa of 0.93 on average). In the following, we refer to our annotated dataset as D-YT.

3.2 Joining with MusicRecoNER

Additionally, we join our data with the MusicRecoNER dataset (Epure and Hennequin, 2023). This dataset is based on Reddit queries by users requesting music recommendations. We use the unprocessed Reddit queries, since their pre-processing includes removal of various special characters, as the authors focused on a conversational use case rather than web data. Since MusicRecoNER only contains IOB tags for the processed dataset, we must relabel the word sequences. To achieve this, we align the unprocessed and processed word sequences by matching every word in the unprocessed sequence to the respective word in the processed sequence based not only on the word itself, but also its relative position. The resulting gaps, which are special characters, are labeled based on the two surrounding tags. For instance, if an unlabeled token x in the unprocessed sequence is surrounded by tags of one class and one utterance (e.g., *B-WoA* x *I-WoA* or *I-WoA* x *I-WoA*), x is assigned an inside tag of the same class (here I-WoA). In all other cases, x

[3] https://secondhandsongs.com/

[4] see: Google Support Answer no. 9783148

Dataset	Items	Words		Entities	
		WoA	Artist	WoA	Artist
D-YT	609	3/15	2/9	1/3	1/4
D-RD+YT	2,977	2/15	2/9	0/5	0/7

Table 1: Statistics (median/maximum) of the words per entity utterance (*Words*) and the entity utterances per sample (*Entities*) for D-YT and D-RD+YT.

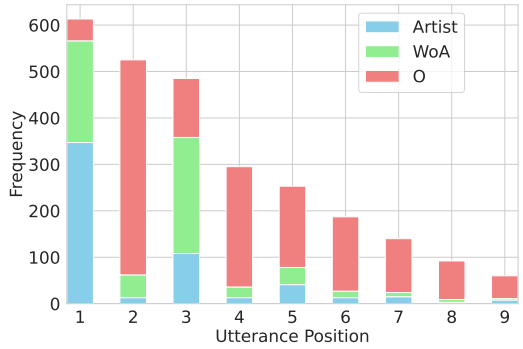

Figure 1: Relative positions of the utterances per class in D-YT up to the 9th utterance. *O* refers to the outside tag in the IOB format.

is assigned an outside tag. We refer to this dataset as D-RD and to the combined dataset of the latter with D-YT as D-RD+YT. For five-fold cross validation we stratify both datasets to five subsets each representing approximately the same ratio of YouTube texts to Reddit texts and we ensure that the same WoAs and Artists only occur in one of the subsets together.

3.3 Dataset Statistics

As can be seen in Table 1, WoA utterances appear to be longer than Artist utterances. We additionally checked the number of representations without WoA and Artist utterances in the curated subset which account for 6% and 23% respectively.

Figure 1 shows the relative position of utterances regarding the classes in D-YT. We observe that video uploaders mostly mention the artist before the WoA. The second utterance is neither WoA nor Artist in the majority of cases, which can indicate the use of a separator (e.g., a dash) before the WoA. However, the utterance order is widely spread as we will see later.

4 Robustness Study

We aim to evaluate the robustness of LLMs using our dataset. To isolate the surrounding contexts from the entities and to ensure the same class sizes for seen and unseen entities, we additionally perform an experiment on a synthesized dataset using the best performing LLM GPT-4o-mini.

As discussed in Section 1, the exposure of entities in pre-training can have a significant impact on the performance of SLMs. Thus, we focus on the robustness with regard to unseen entities. Since our domain concerns UGC which is prone to peculiarities such as typos, we further investigate the robustness of the LLM towards perturbations.

4.1 Quantifying Exposure

Factual Memorization Test Using the rich SHS metadata as described in Section 3.1, we are able to construct a test to model factual memorization of LLMs as defined by Hartmann et al. (2023). In the following, we refer to this test as FMT.

In our domain of cover songs, factual knowledge can be modeled on the level of a musical work. We regard a musical work as a group of cover songs. By definition, a musical work has a composer and an original version with a corresponding original artist (Yesiler et al., 2021). Based on these two attributes, we can model factual knowledge as tuples with a subject (the work), a relationship and an object, for instance: *(Yesterday; composed_by; John Lennon, Paul McCartney)*. Based on the relationships to original artists and composers of musical works, we construct our FMT with two questions as shown in Figure 2. Based on the outcomes of the test with regard to the two questions, we distinguish between three FMT outcomes:

Passed ✓ Both questions answered correctly.

Partial (✓) One question is correctly answered or at least one answer contains an artist entity which is a performing artist of any cover of the work.

Failed ✗ All other outcomes.

We conduct the FMT for all entities in D-YT, due to the necessity of entity links to SHS. We provide more details about the implementation of the factual memorization test in Appendix A.3. Next, we also retrieve a subset of real-world music entities that we use for data synthesis.

Debut Artists Beside relying solely on our data, which might be memorized by our used LLMs even if the means in the previous sections indicate otherwise, we use music entity information of works

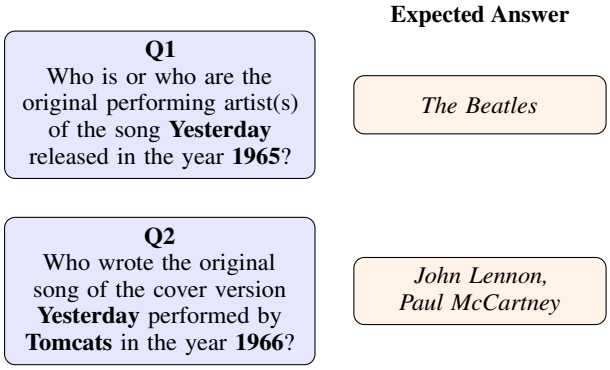

Figure 2: Questions of our factual memorization test (FMT) on the example of the musical work *Yesterday* originally performed by The Beatles.

which are released after the knowledge cutoff of GPT-4o-mini which is at the end of 2023. We sample random entities per cloze and denote the resulting synthesized dataset as Post-Cutoff. We use the API of MusicBrainz[5] to obtain 100 international debut artists of the year 2024 with their debut WoA released. More details about the crawling of MusicBrainz is supplied in Appendix A.4.

4.2 Data Synthesis

We synthesize data of UGC in the music domain based on our joint dataset described in Section 3.2. To investigate the impact of context tokens which surround the entity utterances, we first create clozes in the IOB format.

Cloze Dataset A cloze is a text where some words are masked. In natural language processing, cloze tasks are used to train language models to predict the masked words. In our case, we create clozes from our dataset to construct templates of surrounding context which can be applied to different music entities.

In all texts, we replace the full utterances by a one-word mask per class. Thus, all words which represent B- or I- are replaced. For example, the sequence *songs like bohemian rhapsody* is replaced by *songs like [WoA]*. Using this strategy, we can distinguish the isolated surrounding contexts independently of entity mentions. Furthermore, we replace years with a mask, which we detect with regular expressions.[6] After these steps and the removal of utterances without any entity mention (only outside tags), we obtain 1,067 unique clozes. We use these clozes to synthesize data by filling in music entities from randomly sampled WoAs with corresponding Artists from two of the FMT outcomes which we denote as FMT Passed, and FMT Failed respectively. Additionally, we create a dataset using the entities from Post-Cutoff. As a result, we obtain three datasets of the same size using all of the unique clozes. In the next step, we use these three datasets to create perturbed versions.

Perturbations As described before, certain perturbations are not unlikely in the case of UGC. We model different types of perturbations in the utterances of music entities, namely character-level and word-level perturbations of entity mentions similar to Feng et al. (2024). Table 2 shows corresponding examples: Word-level perturbation modifies the sequence of words by either deleting or shuffling tokens. Character-level perturbation alters the characters within a word by performing deletions, insertions, or substitutions. We additionally consider a third type of perturbation which addresses abbreviations which are sometimes used in artist utterances. To model abbreviations, we simply use the first character per word in the artist string. We create two perturbed datasets, which we denote as Level-1 and Level-2. We set the perturbation probability to $p = 0.5$. In Level-1 with a probability of p we apply one randomly selected perturbation out of two (word-level or character-level). In Level-2, we apply up to two perturbations each with a probability of p. The first time, it is either an abbreviation perturbation or a word-level perturbation. The second time, it is a character-level perturbation. For more details about the perturbation implementation, please refer to our repository.

Level	Operation	Example
Input	None	*johnny b goode*
Character	Deletion	*jonny b goode*
Character	Insertion	*johnny b goodey*
Character	Substitution	*johnni b goode*
Word	Deletion	*johnny b*
Word	Shuffle	*johnny goode b*

Table 2: Example of perturbations per type for the WoA *Johnny B. Goode*.

[5] https://musicbrainz.org/doc/MusicBrainz_API

[6] We match words with four characters and starting with *19* or *20*.

5 Experimental Design

With regards to our first contribution, we conduct benchmarks on D-YT and D-RD+YT using five-fold cross validation. We ensure that the same entities are only occurring in one fold. In D-RD+YT, we stratify the folds to gather approximately the same ratio of items from D-YT and D-RD. For the robustness study described in Section 4, we use all 1,067 templates for each of the three synthesized datasets and split the synthesized sets two-fold into test and few-shot sets during experiments to avoid using the exact same clozes in both sets. With the identifier on music entity level, we ensure that utterances of the same music entity do not occur in both sets. In the following, we first describe the SLM and LLM models used for our benchmarks.

5.1 NER with SLMs

We fine-tune RoBERTa (Liu et al., 2019) and BERT (Devlin et al., 2018) for the sequence labeling task. Hence, each of the two masked language models transforms an input sequence into a sequence of IOB tag predictions. We use the training parameters as proposed by Epure and Hennequin (2023). Since we observed that training more epochs was beneficial, we trained the models on 5 epochs instead of 2.

5.2 NER-like IE with LLMs

Instruction We use LLMs with a prompt validated in a previous study (Hachmeier and Jäschke, 2024). Rather than mapping the input texts to a sequence of IOB tags like in sequence labeling, we use LLMs with ICL in an IE fashion[7] and extract information into structured output format. The output formats depend on the LLM and are either JavaScript Object Notation (JSON) (Pezoa et al., 2016) or Pydantic (Colvin et al., 2023). The key *utterance* maps to the detected music entity in exactly its uttered form which might include potential typos and abbreviations. The key *label* maps to the class of the music entity, namely WoA or Artist. This way, we are able to match the utterances with the input texts to obtain a sequence of labels like in NER. In previous works, several authors discovered the relevance of detailed attribute explanations to effectively leverage LLMs in IE (Wang et al., 2023; Ashok and Lipton, 2023; Zhang et al., 2023). Therefore, we include detailed attribute explanations. We provide more details about the prompt from (Hachmeier and Jäschke, 2024) in Appendix A.5.

Sampling of Few-Shot Examples We use the training split as a few-shot example dataset. At each iteration, k few-shot examples are sampled to be included in the prompt. We employ a tf-idf-sampling approach that retrieves most similar examples to the current sample which was shown to be superior to simply random sampling Hachmeier and Jäschke (2024).

LLMs We benchmark the following LLMs:

FireFunction-v2 A model based on Llama3-70B but optimized for function calling. The authors claim that its performance in benchmarks related to function calling tasks is comparable to GPT-4o (Garbacki and Chen, 2024). We use the Ollama commit *b1ed6b22fb67*.[8]

GPT-4o-mini The smaller version of the most recent flagship model of OpenAI (OpenAI, 2024) advancing in performance over the prior series of GPT models (OpenAI et al., 2024). We use the version *gpt-4o-mini-2024-07-18*.[9]

Llama3.1-70B The 70B version of the most recent iteration of Llama models (Dubey et al., 2024). We use the Ollama commit *c0df3564cfe8*.[10]

Mixtral-8x22B A larger version of the model proposed by Jiang et al. (2024). It follows the mixture of experts (MoE) paradigm (Mistral AI Team, 2024) and was the best model in our previous study (Hachmeier and Jäschke, 2024). We use the Ollama commit *e8479ee1cb51*.[11]

For reproducibility, we set the temperature parameter to 0 for all experiments and use the respective default parameters when using LLMs via the OpenAI API and Ollama, respectively. The output format is defined in Pydantic schema for GPT-4o-mini and Mixtral-8x22B and in JSON for Llama3.1-70B and FireFunction-v2.

[7] We experimented with sequence labeling, but found that the LLM outputs for sequence labeling were generally not very reliable. It is noteworthy that sophisticated methods can counteract this issue (Dukić and Šnajder, 2024).

[8] https://ollama.com/library/firefunction-v2

[9] https://platform.openai.com/docs/models/gpt-4o-mini

[10] https://ollama.com/library/llama3.1:70b

[11] https://ollama.com/library/mixtral:8x22b

5.3 Metrics

To measure the overall benchmark results, we consider the F1 scores per entity class (WoA and Artist) and their macro average in the strict evaluation scheme (Segura-Bedmar et al., 2013).

In the robustness study on our dataset, we focus on three erroneous outcomes of NER as defined by Batista (2018). Given an example text with *the beatles* as actual Artist and *yesterday* as actual WoA entities, we show the three outcomes:

Incorrect Correct entity boundaries but incorrect type or correct type but incorrect boundaries (e.g., *the beatles* as WoA or *yesterday* as Artist).

Spurious Neither boundaries nor type matches (e.g., *beatles* as WoA).

Missed Entity not matched at all (e.g., *the beatles* assigned with two outside tags).

6 Results

6.1 Benchmark

Table 3 shows the benchmark results. We observe that the LLM performance increases at least up to $k = 15$ few-shot samples for all models. This is especially due to the increased performance in detecting WoA entities which appear to be generally harder to detect than Artist entities.

The performance of the baseline SLMs is higher for BERT than for RoBERTa, but generally lower than the performance of LLMs, especially when comparing to GPT-4o-mini and FireFunction-v2.

The best model is GPT-4o-mini, which in the zero-shot setting almost competes with all other models in few-shot settings, but the highest performance is achieved with $k = 35$. Thus, we decide to use GPT-4o-mini in our robustness study of which we present the results in Section 6.2.

In Table 4 we report the results per subset based on the results of the FMT introduced in Section 4.1. We observe that the recall of WoA recognition drops by a large margin and up to .24 for GPT-4o-mini when comparing the performance at works with passed FMT and failed FMT. The effect is less apparent when comparing the results of the outcome *partial*. However, we can already observe a drop in recall here for the models Mixtral-8x22B and GPT-4o-mini. The results indicate that in fact the factual knowledge has an impact on the NER performance, similarly like in the case of SLMs. Hence, we further investigate the LLM more closely by our synthesized data.

LLM	k	D-YT Artist/WoA/Avg.	D-RD+YT Artist/WoA/Avg.
FireFunction-v2	0	.76/.67/.71	.78/.68/.73
	5	.83/.78/.80	.82/.70/.80
	15	.86/.81/**.84**	.84/.78/.81
	25	.84/.81/.82	.85/.78/.81
	35	.85/.80/.82	.86/.79/.82
GPT-4o-mini	0	.85/.78/.81	.86/.75/.81
	5	.86/**.82**/**.84**	**.88**/.77/.82
	15	**.87**/.81/**.84**	**.88**/.78/.83
	25	.86/.81/**.84**	**.88**/.79/.83
	35	.85/**.82**/**.84**	**.88**/**.80**/**.84**
Llama3.1-70B	0	.81/.78/.79	.82/.70/.76
	5	.82/.81/.81	.82/.74/.78
	15	.84/.81/.83	.84/.76/.80
	25	.83/**.82**/.83	.84/.76/.80
	35	.82/.81/.82	.84/.75/.80
Mixtral-8x22B	0	.81/.68/.75	.73/.67/.80
	5	.83/.79/.81	.84/.75/.79
	15	.83/.80/.82	.86/.78/.82
	25	.83/.80/.82	.86/.78/.82
	35	.83/.80/.81	.86/.78/.82
RoBERTa	-	.78/.72/.75	.78/.74/.76
BERT	-	.82/.74/.79	.80/.73/.76

Table 3: Mean F1 scores (a/b/c) for a) Artist, b) WoA, and c) macro average (Avg.) between a) and b) using the strict evaluation scheme (Segura-Bedmar et al., 2013) on the datasets using five-fold cross-validation. Highest results are marked in bold.

