FORMALIZING THE PROBLEM OF MUSIC DESCRIPTION

Bob L. Sturm
Aalborg University
Denmark
bst@create.aau.dk

Rolf Bardeli
Fraunhofer IAIS
Germany
rolf.bardeli@iais.fraunhofer.de

Thibault Langlois
Lisbon University
Portugal
tl@di.fc.ul.pt

Valentin Emiya
Aix-Marseille Université
CNRS UMR 7279 LIF
valentin.emiya@lif.univ-mrs.fr

ABSTRACT

The lack of a formalism for "the problem of music description" results in, among other things: ambiguity in what problem a music description system must address, how it should be evaluated, what criteria define its success, and the paradox that a music description system can reproduce the "ground truth" of a music dataset without attending to the music it contains. To address these issues, we formalize the problem of music description such that all elements of an instance of it are made explicit. This can thus inform the building of a system, and how it should be evaluated in a meaningful way. We provide illustrations of this formalism applied to three examples drawn from the literature.

1. INTRODUCTION

Before one can address a problem with an algorithm (a finite series of well-defined operations that transduce a well-specified input into a well-specified output) one needs to define and decompose that problem in a way that is compatible with the formal nature of algorithms [17]. A very simple example is the problem of adding any two positive integers. Addressing this problem with an algorithm entails defining the entity "positive integer", the function "adding", and then producing a finite series of well-defined operations that applies the function to an input of two positive integers to output the correct positive integer.

A more complex example is "the problem of music description." While much work in music information retrieval (MIR) has proposed systems to attempt to address the problem of music description [4, 12, 29], and much work attempts to evaluate the capacity of these systems for addressing that problem [9, 20], we have yet to find any work that actually *defines* it. (The closest we have found is that of [24].) Instead, there are many allusions to the problem: predict the "genre" of a piece of recorded music [25]; label music with "useful tags" [1]; predict what a listener will "feel" when "listening" to some music [29]; find music "similar" to some other music [26]. These allusions are deceptively simple, however, since behind them lie many problems and questions that have major repercussions on the design and evaluation of any proposed system. For example, What is "genre"? What is "useful"? How is "feeling" related to "listening"? "Similar" in what respects?

With respect to the problem of music description, some work in MIR discusses the meaningfulness, worth, and futility of designing artificial systems to describe music [28]; the idea of and the difficulty in "ground truth" [3, 6, 15]; the size of datasets [5], a lack of statistics [10], the existence of bias [16], and the ways such systems are evaluated [21, 22, 27]. Since a foundational goal of MIR is to develop systems that can imitate the human ability to describe music, these discussions are necessary. However, what remains missing is a formal definition of the problem of music description such that it can be addressed by algorithms, and relevant and valid evaluations can be designed.

In this work, we formalize the problem of music description and try to avoid ambiguity arising from semantics. This leads to a rather abstract form, and so we illustrate its aspects using examples from the literature. The most practical benefit of our formalization is a specification of all elements that should be explicitly defined when addressing an instance of the problem of music description.

2. FORMALISM

We start our formalization by defining the domain of the problem of music description. In particular, we discriminate between the music that is to be described and a recording of it since the former is intangible and the latter is data that a system can analyze. We then define the problem of music description, a recorded music description system (RMDS), and the analysis of such a system. This leads to the central role of the use case.

2.1 Domain

Denote a *music universe*, Ω, a set of music, e.g., Vivaldi's "The Four Seasons", the piano part of Gershwin's "Rhapsody in Blue", and the first few measures of the first movement of Beethoven's Fifth Symphony. A member of Ω is intangible. One cannot hear, see or point to any member of Ω; but one can hear a performance *of* Vivaldi's "The Four Seasons", read sheet music notating the piano part *of* Gershwin's "Rhapsody in Blue", and point to a printed score *of* Beethoven's Fifth Symphony. Likewise, a recorded performance of Vivaldi's "The Four Seasons" is *not* Vivaldi's "The Four Seasons", and sheet music notating the piano part of Gershwin's "Rhapsody in Blue" is *not* the piano part of Gershwin's "Rhapsody in Blue".

In the tangible world, there may exist tangible recordings of the members of Ω. