# Multimodal music information processing and retrieval: survey and future challenges

we can recognize music played on different instruments, with different timings, intensity changes, tonalities, tunings, background noises and so on.

Better results through multimodal approaches.

In this paper, we review the existing literature about Music Information Retrieval.

Ways to digitalize music information: audio, lyrics, album covers, symbolic scores.

Multimodal music information processing as an MIR approach which takes as input multiple modalities of the same piece of music.

Multimodal music processing tasks:

some tasks have been extensively studied with a multimodal approach, such as audioto-score alignment, score-informed source separation, music segmentation, emotion or mood recognition; other tasks, instead, have been little explored and are worth of more attention.

classification of music, synchronization of different representations, similarity computation between two or more modalities, and time-dependent representation.

- Similarity:

Often, this task has the purpose of retrieving documents from a collection through a query.

A very common example of explicit queries for retrieving another modality is query-by humming or query-by example, in which the query is represented by an audio recording and the system retrieves the correct song.

An example of implicit query systems, instead, are recommender systems and playlist generators.

Most of the recent research in this field tries to exploit multimodal approaches – also called hybrid – involving metadata, user context, audio features.

An emerging field in the retrieval context is the so-called multimodal queries [16], [24].

- Classification:

The classification process consists in taking as input a music document and returning one or more labels.

Examples: mood or emotion recognition, genre classification, artist identification, derivative works classification, instrument classification, tonic identification, expressive musical description.

Feature extraction in multimodal approaches

Various types of features can be extracted from each modality. In this section, we provide a general description for audio, video, textual and symbolic score features.

- text features [70]: The most common text representations are based on tf-idf. In order to make BoW and tf-idf models effective, a few preliminary steps are usually performed, such as punctuation and stop-words removal and stemming. More sophisticated methods are also available, allowing for topic- or semantics-based analysis, such as Latent Dirichlet Allocation (LDA), Latent Semantic Analysis (LSA), Explicit Semantic Analysis (ESA) and CNN feature extraction.

Conversion to common space

The conversion of the extracted features to a common space is often a mandatory step in early fusion approaches. The conversion to a common space consists in the mapping of the features coming from different modalities to a new space where they are comparable.

many papers describe techniques which include a mapping of the features to a common space, both in the pre-processing and in the processing stages.

Common methods include: normalization, conversion from one modality to another, machine learning algorithms and dimensionality reduction algorithms which usually search for a new space where data samples are representable with a fewer number of dimensions without losing the ability to separate them

Future Directions

First of all, we note the unavailability of datasets of suitable size. Although a few datasets have been recently created [37], [76], [78], a great effort should still be carried out in this direction. Indeed, existing multimodal music datasets are usually characterized by limited sizes and only rarely include a wide range of modalities.

We argue that this limit is due to two main reasons: first, the precise alignment of

various modalities is a hard computational task and should be controlled by human supervision; second, no largely adopted standard exists for multimodal music representation.

# Audio Retrieval with Natural Language Queries: A Benchmark Study

The objectives of this work are cross-modal text-audio and audio-text retrieval, in which the goal is to retrieve the audio content from a pool of candidates that best matches a given written description and vice versa. Text-audio retrieval enables users to search large databases through an intuitive interface: they simply issue free-form natural language descriptions of the sound. … these tasks have received limited attention in the existing literature.

The vast and unabated growth of user-generated content in recent years has introduced a pressing need to search ever-growing databases of multimedia. Free-form natural language sentences (i.e. sequences of text that are written as they would be spoken) form an intuitive and powerful interface

for composing search queries for these databases since they allow for expressing virtually any concept. Spanning multiple modalities, different retrieval strategies were developed for content as diverse as text (including web pages and books), images [1], and videos [2], [3]. Surprisingly, while search

engines currently exist for these modalities (e.g. Google, Flickr and YouTube, respectively), unstructured audio is not accessible in the same way.

It is important to distinguish content-based retrieval from that based on metadata, such as the title of a video or song or an audio tag. Metadata retrieval is feasible for manually curated databases such as song or movie catalogues. However, content-based retrieval is more important in user-generated

data, which often has little structure or metadata. There are methods to search for audio which matches an audio query [4], [5], but satisfying the requirement to input an example audio

query can be difficult for a human (e.g. making convincing frog sounds is difficult). We, on the other hand, propose a framework which enables the searching of a sound database using detailed free-form natural language queries of the desired sound .

Furthermore, natural language queries are a familiar user interface widely used in current search engines. Therefore, our proposed audio retrieval with free-form text queries could be a first step towards more natural and flexible audio-only search.

Text-based audio retrieval could also be beneficial for video retrieval. The majority of recent works that address the text-based video retrieval task focus heavily on the visual and text domains [1], [2], [6], [7]. Since audio and visual information inherently have natural semantic alignment for a significant portion of video data, text-based audio retrieval could also be used for querying video databases by only considering the audio stream of the video data.

Learning to retrieve audio, given natural language queries, requires data with paired text and sound. Audio captioning <<definition of captioning>> datasets naturally lend themselves to this task, since they contain audio and a matching text description for the sound.

However, existing captioning datasets are limited in size and in the diversity of their audio content.

Audio captioning consists of generating a natural language description for a sound [27]. This requires a more detailed understanding of the sound than simply mapping the sound to a set of labels (sound event recognition).

Audio-based retrieval. Multiple content-based audio retrieval frameworks, in particular query by example methods, leverage the similarity of sound features that represent different aspects of sounds (e.g. pitch, or loudness) [37], [38], [39], [5]. More recently, [4] use a twin neural network framework to learn to encode semantically similar audio close together in the embedding space. [40] address multimedia event detection using only audio data, while [41] tackle near-duplicate video retrieval by

audio retrieval.