	✓		(✓)		✗	
Mixtral-8x22B	.80	(338)	.68	(317)	**.67**	(46)
Llama3.1-70B	.78	(368)	.73	(332)	.61	(53)
FireFunction-v2	.69	(302)	.68	(389)	.54	(60)
GPT-4o-mini	**.85**	(229)	**.75**	(420)	.61	(103)

Table 4: Recall in detecting WoAs and the respective support per outcome of the of the FMT (Correct: ✓, Partial: (✓), and False: ✗). All reported results are with few-shot settings with $k = 35$.

6.2 Robustness Study

We investigate the robustness of the best-performing model in our benchmark, namely GPT-4o-mini. Figure 3 shows the metrics for error anal-

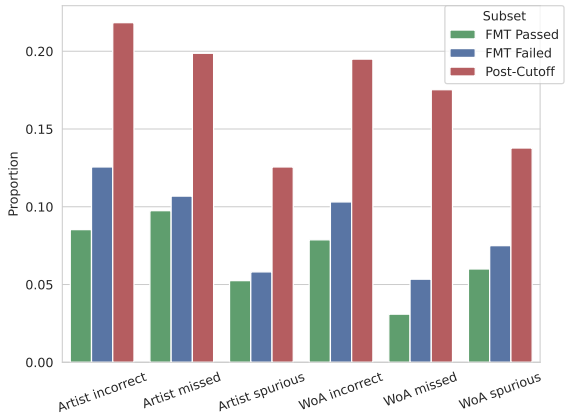

Figure 3: Proportions of errors per group based on our synthesized datasets without perturbation. The total amount is 1,067, the number of all unique clozes.

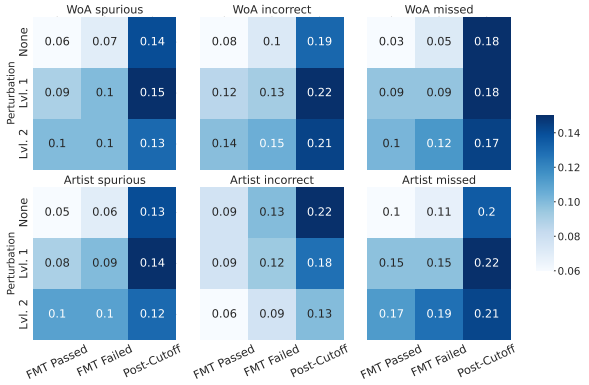

Figure 4: Error proportions per metric per imposed perturbation level and synthesized dataset. The total amount is 1,067, the number of all unique clozes.

ysis. Apparently, the proportions of all errors increase slightly when comparing entities from the passed FMT with the failed FMT. However, a much larger difference is visible in case of the entities from Post-Cutoff. We further see that the most prominent problems are incorrect Artists or WoAs as well as missed Artists. In Figure 4 we analyze the impact of perturbation and exposure on the ability of robustness of GPT-4o-mini. In case of the two groups which originate from our dataset and the FMT groups, increasing the level of perturbation also increases the proportion of errors in almost all cases with the highest increase being for missed artists. Interestingly, in case of Post-Cutoff the effect is given for all metrics and the perturbation levels decrease the errors for some metrics, such as incorrect Artists. However, this can be due to the dependency of different errors on each other: for example, an actual WoA is predicted as an Artist and hence results in a *missed* WoA and an *incorrect* Artist. Overall, we observe that the effect of exposure appears to be stronger than the effect of perturbation.

Lastly, we examine the relevance of the surrounding contexts of music entities. In Figure 5 we compare by the data sources YouTube and Reddit for Post-Cutoff. The distributions indicate that the contexts of Reddit are generally more helpful for the LLM to detect unseen entities. This is probably due to the richer contextual cues in questions (e.g., *songs like ...*) than in online video metadata. However, for both data sources we identified erroneous surrounding contexts (see Table 8 in Appendix A.6).

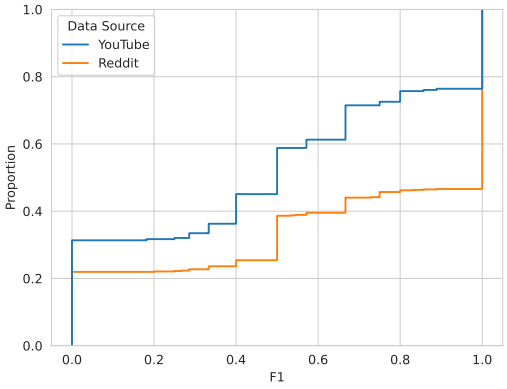

Figure 5: Cumulative distribution functions of F1 scores per data source on our synthesized dataset.

7 Conclusion

In this paper, we propose a novel dataset for NER of music entities in user-generated content. The dataset was created using human annotations supported by automatic annotation and additionally joint with the MusicRecoNER dataset. In a benchmark experiment, we compared four different LLMs in an ICL setting to strong baselines. In our second experiment, we synthesize data to gather more insights about the impact of perturbation and entity exposure on the LLM performance. Our results indicate that LLMs with ICL are a strong choice for music entity extraction. However, part of this appears to be due to entity exposure during pre-training. In future studies, logical next steps include the consideration of music gazetteers, which has been pursued in the past (Xu and Qi, 2022). In context of LLMs, these could be combined with retrieval-augmented generation methods.