Text-based video retrieval. More closely related to our work, a number of methods showed that embedding video and text jointly into a shared space (such that their similarity can be computed efficiently) is an effective approach [1], [3], [2], [6], [43], [7], [44] (though other formulations, such as

computing similarities directly in visual space have also been explored [45]). One particular trend has been to combine cues from several “experts”—pre-trained models that specialise in different tasks (such as object recognition, action classification etc.) to inform the joint embedding. Recently, transformer-based architectures have demonstrated impressive results for text-based video retrieval [7], [46], [47]. In this work, we propose to adapt three expert-based methods: the Mixture of Embedded Experts method of [2], the Collaborative Experts model of [6], and the Multi-Modal Transformer [7] by repurposing them for the task of audio retrieval (described in more detail in Sec. IV).

Cross-domain audio retrieval. Methods that retrieve audio by matching associated text, such as metadata or sound event labels, have the implicit assumption that the text is relevant [48]. In contrast, [49] is an early work that proposes to link audio and text representations in hierarchical semantic and acoustic spaces. [50] builds on this using mixture-of-probability-expert models for each of the modalities. ……

Other works have explored using images [55] or video data [56], [57], [58], [59], [60] as queries for retrieving audio. More recently, [61] use a twin network to learn a shared latent text and sound space for cross-modal retrieval.

Problem formulation. Given a natural language query (i.e. a written description of an audio event to be retrieved) and a pool of audio samples, the objective of text-audio (abbreviated to

t2a) retrieval is to rank the audio samples according to their similarity to the query.

Methods. To tackle the problem of text-audio retrieval, we propose to learn cross-modal embeddings. we aim to learn embedding functions, that project each audio sample and text sample into a shared space, such that ai  and ti in said space are close when the text describes the audio, and far apart

otherwise.

Experiments

Retrieval performance metrics are recall at rank k and median and mean rank.

In this section, we first compare text-audio retrieval (t2a) and audio-text retrieval (a2t) performance on audio-centric and visual-centric datasets. Next, we perform an ablation study on the contributions of different experts and present our baselines for the proposed AUDIOCAPS, CLOTHO, and SOUNDDESCS benchmarks. Finally we perform experiments to assess the influence of pre-training, audio segment duration and training dataset size, and give qualitative examples of retrieval results. Throughout the section, we use the standard retrieval metrics: recall at rank k (R@k) which measures the percentage of targets retrieved within the top k ranked results (higher is better), along with the median (medR) and mean (meanR) rank. For all metrics, we report the mean and standard deviation of three different randomly seeded runs.

# TOWARD UNIVERSAL TEXT-TO-MUSIC RETRIEVAL

An ideal text-based retrieval system would support various input queries such as pre-defined tags, unseen tags, and sentence-level descriptions. In reality, most previous works mainly focused on a single query type (tag or sentence) which may not generalize to another input type.

The demand for efficient music retrieval has been increasing as massive music libraries become easily accessible. While various methods have been proposed for efficient retrieval [1, 2, 3, 4], text-based retrieval remains the most prevalent [5, 6, 7]. Text-based retrieval is challenging because it needs to handle not only editorial metadata (e.g., title, artist, release year) but also semantic information (e.g.,

genre, mood, theme). Furthermore, modern retrieval systems, such as voice assistants [8], need to generalize to sentence-level natural language inputs beyond fixed tag vocabularies.

While much research has addressed text-based retrieval, there are two dominant approaches: classification and metric learning. Classification models [9, 6] are trained with a set of fixed tag labels,

and then the predicted tags are utilized in retrieval. Despite its successful classification performance, this approach is limited to a fixed vocabulary. In contrast, metric learning models are more flexible

by using pre-trained word embeddings [10, 11] or language models [12, 13, 14, 15]. Especially pre-trained language models enable free-form text inputs for music retrieval by representing sentence-level semantics.

An ideal text-based retrieval system needs to be flexible to allow various input types (e.g. word, sentence) and abundant vocabularies. For example, one can use broadly used tags, such as genre, to explore the music library. Sometimes the input queries may include unseen types of music tags. Also, another can use more detailed sentence-level descriptions to discover music.

However, to the best of our knowledge, previous works mainly focused on improving a single type of input queries. Also, they are using respective datasets and evaluation metrics which makes it difficult to choose the appropriate solution for universal music retrieval.

Text Encoding

We use tag and sentence text representation for input of the text encoders. For this, we use a pre-trained word embedding GloVe [21] and a pre-trained Bidirectional Encoder Transformer (BERT) [22]

with a base-uncased architecture.

In the case of BERT model, the input text sequence is tokenized by wordpiece tokenizer, and the max sequence length is 64. Similar to audio feature embedding, the text sequence attaches [SOS] token at first position and the output of the last layer of the transformer at the [SOS] token are treated as the feature representation of the text which is layer normalized and then linearly projected embedding space.

# MULAN: A JOINT EMBEDDING OF MUSIC AUDIO AND NATURAL LANGUAGE

Music tagging and content-based retrieval systems have traditionally been constructed using pre-defined ontologies covering a rigid set of music attributes or text queries. This paper presents MuLan: a first attempt at a new generation of acoustic models that link music audio directly to unconstrained natural language music descriptions. MuLan takes the form of a two-tower, joint audio-text embedding model trained using 44 million music recordings (370K hours) and weakly-associated, free-form text annotations. Through its compatibility with a wide range of music genres and text styles (including conventional music tags), the resulting audio-text representation subsumes existing ontologies while graduating to true zero-shot functionalities. We demonstrate the versatility of the MuLan embeddings with a range of experiments including transfer learning, zero-shot music tagging, language understanding in the music domain, and cross-modal retrieval applications.

Classifiers are generally trained to label examples with predefined and fixed class inventories, which are often manually specified as a structured ontology indicating interclass relationships. Empowered by recent advances in neural language modeling and their demonstrated transfer learning competence, researchers have begun exploring less restrictive natural language interfaces to access

the categorical information underlying raw content signals. The majority of this work has been in the visual and audio event domain, where a recent series of studies have demonstrated the utility of jointly embedding media content with natural language captions [1–5]. These joint embeddings have demonstrated strong capabilities in a range of applications, including transfer learning, cross-modal retrieval, automatic captioning, and zero-shot classification.