8 Limitations

With regard to our dataset it is noteworthy that the focus lies on western popular music and covers from other regions are not represented as broadly. Furthermore, the genders of artists are hard to determine based on the metadata on Secondhandsongs. Thus, we cannot guarantee gender diversity. It is noteworthy that we employed three annotators of our organization which are all working in an academic context on a daily basis.

In our experiments, we used open-source LLMs and one closed-source LLM, namely GPT-4o-mini. It is possible that the benchmark performance can still be surpassed by other state-of-the-art models, for example, GPT-4o or the 405B version of Llama3.1-70B, which we cannot run due to local resource limitations. For all LLMs we relied on the default parameters for quantization and did not test different configurations.

Lastly, we clarify that we purely investigated the performance on our task in focus and did not specifically address the scalability. For instance, both BERT and RoBERTa have less parameters than the tested LLMs. In some cases, the former might still be an attractive alternative, even though they achieve inferior performance on the task.

Acknowledgments

We thank our student assistants Chris Herrmann and Jonathan Lüpfert for supporting the annotation process.

References

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

David S. Batista. 2018. Named-entity evaluation metrics based on entity-level. Accessed: 2024-09-13.

Adrian MP Brasoveanu, Albert Weichselbraun, and Lyndon Nixon. 2020. In media res: a corpus for evaluating named entity linking with creative works. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 355–364.

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.

Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 2017. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147, Copenhagen, Denmark. Association for Computational Linguistics.

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.

Zepeng Ding, Ruiyang Ke, Wenhao Huang, Guochao Jiang, Yanda Li, Deqing Yang, Yanghua Xiao, and Jiaqing Liang. 2024. Adaptive reinforcement learning planning: Harnessing large language models for complex information extraction. *arXiv preprint arXiv:2406.11455*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, and Chris Marra et al. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

David Dukić and Jan Šnajder. 2024. Looking right is sometimes right: Investigating the capabilities of decoder-only llms for sequence labeling. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 14168–14181.

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.

Xiaoning Feng, Xiaohong Han, Simin Chen, and Wei Yang. 2024. Llmeffichecker: Understanding and testing efficiency degradation of large language models. *ACM Trans. Softw. Eng. Methodol.*, 33(7).

Pawel Garbacki and Benny Chen. 2024. Firefunction-v2: Function calling capability on par with GPT4o at 2.5x the speed and 10% of the cost. Accessed: 2024-09-06.

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)*.

Valentin Hartmann, Anshuman Suri, Vincent Bindschaedler, David Evans, Shruti Tople, and Robert West. 2023. Sok: Memorization in general-purpose large language models. *arXiv preprint arXiv:2310.18362*.

Wenhao Huang, Jiaqing Liang, Zhixu Li, Yanghua Xiao, and Chuanjun Ji. 2023. Adaptive ordered information extraction with deep reinforcement learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13664–13678, Toronto, Canada. Association for Computational Linguistics.

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.

Valentin Jijkoun, Mahboob Alam Khalid, Maarten Marx, and Maarten De Rijke. 2008. Named entity normalization in user generated content. In *Proceedings of the second workshop on Analytics for noisy unstructured text data*, pages 23–30.

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.

Bill Yuchen Lin, Dong-Ho Lee, Ming Shen, Ryan Moreno, Xiao Huang, Prashant Shiralkar, and Xiang Ren. 2020a. Triggerner: Learning with entity triggers as explanations for named entity recognition. *arXiv preprint arXiv:2004.07493*.

Hongyu Lin, Yaojie Lu, Jialong Tang, Xianpei Han, Le Sun, Zhicheng Wei, and Nicholas Jing Yuan. 2020b. A rigorous study on named entity recognition: Can fine-tuning pretrained model lead to the promised land? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7291–7300, Online. Association for Computational Linguistics.

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.

Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2021. Crossner: Evaluating cross-domain named entity recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13452–13460.

Liwen Ma and Weifeng Liu. 2021. An enhanced method for entity trigger named entity recognition based on pos tag embedding. In *2021 IEEE 7th International Conference on Cloud Computing and Intelligent Systems (CCIS)*, pages 89–93. IEEE.

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.

Zhiwei Ma, Javier E Santo, Greg Lackey, Hari Viswanathan, and Daniel O'Malley. 2024. Information extraction from historical well records using a large language model. *arXiv preprint arXiv:2405.05438*.

Mistral AI Team. 2024. Cheaper, better, faster, stronger continuing to push the frontier of ai and making it accessible to all. Accessed: 2024-09-06.

OpenAI. 2024. Hello GPT-4o. Accessed: 2024-09-11.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, and Kevin Button et al. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Sergio Oramas, Luis Espinosa Anke, Mohamed Sordo, Horacio Saggion, and Xavier Serra. 2016. ELMD: An automatically generated entity linking gold standard dataset in the music domain. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3312–3317, Portorož, Slovenia. European Language Resources Association (ELRA).

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*.

Felipe Pezoa, Juan L Reutter, Fernando Suarez, Martín Ugarte, and Domagoj Vrgoč. 2016. Foundations of json schema. In *Proceedings of the 25th International Conference on World Wide Web*, pages 263–273. International World Wide Web Conferences Steering Committee.

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

Lance A. Ramshaw and Mitchell P. Marcus. 1995. Text chunking using transformation-based learning. *Preprint*, arXiv:cmp-lg/9505040.