The success of these efforts strongly depends on large-scale training resources and hefty neural network architectures that are flexible enough to model the complex, non-monotonic relationship between language and other modalities. In particular, the visual domain has greatly benefited from the availability of large amounts of captioned images available across the web [1]. However, in the general environmental audio domain, such large-scale audio-caption pairs are less readily available and related efforts have relied on small captioned datasets [6, 7]. Critically, these datasets do not span the diversity of sound-descriptive language and their success in the more difficult zero-shot setting has been lacking [3, 8, 9].

our strategy is to assemble a collection of textual annotations extracted from metadata, comments, and playlist data and map them to a training set of over 44 million internet music videos. As was the case with image-text model training in [1], our text data only truly refers to the musical content in a fraction of cases. Therefore, we also explore text pre-filtering using a text classifier separately trained to identify music descriptions.

Music text joint embedding models

Content-based music information retrieval requires linking the rich semantics expressible to free-form text with both broad and fine-grained musical properties. One approach is to consider a large number of text label classes and try to ground the semantics in music with a multi-label classification task …..

our two-tower parallel encoder approach results in a joint embedding space that provides a natural language interface to arbitrary music audio. This opens up downstream opportunities for cross-modal retrieval, zero-shot tagging, and language understanding.

Evaluation:

Music Retrieval from Text Queries

Given a music search collection and a text query, MuLan provides the ability to retrieve the music clips that are closest to the query in the embedding space. This evaluation is relevant to music retrieval applications, where content features can offer finer-grained and more complete similarity information when compared with metadata-based methods [41].

Results and Discussion

Music Retrieval from Text Queries

Even though we start with a BERT checkpoint pretrained with massive language resources, training MuLan with only AudioSet clips and label annotations provides very limited ability to ground indomain natural language to music. Such limited crossmodal supervision does not generalize to the rich semantics that appear in the playlist titles and descriptions, which are more in line with the complex queries that are presented to real-world music search engines. We observe significant gain after including the large-scale short-form tags mined from the internet, which helps the model learn to ground more fine-grained music concepts. There is additional gain when including comments and playlist data, where the complete sentences are helpful for grounding the more complex queries, including multi-term queries (e.g.‘instrumental action movie soundtrack’), compositional queries (e.g. ‘classical music with middle eastern influence’), and even queries with negation (e.g. ‘hard rock without vocals’). Again, we find that training is surprisingly robust to annotation noise, achieving similar performance using unfiltered training text.

# Contrastive audio-language learning for music (25/08/22)

As one of the most intuitive interfaces known to humans, natural language has the potential to mediate many tasks that involve human-computer interaction, especially in application-focused fields like Music Information Retrieval.

Our approach consists of a dual-encoder architecture that learns the alignment between pairs of music audio and descriptive sentences, producing multimodal embeddings that can be used for text-to-audio and audio-to-text retrieval out-of-the-box.

Developing effective methods for finding music is at the core of Music Information Retrieval (MIR). Over the years, many approaches have been proposed to browse, search and discover music through a variety of interfaces. Beyond simple search by metadata, existing music retrieval systems

allow to express queries via lyrics [1,2], audio examples [3], videos [4] and humming [5], among others [6–8]. Although each of these query types has its merits, none of them supports one of the most popular ways of searching for music today: through free-form text. For example, we commonly look for songs by typing text into a search engine [9] or by asking online song naming communities to identify a piece of music we do not have bibliographic information about [10]. It becomes evident then that enabling MIR systems to interpret natural language queries can have farreaching benefits.

This idea is not entirely new to MIR, with prior work [11–13] suggesting similar research directions in

the past. So far, however, multimodal systems that integrate natural language have not been widely adopted within the MIR community, possibly due to a lack of suitable datasets or to the practical limitations of NLP methods predating modern language models.

In light of recent breakthroughs in language modelling, we argue that audio-and-language learning has now the potential of closing the semantic gap in MIR [14], providing a bridge between computational representations of music signals and the high-level abstractions needed to use those

representations in real-world scenarios.

we propose MusCALL, a method for learning alignment between music-related audio and language data via multimodal contrastive learning. Our choice of a contrastive approach is inspired by the recent success of similar methods for joint visio-linguistic modelling (see Section 2.3). In designing MusCALL, we prioritise the ability to perform retrieval at scale and adopt a dual-encoder architecture,

where modalities are processed independently. Compared to multimodal architectures with joint encoders and crossmodal attention mechanisms [15], this design allows to share embedding computations among pairs, resulting in a computationally more efficient model.

Our primary contributions can be summarised as follows: (i) we explore multimodal contrastive learning in the context of music-related audio and language for the first time; (ii) we introduce a method for cross-modal retrieval of music, providing the first example of sentence-based music search; (iii) we perform an extensive set of experiments to systematically validate details of our approach and evaluate its performance on popular MIR tasks in a zero-shot setting.

<https://github.com/ilaria-manco/muscall>

Natural Language Processing in MIR

Prior works in the MIR literature have explored leveraging natural language in the music domain from different angles. Early efforts focused on text as a modality in isolation, adopting NLP techniques to construct knowledge bases from music-related text corpora [16], build semantic graphs for artist similarity from biographies [17], or perform genre classification based on album reviews [18]. Recent efforts, more closely related to the present work, have instead started favouring multimodal approaches. These have explored deep learning with multimodal input data, typically audio combined with text such as reviews or lyrics, for applications as varied as music classification and recommendation [19], mood detection [20], music emotion recognition [21] and music captioning [22–24].

Audio-Text Cross-modal Learning

Others have explored pre-trained word embeddings in triplet networks to perform tag-based music retrieval via a multimodal embedding space [28, 29]. At a high level, these works share a similar approach to ours and all aim to learn multimodal audio representations by leveraging text. Unlike our work, however, none of them makes use of natural language, using tags as text input instead.

Finally, similarly to our work, [30] also addresses the problem of matching audio and long-form text for music retrieval, but offers a fundamentally different approach, which relies on bridging the audio and text modalities via a common emotion embedding space.

Learning from Language Supervision

The key insight behind these models is that language captures many of the abstractions humans use to navigate the world and can therefore act as a rich supervisory signal for general purpose learning, even in tasks that are not directly based on language [35]

In adjacent fields such as computer vision, jointly modelling vision and language has instead

become a very active area of research, with several successful attempts at using multimodality to develop task-agnostic models that can easily adapt to novel tasks [31-34] …… These breakthroughs suggest that natural language supervision has a large potential beyond the image domain. This

has recently prompted the adoption of similar approaches in machine listening, where applications to both music [15] and non-music audio [36, 37] have started to emerge.

# Improving information retrieval by semantic embedding (2020?)

This research focuses on using semantic embedding to improve the performance of Information Retrieval (IR) for the Covid-19 related tasks. According to previous research, the technology of word embedding can significantly improve the performance of IR.

… the IR [1] method of these two data sets is keyword-based. Although information retrieval with the traditional keyword-based [2] method is workable, there is a large room for improvement in efficiency and accuracy. In the traditional keyword-based way, you can retrieve the accurate results with certain keywords, but … . Searching information with a keyword-based search engine is often time-consuming and returns poor data results, which reduces the efficiency of

the process of coronavirus-related research. We hope to have smarter word embedding [3] IR models that considers semantic similarities of words in the matching process, to generate high

usable searching results.

Word and document embedding

In this research, word2vec, Fasttext and GloVe are word embedding frameworks. They cannot be used to do document searching directly. After the word embedding training complete, each document in the dataset must be converted into a document embedding vector. This process is accomplished by the following algorithm:

*for d in documents:*

*for entity in d:*

*v = e.vector\_embedding*

*v2 = v2 + v*

*v3 = mean(v)*

*result.append(v)*

*return result*

Cosine similarity …

In this research, three main properties are required to be measured: precision, recall, and relevance. The followings are the definition of precision and recall: Precision@1: the first result is the target one. Recall@10: the target result exists in the result set.

# Semantic similarity for music retrieval (2007)

Our query-by-example music information retrieval (MIR) system takes an audio track as a query and retrieves new audio tracks that have similar semantic descriptions to the query track. For example, given a piece of music that a listener might describe as “crazy guitar rock with a screaming female singer that makes me want to get up and dance”, our system ranks all retrievable songs by how well they fit this description. The system is based on the models of [9, 3] which have shown promise in the domains of audio and image retrieval. Audio models are learned from a database of audio tracks with associated text captions that describe the audio content.

The semantic feature for a track, c, is a bag of words, represented as a binary vector, where ci = 1 indicates the presence of word wi in the text caption.

Query-by-semantic-example (QBSE) is an information retrieval method that has been applied to images [6], sound effects [1] and music [9]. QBSE uses semantic information to retrieve semantically meaningful audio from the database. In many cases, a semantic understanding of the audio signal enables retrieval of tracks that, while acoustically different, are semantically similar to the query. For

example, given a query with a high pitched, electric guitar sound, a system based on acoustics alone might retrieve songs with other high-pitched, harmonic sounds like violins or a female vocalist. On the other hand, QBSE would retrieve acoustic guitars, distorted guitars or banjos.

# Evaluating Embedding APIs for Information Retrieval (6/07/2023)

The ever-increasing size of language models curtails their widespread availability to the community, thereby galvanizing many companies into offering access to large language models through APIs. One particular type, suitable for dense retrieval, is a semantic embedding service that builds vector representations of input text.

With a growing number of publicly available APIs, our goal in this paper is to analyze existing offerings in realistic retrieval scenarios, to assist practitioners and researchers in finding suitable services according to their needs. Specifically, we investigate the capabilities of existing semantic embedding APIs on domain generalization and multilingual retrieval.

Language models (LMs), pre-trained on a massive amount of text, power dense retrieval models in ad

hoc retrieval (Lin et al., 2021b). Dense retrievers (Lee et al. 2019; Karpukhin et al. 2020; Xiong et al.

2021; Khattab and Zaharia 2020; Hofstätter et al. 2021; Izacard et al. 2022; inter alia) essentially

measure relevance via similarity between the representations of documents and queries. As LMs are

rapidly scaling up to gigantic models (Radford et al. 2019; Brown et al. 2020; Lieber et al. 2021; Chowdhery et al. 2022; Smith et al. 2022, inter alia), their use as the backbone of dense retrieval models has become limited primarily because large language models (LLMs) are computationally expensive and deploying them on commodity hardware is cumbersome and often impractical.

To alleviate this problem, many companies, e.g., OpenAI, and Cohere, set out to offer access to their

proprietary LLMs through a family of APIs.

APIs

- Aleph-Alpha : three flavours in size, base (13B), extended (30B), and supreme (70B). The luminous models support five high-resource languages: English, French, German, Italian, and Spanish.

<https://docs.aleph-alpha.com/docs/introduction/luminous>

- Cohere: two sizes, small (410M) and large (6B), generating 1024-dimension and 4096-dimension embedding vectors

<https://docs.cohere.ai/docs/representation-card>

- OpenAI: We use the recommended second-generation model, text-embedding-ada-002 (Neelakantan et al., 2022) that embeds text into a vector of 1536 dimensions. The model, initialized from a pretrained GPT model, is fine-tuned on naturally occurring paired data with no explicit labels, mainly scraped from the web, using contrastive learning with in-batch negatives.