Ekagra Ranjan and Naman Poddar. 2022. Multilingual abusiveness identification on code-mixed social media text. *Preprint*, arXiv:2204.01848.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. 2013. SemEval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (DDIExtraction 2013). In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 341–350, Atlanta, Georgia, USA. Association for Computational Linguistics.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

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. 2023. Gpt-ner: Named entity recognition via large language models. *Preprint*, arXiv:2304.10428.

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.

Xiaoshuo Xu, Xiaoou Chen, and Deshun Yang. 2018. Key-invariant convolutional neural network toward efficient cover song identification. In *2018 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6.

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*.

Furkan Yesiler, Guillaume Doras, Rachel M Bittner, Christopher J Tralie, and Joan Serrà. 2021. Audio-based musical version identification: Elements and challenges. *IEEE Signal Processing Magazine*, 38(6):115–136.

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.

A Appendix

A.1 Automatic Annotation

Pre-Processing We conduct various pre-processing steps to ensure a robust automatic matching where we match the SHS content against the UGC from YouTube. First, all texts are transformed to lowercase and apostrophes are removed. We then apply specific pre-processing methods for the two data sources due to their peculiarities. In the SHS song-level metadata we found that WoAs are often accompanied by additional information in brackets, such as *(acoustic)* or *(remastered)*. We discard this additional information to just retain the creative content. In case of the artist strings, we consider artists string variations with and without articles (i.e., *the beatles* and *beatles*). We found that featuring artists are represented within single strings in the SHS metadata. Hence, we separate the artists to detect these as individual entities. For both, articles and representations for featuring separators (e.g., *feat.*), we use pre-defined lists covering multiple languages which we provide in Appendix A.1.1. In case of the YouTube video titles, we discovered the use of font-like non-Latin texts. To enable matching these texts as well, we perform Unicode normalization similar to (Ranjan and Poddar, 2022). For example, the Unicode character *Mathematical Fraktur Capital F*(code point U+1D509) is normalized to *Latin Capital Letter F* (code point U+0046).

A.1.1 Pre-Defined Lists

We consider the following languages for our dataset, which are contained in the source dataset SHS100K and which represent frequent languages in western popular music: English, French, German, Italian, Portuguese, and Spanish. For each of these languages, we document the articles (e.g., *the* for English or *der*, *die* and *das* for German) and different expressions representing separators for featuring artists. For the latter, we consider the form *and* + *pronoun* (e.g., *and her*, for cases like

Billie Holliday and her Orchestra) and *with + pronoun* (e.g., *and her*, for cases like *Billie Holliday with her Orchestra*). The details can be found in the preprocessing script in our repository.

A.1.2 Matching

We run an algorithm to match the pre-processed variations of song-level attributes WoA and Artist from SHS with the pre-processed YouTube metadata texts. We use the partial alignment ratio,[12] which is the normalized Indel similarity of the optimal alignment of the shorter string to the longer string. Since it is handling alignment of strings of different lengths it is well suited for our use case, where often the utterances occur at different word indices among additional information (e.g., *. . . performing yesterday in . . .*). We set the minimum matching threshold to $\tau = 80$ of 100.

Table 5 shows the resulting matching statistics. From the subset of samples where both entities match, in 87% of the cases both entities are matched with $s = 100$. For human annotation, we sample a minimum of 150 representations randomly stratified to subsets shown in Table 5 and the initial subsets of SHS100K. Using the annotated dataset which we obtain in Section 3.1, we are able to evaluate the automatic matching algorithm, which yields 0.92 in precision and 0.50 in recall. Since the low recall indicates that half of the entities are missed, we only make use of our human annotated dataset in this paper. However, we also provide the automatically matched data in our repository.

Matched Entities	Count	Fraction
Both	77,889	87%
Only WoA	7,061	8%
Only Artist	3,986	4%
None	827	1%

Table 5: Numbers of samples with matches for both attributes (top) and with at least one non-matching attribute (bottom) based on the similarity s and threshold $\tau = 80$.

A.2 Annotation Guidelines

The purpose of this annotation task is to label each token in YouTube video titles using the inside-outside-beginning (IOB) format. The task focuses on identifying two specific classes: Artist and WoA (Work of Art).

A.2.1 Representation

Each title is split into individual tokens (words, punctuation marks, etc.). A token typically corresponds to a word, but punctuation marks (e.g., commas, apostrophes) and special characters are treated as separate tokens. All text is converted to lowercase before annotation.

A.2.2 Classes

The following classes are in the focus of this annotation task:

Artist This refers to the name of a musical performing artist or group. It includes singers, bands, DJs, or any individual/group credited for the creation or performance of the music. Group members which do not perform music as individuals are excluded.

WoA This refers to the titles of songs, albums, EPs, or any other artistic work related to music.

A.2.3 IOB Format

The IOB format was proposed by Ramshaw and Marcus (1995) and consists of three tags:

B- (Beginning) Indicates the first token of a named entity.

I- (Inside) Indicates any token that is inside a named entity but not the first one.

O (Outside) Indicates tokens that do not belong to any named entity.

If a named entity (here: Artist or WoA) consists of a single token, label it with the B- prefix (i.e., B-Artist or B-WoA). If a named entity spans multiple tokens, label the first token with the B- prefix and the subsequent tokens with the I- prefix (i.e., B-Artist I-Artist). All other tokens not related to Artist or WoA should be labeled as O.