Our analysis is based on information reported as of Feb 1, 2023. OpenAI and Aleph-Alpha charge based on the number of tokens and model size: ada2 and luminous base cost $0.0004 USD and e0.078 ≈ $0.086 6 per 1,000 tokens. On the other hand, Cohere follows a simpler cost structure, charging based only on the number of API calls, i.e., $1.00 USD per 1,000 calls. Our re-ranking experiments on BEIR cost around $170 USD on OpenAI, whereas it would cost roughly $2,500USD on Cohere based on their quoted prices. Cohere offers a free-tier access with a restricted API call rate limit of 100 calls per minute, which we opted for, albeit sacrificing speed.

Our findings suggest that re-ranking BM25 results is a suitable and cost-effective option for English; on the BEIR benchmark, OpenAIada2 performs the best on average. In multilingual settings, while re-ranking remains a viable technique, a hybrid approach produces the most favorable results. We hope that our insights aid practitioners and researchers in selecting appropriate APIs based on their needs in this rapidly growing market.

# Vector model based information retrieval system with word embedding transformation (2022)

Vector based information retrieval system has been one of the trending methods in Natural Language Processing. The embeddings vector generated from a document helps in identifying most relevant document related to the query. There is various approach were embedding vectors can be generated and some of them which have implemented are Word2vec, Glove2vec and Sentence BERT. For information retrieval system also used word embedding transformation like PCA and Factor Analysis to improvise the model’s performance. Most of information retrieval system

involves getting query from the user, preprocessing of the query and generating most relevant information to the query. Results obtained by post processing methods such as PCA and Factor Analysis shows a comparatively better results with an increase of 2-3% of Mean average precision.

In pattern recognition and other domains, existing information retrieval models, such as the vector space model, are built on predefined criteria to model text. The Vector Space Model (VSM) divides, filters, and classifies material that appears to be very abstract, and then applies statistics to the text's word frequency data [4].

The search and retrieval of knowledge-based information from databases is what information retrieval (IR) is all about. [2] many concepts and strategies for retrieving information are represented in this study. Different indexing approaches for minimizing search space and different searching procedures for retrieving information are described by [2] and also give a rundown of classic IR models.

The way Information retrieval system [2,4,10] works is by taking an input from user in the form of query. Then the system processes all the queries and generate vector embeddings which helps in matching the query with existing collection of documents present in the corpus. It not only finds the relevant document but also sends these documents to user in the decreasing order of relevance based on the rank of the document which helps in identifying how results are better.

This paper has contributed to improving the results generated by the vector space model in this paper by using principal component analysis and factor analysis, which are methods for reducing the dimensionality of the embedding vector and finding the correlation between the embedding vector, which improves the model's performance. The larger the dimension of the

embedding vector, the more likely the search result will be incorrect. Improving this embedding vector is one of the most important contributions that have been made in research. This method was tested on pre-trained models such as word2vec, glovec, and SBERT, and according to findings, similarity scores, document results, and mean average precision all improved.

Word embeddings of a pre-trained models are used in a various application, as well as in the construction of representations for sentences, paragraphs, and documents. Recently, most of the researchers are focusing on improving pretrained word vectors using post-processing algorithms.

Reduced dimensionality of word embeddings is one area where improvements can be made and transforming these vectors into better latent form is one more area where improvements in the results can be observed. [1] presents a novel technique for efficiently combining PCA-based dimensionality reduction with a recently proposed post-processing algorithm to create effective lower-dimensional word embeddings.

Sentence BERT [9,14] is a machine learning based algorithm which uses sentence transformer to generate sentence embedding using Siamese BERT neural network. SBERT can be beneficial for semantic search and semantic textual resemblance. BERT also helps to understand meaning of ambiguous language in text document or a query.

Most of the Information Retrieval System [10,11,12,15] is measured using cosine similarity [4] and Euclidean distance [4]. In Information retrieval system [15] cosine similarity refers to the similarity between any two text vectors of the vector space model [15] and Euclidean distance calculates the distance between the two text vectors. Apart from this mean average precision as a metric to measure performance of the model has been used.

# Neural Models for Information Retrieval (2017)

Neural ranking models for information retrieval (IR) use shallow or deep neural networks to rank search results in response to a query. Traditional learning to rank models employ machine learning techniques over hand-crafted IR features. By contrast, neural models learn representations of language from raw text that can bridge the gap between query and document vocabulary.

Since the turn of the decade, there have been dramatic improvements in performance in computer

vision, speech recognition, and machine translation tasks, witnessed in research and in real-world applications [112]. These breakthroughs were largely fuelled by recent advances in neural network

models, usually with multiple hidden layers, known as deep architectures [8, 49, 81, 103, 112]. Exciting novel applications, such as conversational agents [185, 203], have also emerged, as well

as game-playing agents with human-level performance [147, 180].

Retrieval of information can take many forms. Users can express their information need in the form of

a text query—by typing on a keyboard, by selecting a query suggestion, or by voice recognition—or

the query can be in the form of an image, or in some cases the need can even be implicit. Retrieval can involve ranking existing pieces of content, such as documents or short-text answers, or composing new responses incorporating retrieved information. Both the information need and the retrieved results may use the same modality (e.g., retrieving text documents in response to keyword queries), or different ones (e.g., image search using text queries).

Neural IR refers to the application of shallow or deep neural networks to these retrieval tasks. This

tutorial serves as an introduction to neural methods for ranking documents in response to a query, an important IR task.

Neural models for IR use vector representations of text, and usually contain a large number of parameters that needs to be tuned. ML models with large set of parameters typically require a large quantity of training data [196]. Unlike traditional learning to rank (L2R) approaches that train ML models over a set of hand-crafted features, neural models for IR typically accept the raw text of a query and document as input. Learning suitable representations of text also demands large-scale datasets for training [141]. Therefore, unlike classical IR models, these neural approaches tend to be data-hungry, with performance that improves with more training data.

IR systems must deal with short queries that may contain previously unseen vocabulary, to match

against documents that vary in length, to find relevant documents that may also contain large sections of irrelevant text.

IR systems should learn patterns in query and document text that indicate relevance, even if query and document use different vocabulary, and even if the patterns are task-specific or context-specific.