A.2.4 Ambiguous Cases

Entity Utterances with Additional Information Any additional information, such as regarding the version of the WoA, should be labeled as O. For instance in *the beatles - yesterday (karaoke version)*, only *yesterday* should be labeled as WoA. This results in *B-Artist I-Artist O B-WoA O O O O*.

[12] https://rapidfuzz.github.io/RapidFuzz/Usage/fuzz.html#partial-ratio-alignment

Ambiguity Between Artist and WoA If a token could be interpreted as either an Artist or a WoA, use the surrounding context to make the correct annotation. If context is insufficient, try to find the correct class for the utterance on the web.

Incorrect Tokens In cases where the title contains additional, incorrect tokens within a WoA, annotate all relevant tokens as part of the WoA to maintain the entity's integrity. For example, *nothing else random matters* should be annotated as *B-WoA I-WoA I-WoA I-WoA* to treat the phrase as a single entity despite the inclusion of the irrelevant word *random*. However, if the number of incorrect tokens is large and the entity is not recognizable, you can consider splitting the utterance or just annotating half of the utterance with the corresponding class. Ultimately, this is a case-by-case decision and should depend on your perception of readability.

Featuring Artists In titles that include featured artists, both the main artist and the featured artist should be annotated. For example, *rihanna feat . drake* should be annotated as *B-Artist O O B-Artist.*

Punctuation Marks Punctuation marks should generally be labeled as O unless they are part of the official name of the Artist or WoA. *p ! ink* should be annotated as *B-Artist I-Artist I-Artist*.

Nested Entity Utterances Sometimes entity utterances can be nested. In these cases, favor the outermost entity. For example, *b - sides the beatles* should be labeled as one WoA entity, because the utterance refers to the tribute album *B-Sides The Beatles* from The Beatles. The only exception are medleys. These shall be annotated as separate WoAs.

A.2.5 Tool Usage

The annotation process will be conducted using our custom annotation tool which was developed using Streamlit.[13] An example of the tool's interface is shown in Figure 1, illustrating how tokens are presented with corresponding dropdowns for IOB tag selection and how reference information is displayed.

Dropdown Selection In the tool, each token of the video title is displayed with a dropdown menu underneath it. Annotators should use these dropdowns to assign the appropriate IOB tags B-Artist, I-Artist, B-WoA, I-WoA, O).

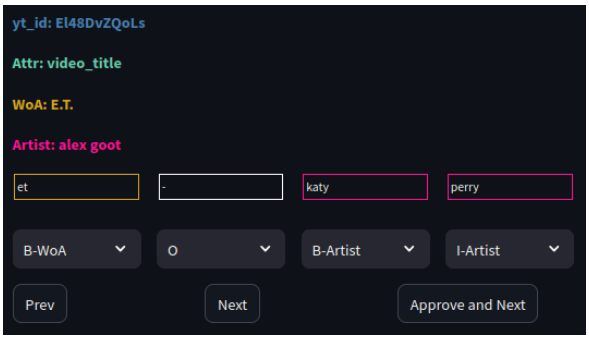

Figure 6: Graphical user interface of our annotation tool. The example shown is a cover of *E.T.* by Katy Perry from the performing artist Alex Goot.

Reference Information To assist with accurate annotation, the tool also displays one correct Artist and WoA (Work of Art) that are relevant to the YouTube video. This information is supplementary, since also other WoAs or Artists shall be annotated.

A.2.6 Examples

Some examples for correct annotations are given in Table 6.

adele	-	hello					
B-Artist	O	B-WoA					
the	beatles	-	hey	jude			
B-Artist	I-Artist	O	B-WoA	I-WoA			
rihanna	feat	.	drake	-	work		
B-Artist	O	O	B-Artist	O	B-WoA		
taylor	swift	-	red	(	deluxe	)	
B-Artist	I-Artist	O	B-WoA	O	O	O	

Table 6: Examples for correct annotations.

A.3 Factual Memorization Test

Figure 7 provides an overview of the outcomes of the factual memorization test using Llama3.1-70B. Further, we provide some details about our definition of correctness.

We model the correctness of an answer as a matching string to the ground truth attribute from SHS. Since multiple strings (artists or composer) can be correct answers (e.g., Paul McCartney and John Lennon are composers of the song *Yesterday*),

[13] https://github.com/streamlit/streamlit

we decided that one correct answers is sufficient. For the string matching step, we apply the same preprocessing techniques as described in Section A.1.

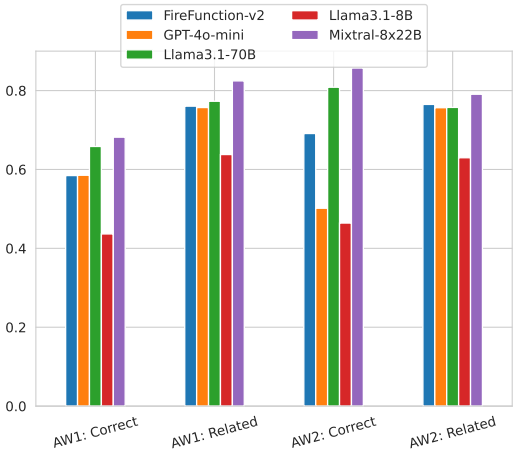

Figure 7: Fractions of correctly answered questions in the memorization test. We distinguish between *correct* and *related* where the artist/composer mentioned is related (e.g., a covering artist) to the work, but not strictly correct.

A.4 Debut Artists from MusicBrainz

Using the MusicBrainz API, we crawl for artists with the query text *begin:2024*. We limit the results to the first 100. We show some examples of the artists and corresponding debut releases in Table 7.

A.5 In-Context-Learning

Prompt Figure 8 shows an example of a prompt to the LLMs with sampled few-shot examples. We also found that the use of a third label as a wildcard (*other*) is helpful to improve LLM performance, since in many cases with no true entities the models still labeled utterances as *WoA*. Beside utterance and label attributes, we request contextual cues.

tf-idf Sampling Term frequency inverse document frequency (tf-idf) (Sparck Jones, 1972) is a measure of the importance of words in a document in information retrieval. We obtain tf-idf vectors for texts during few-shot sampling. To focus on the similarity of the syntactical structure of the UGC utterances rather than the content of the tokens of the attributes, we mask the entities when computing the tf-idf similarity. For instance, for the example *songs like nothing else matters by metallica* we get *songs like [WoA] by [Artist]*. The prompt contains the actual texts as shown in Figure 8. We compared to random sampling (Hachmeier and Jäschke, 2024) which turned out to yield inferior

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

Figure 8: Prompt with few-shot examples and input text.

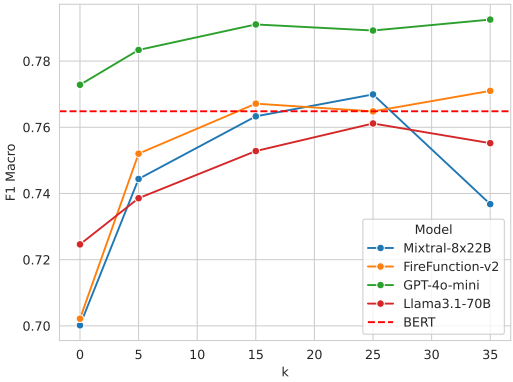

Figure 9: Performance on D-RD+YT using random sampling as opposed to tf-idf-sampling.

performance. Figure 9 shows the corresponding performance of random sampling for different values of k on D-RD+YT.

A.6 Clozes

Figure 10 shows the distributions of numbers of outside tokens. Figure 11 shows our clozes in the two-dimensional plane after conducting t-SNE. Table 6 shows some of the clozes yielding the highest number of errors per error type.

Artist	Debut WoA	Release Date
Ben Keller	Fake Yøu øut	02-06-2024
The Houseboat Tapes	The Houseboat Tapes	03-13-2024
Human Fade	Getter	01-06-2024
Green Buffalo Steak Ensemble	001: Mary Juana Had a Little Lamb	05-03-2024
Veskraunghulthyr	Lornmyr Frost	02-28-2024
I:mond	WE ARE GRAVITY	06-04-2024
Hitori Kakurenbo	Maquetas	01-27-2024

Table 7: Examples of debut artists in 2024 and their debut WoAs retrieved from MusicBrainz.

Artist	**Incorrect**	[Artist] - [WoA] - [Artist] ! ! geroiam [Artist] ! ! rap songs like [Artist] [Artist] sings [Artist] (full album - [Year])
	Spurious	[Artist] - [WoA] (remix) old [Artist] / [Artist] sounding dudes looking for music that is like [Artist] music
	Missed	who is the french [Artist] ? what is the name of this kind [Artist] ? [Artist] (ft . [Artist]) cover of [WoA] by [Artist]
WoA	**Incorrect**	[WoA] - [WoA] (disco version) hip hop similar to [WoA] from [WoA] album ? songs / bands like the metalcore [WoA] from [WoA] : waw zombies ?
	Spurious	live in central park [revisited] : [Artist] [WoA] - pickin on [Artist] : a bluegrass tribute to [Artist] - pickin on series similar songs to [Artist] - [WoA] ?
	Missed	any songs with some minor [WoA] themes ? trolls [WoA] comic - con clip | trolls [Artist] / [Artist] [[WoA]] live audio cover

Table 8: Clozes by data source: YouTube and Reddit.

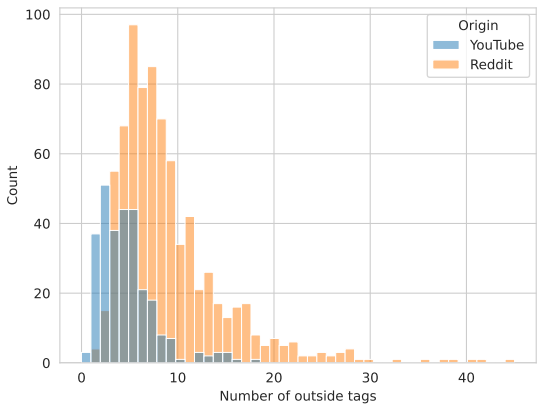

Figure 10: Distribution of the number of outside tokens.

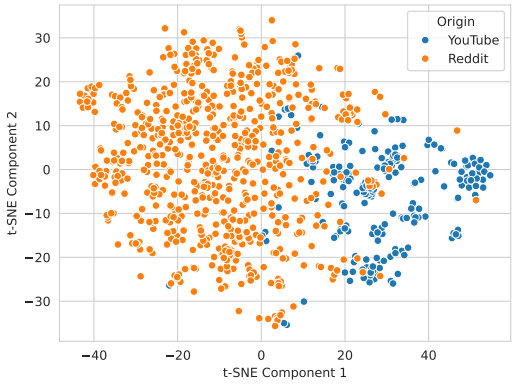

Figure 11: Clozes after dimensionality reduction with t-SNE.