Fundamentals of text retrieval

We focus on text retrieval in IR, where the user enters a text query and the system returns a ranked

list of search results. Search results may be passages of text or full text documents. The system’s

goal is to rank the user’s preferred search results at the top. This problem is a central one in the IR

literature, with well understood challenges and solutions.

Ad-hoc retrieval Ranked document retrieval is a classic problem in information retrieval, as in the

main task of the Text Retrieval Conference [205], and performed by popular search engines such

as Google, Bing, Baidu, or Yandex.

A large number of IR studies [52, 65, 70, 84, 92, 93, 106, 144] have demonstrated that users of retrieval systems tend to pay attention mostly to top-ranked results. IR metrics, therefore, focus on

rank-based comparisons of the retrieved result set R to an ideal ranking of documents, as determined

by manual judgments or implicit feedback from user behaviour data. These metrics are typically

computed at a rank position, say k, and then averaged over all queries in the test set.

Precision and recall both compute the fraction of relevant documents retrieved for a query q, but with respect to the total number of documents in the retrieved set Rq and the total number of relevant documents in the collection D, respectively. Both metrics assume that the relevance labels are binary.

Traditional IR models

TF-IDF There is a broad family of statistical functions in IR that consider the number of occurrences

of each query term in the document (term-frequency) and the corresponding inverse document

frequency of the same terms in the full collection (as an indicator of the informativeness of the term).

One theoretical basis for such formulations is the probabilistic model of IR that yielded the popular

BM25 [166] ranking function.

Language modelling (LM) In the language modelling based approach [79, 161, 230], documents are ranked by the posterior probability *p(d|q)*. where, ^ p(E ) is the maximum likelihood estimate (MLE) of the probability of event E . p(q|d) indicates the probability of generating query q by randomly sampling terms from document d. For smoothing, terms are sampled from both the document d and the full collection D—the two events are treated as mutually exclusive, and their probability is given by λ and (1 - λ ), respectively.

Both TF-IDF and language modelling based approaches estimate document relevance based on the

count of only the query terms in the document. The position of these occurrences and the relationship with other terms in the document are ignored.

Learning to rank (L2R)

In learning to rank, a query-document pair is represented by a vector of numerical features

and a model f : R is trained that maps the feature vector to a real-valued score. The training

dataset for the model consists of a set of queries and a set of documents per query. Depending on

the flavour of L2R, in addition to the feature vector, each query-document pair in the training data is

augmented with some relevance information.

Many machine learning models—including support vector machines, neural networks, and boosted

decision trees—have been employed over the years for the learning to rank task, and a correspondingly large number of different loss functions have been explored.

Examples of different neural approaches to IR. In (a) and (b) the neural network is only used at the point of matching, whereas in **(c) the focus is on learning effective representations of text using neural methods**. Neural models can also be used to expand or augment the query before applying traditional IR techniques, as shown in (d).

(c) Learning query and document representations for matching (e.g., [88, 143])

Many (shallow and deep) neural IR models depend on learning good low-dimensional vector representations—or embeddings—of query and document text, and using them within traditional IR models or in conjunction with simple similarity metrics (e.g., cosine similarity). These models shown in Figure 5c may learn the embeddings by optimizing directly for the IR task (e.g., [88]), or separately in an unsupervised fashion (e.g., [143]).

When the vectors are high-dimensional, sparse, and based on distributional feature they are referred

to as explicit vector representations [117]. On the other hand, when the vectors are dense, small, and learnt from data then they are commonly referred to as embeddings. For both explicit and embedding based representations several distance metrics can be used to define similarity between terms, although cosine similarity is commonly used.

Embeddings

An embedding is a representation of items in a new space such that the properties of, and the

relationships between, the items are preserved. Goodfellow et al. [64] articulate that the goal of

an embedding is to generate a simpler representation—where simplification may mean a reduction

in the number of dimensions, an increase in the sparseness of the representation, disentangling the

principle components of the vector space, or a combination of these goals. In the context of term

embeddings, the explicit feature vectors—like those we discussed in Section 4.3—constitutes the

original representation. An embedding trained from these features assimilate the properties of the

terms and the inter-term relationships observable in the original feature space.

The most popular approaches for learning embeddings include either factorizing the term-feature

matrix (e.g. LSA [48]) or using gradient descent based methods that try to predict the features given

the term (e.g., [15, 134]).

LSA [48] involves performing singular value decomposition(SVD) [63] on a term-document (or term-passage) matrix X to obtain its low-rank approximation [130].

Neural term embedding models are typically trained by setting up a prediction task. Instead of

factorizing the term-feature matrix—as in LSA—neural models are trained to predict the term

from its features. Both the term and the features have one-hot representations in the input and

the output layers, respectively, and the model learns dense low-dimensional representations in the

process of minimizing the prediction error. These approaches are based on the information bottleneck method [197]—discussed in more details in Section 6.2—with the low-dimensional representations acting as the bottleneck. The training data may contain many instances of the same term-feature pair proportional to their frequency in the corpus (e.g., word2vec [134]), or their counts can be pre-aggregated (e.g., GloVe [160])

Word2vec, ….

GloVe, …..

Paragraph2vec, …..

Traditional IR models use local representations of terms for query-document matching. The most

straight-forward use case for term embeddings in IR is to enable inexact matching in the embedding

space.

# Music Information Retrieval J. Stephen Downie (2003)

Why Is MIR Development So Challenging?

Developers and evaluators must constantly take into account the many different ways music can be represented (i.e., the “multirepresentational challenge”). Music transcends time and cultural boundaries, yet each historic epoch, culture, and subculture has created its own unique way of expressing itself musically. This wide variety of expression gives rise to the “multicultural challenge.”

Comprehending and responding to the many different ways individuals interact with music and MIR systems constitutes the “multiexperiential challenge.” Maximizing the benefits of having a multidisciplinary research community while minimizing its inherent drawbacks represents MIR’s “multidisciplinarity challenge.”

# Music Information Retrieval: Recent Developments and Applications (2014)

Broadly speaking, the research field of Music Information Retrieval (MIR) is foremost concerned with the extraction and inference of meaningful features from music(from the audio signal, symbolic representation or external sources such as web pages),indexing of music using these features, and the development of different search and retrieval schemes (for instance, content-based search, music recommendation systems, or user interfaces for browsing large music collections), as defined by Downie [52]. To this end, different representations of music-related subjects (e.g., songwriters, composers, performers, consumer) and items (music pieces, albums, video clips, etc.) are considered.

Whereas early MIR research focused on working with symbolic representations of music pieces (i.e. a structured, digital representation of musical scores such as MIDI), increased computing power enabled the application of the full armory of signal processing techniques directly to the music audio signal during the early 2000s.

As for evaluation, user-centric strategies aim at taking into account different factors in the perception of music qualities, in particular of music similarity. This is particularly important as the notions of music similarity and of music genre (the latter often being used as a proxy for the former) are ill-defined. In fact several authors such as Lippens et al. [157] or Seyerlehner [252] have shown that human agreement on which music pieces belong to a particular genre ranges only between 75% and 80%. Likewise, the agreement among humans on the similarity between two music pieces is also bounded at about 80% as stated in the literature [282, 230, 287, 112].

Music is a highly multimodal human artifact. It can come as audio, symbolic representation (score), text (lyrics), image (photograph of a musician or album cover), gesture (performer) or even only a mental model of a particular tune. Usually, however, it is a mixture of these representations that form an individual’s model of a music entity.

Computational MIR approaches typically use features and create models to describe music by one or more of the following categories of music perception:music content, music context, user properties,and user context, as shown in Figure 1.1 and specified below.

From a general point of view,music content refers to aspects that are encoded in the audio signal, while music context comprises factors that cannot be extracted directly from the audio but are nevertheless related to the music item, artist, or performer. To give some examples, rhythmic structure, melody, and timbre features belong to the former category, whereas information about an artist’s cultural or political background, semantic labels, and album cover artwork belong to the latter.

Please note that there are interconnections between some features from different categories. For instance, aspects reflected in collaborative tags (e.g. musical genre) can be modeled by music content (e.g. instrumentation) while some others (e.g. geographical location, influences) are linked to music context. Another example is semantic labels, which can be used to describe both the mood of a music piece and the emotion of a user.

MIR as a research field is driven by a set of core applications that we present here from a user point of view.

- Music retrieval applications are intended to help users find music in large collections by a particular similarity criterion.

- Music recommendation systems typically propose a list of music pieces based on modeling the user’s musical preferences.

- Automatic music playlist generation, which is sometimes informally called “Automatic DJing”, can be regarded as highly related to music recommendation. Its aim is to create an ordered list of results, such as music tracks or artists, to provide meaningful playlists enjoyable by the listener.

We have seen that research on MIR comprises a rich and diverse set of areas whose scope goes well beyond mere retrieval of documents ….

MIR researchers have then been focusing on a set of concrete research tasks, which are the basis for

final applications.

A first group of topics are related to the extraction of meaningful features from music content and context. These features are then used to compute similarity between two musical pieces or to classify

music pieces according to different criteria (e.g. mood, instrument, or genre). Features, similarity algorithms and classification methods are then tailored to different applications as described below.

Tasks

FEATURE EXTRACTION, SIMILARITY, CLASSIFICATION,

## Semi-supervised music tagging transformer (2021)

Transformer [21, 22] has shown its ability in sequence modeling, establishing itself de facto state-of-the-art in natural language processing.

# Natural Language Processing for MIR

NLP is a field of Computer Science and Artificial Intelligence concerned with the interaction between computers and human (natural) language. NLP is a core component in daily life technologies: web search, speech recognition and synthesis, automatic summaries in the web, product (including music) recommendation, machine translation.

Identify relations holding between words or phrases in the sentence, and what is their function. By analyzing sentence structure, we understand the underlying meaning in a sentence.

Lots of unstructured information about music in the form of natural language.

Potential for improving MIR and musicological resources by integrating automatically acquired knowledge via Natural Language Processing.

The extraction of high level semantic representations from text have been shown useful in different MIR problems.

There is a need of development of new methodologies that exploit these semantic representations in MIR.

Word Embeddings and Deep Learning opens a new world of already unexploited possibilities for multimodal approaches.

# MusCaps: Generating Captions for Music Audio (2021)

In this work, we propose to address music description via audio captioning, defined as the task of generating a natural language description of music audio content in a human-like manner. To this end, we present the first music audio captioning model, MusCaps, consisting of an encoder-decoder with temporal attention.

Our model represents a shift away from classification-based music description and combines tasks requiring both auditory and linguistic understanding to bridge the semantic gap in music information retrieval.

Current music information retrieval (MIR) approaches to music description typically rely on single- or multi-label classification. A prominent example is music auto-tagging [1]–[3], in which descriptive keywords are assigned to a music clip so as to convey high-level characteristics of the input such as

genre, instrumentation and emotion.

It is in fact through natural language that we often query music collections and search for known and unknown music content.

While a wealth of music information is encoded in text, research in machine listening and MIR has traditionally overlooked the relationship between audio and natural language.

Captioning systems not only need to recognise signal-level features such as instrumentation and high-level descriptors such as genre, but they must also encode the relationship between them, thus

better capturing the nuances of musical content; they also produce fully formed, descriptive sentences, that more closely match human queries.

Through its joint use and processing of audio and linguistic information, music captioning also provides a first step towards the development of audio-and-language models for music understanding.

Finally, music captioning has several useful applications, such as producing descriptions for items in large music catalogues or vast collections of amateur and user-generated content; automatically generating evocative descriptions of music in films and videos for deaf and hard-of-hearing people; enabling search and discovery of music through more human-like queries; and providing explanations for automatic music recommendations.

To the best of our knowledge, this is the first work on music audio captioning. In the absence of benchmark datasets and established research on this task, we build upon previous literature on image and audio captioning and compare our model performance to a baseline sequence-to-sequence model. Differently from pioneering work on neural architectures for audio captioning, typically composed of an audio encoder and a text decoder, we propose a multimodal encoder that learns a joint representation of both audio and text to better account for the need to capture high-level semantics and summarise information that emerges at different levels of granularity in the input.

Audio & Language

While multimodal tasks have long been studied in the visual domain, audio-and-language research has only recently started to emerge, with the first audio captioning model proposed in [14]. Following a similar development to its visual counterpart, audio captioning has seen a rapid progress over the last few years [15]–[22], greatly encouraged by the recently introduced DCASE Challenge dedicated to the task.

Most prior audio captioning methods make use of encoder-decoder models, frequently including sequence modelling modules, such as RNNs [20] or variants like gated recurrent units (GRU) [14], [17]

and long short-term memory (LSTM) networks [18], in their encoder to take care of the temporal structure of audio inputs. Most of these works also make use of attention mechanisms to align the audio and text modalities [14], [15], [18], [21]. More recently, following the success of self-attention in V&L models, a small body of work has also started exploring the use of Transformer-based models in audio captioning [19], [23].

Our work is inspired by CNN-RNN architectures developed for image and audio captioning, but focusses on how such approaches can be extended to the music domain for the first time. The only prior works that attempt a similar goal can be found in [24] and [25]. However, the method proposed in [24] fails to generate grammatically correct sentences, while [25] simplifies the task by reframing it as the generation of a sequence of tags. Similarly, prior work on audio-text representation learning also makes use of tags [26], while we stress that our approach focusses on natural language.

# Music autotagging as captioning (2020)

Several papers have explored the co-occurrence relationships between tags: Miotto et al. (2010)

present one of the early works that explicitly used tag co-occurrence modeled by a Dirichlet mixture.

Shao et al. (2018) modeled the tag co-occurrence pattern of a song via Latent Music Semantic Analysis (LMSA). Larochelle et al. (2012); Mandel et al. (2010, 2011a,b) utilized tags alone to build a conditional restricted boltzmann machine and hence demonstrated the value of tag-tag relationships in predicting tags.

Recent works such as (Choi et al., 2018) discussed the effect of tags from the perspective of mislabeling under the theme of multi-label classification.

Ikawa and Kashino (2019) used a captioning model to describe environmental audio. They proposed an extension to the standard sequence-to-sequence model in the captioning task by adding a controllable parameter, specifying the amount of context to provide in the caption.

### Vec4IR Word embeddings for information retrieval.

<https://github.com/lgalke/vec4ir>

### Information Extraction from Natural Language on the Web using LLMs and Iterative Set Expansion

<https://github.com/yaminivibha/LLM_InformationRetrieval>

### gensim – Topic Modelling in Python

<https://github.com/piskvorky/gensim>

Gensim is a Python library for topic modelling, document indexing and similarity retrieval with large corpora. Target audience is the natural language processing (NLP) and information retrieval (IR) community.

### txtai is an all-in-one embeddings database for semantic search, LLM orchestration and language model workflows

<https://github.com/neuml/txtai>

### LLM Embedder

LLM Embedder is fine-tuned based on the feedback from LLMs. It can support the retrieval augmentation needs of large language models, including knowledge retrieval, memory retrieval, examplar retrieval, and tool retrieval. It is fine-tuned over 6 tasks: Question Answering, Conversational Search, Long Conversation, Long-Range Language Modeling, In-Context Learning, and Tool Learning. For more details please refer to https://github.com/FlagOpen/FlagEmbedding/llm\_embedder/README.md

<https://github.com/FlagOpen/FlagEmbedding/tree/master/FlagEmbedding/llm_embedder>

### BGE Embedding

BGE embedding is a general Embedding Model. We pre-train the models using retromae and train them on large-scale pair data using contrastive learning. You can fine-tune the embedding model on your data following our examples. We also provide a pre-train example. Note that the goal of pre-training is to reconstruct the text, and the pre-trained model cannot be used for similarity calculation directly, it needs to be fine-tuned. For more training details for bge see <https://github.com/FlagOpen/FlagEmbedding/blob/master/FlagEmbedding/baai_general_embedding/README.md> .

<https://github.com/FlagOpen/FlagEmbedding/tree/master/FlagEmbedding/baai_general_embedding>

### One Embedder, Any Task: Instruction-Finetuned Text Embeddings

We introduce Instructor, an instruction-finetuned text embedding model that can generate text embeddings tailored to any task (e.g., classification, retrieval, clustering, text evaluation, etc.) and domains (e.g., science, finance, etc.) by simply providing the task instruction, without any finetuning. Instructor achieves sota on 70 diverse embedding tasks!

<https://github.com/xlang-ai/instructor-embedding>

### COVID19-document-retrieval-with-BERT

This project is about developing a document retrieval system to return titles and the context of scientific papers containing the answer to a given user question. We will be using CORD-19 CORD-19 dataset for the dataset which is the biggest corpus of academic papers about COVID-19 and related coronavirus research. In this project, we will use and evaluate 4 different methods for creating contextual sentence embeddings. These embeddings then will be used to find the most semantically meaningful document that relates to our question.

<https://github.com/AGiannoutsos/COVID19-document-retrieval-with-BERT>

### Document-Retrieval-and-Question-Answering-with-BERT

In first part of this repository we developed, with 2 different techniques, a Document Retrieval System, which aims to return titles of scientific papers containing the answer to a given user question. To achieve the goal of this exercise, we first read Sentence Embeddings using Siamese BERT-Networks, in order to understand how to create sentence embeddings. We decided to transform sentences to embeddings, with the help of SBERT Library. So the main concept behind our retrieval system was to compare sentence embeddings (question's and possible answer's embedding) by cosine similarity.

<https://github.com/spyros-briakos/Document-Retrieval-and-Question-Answering-with-BERT>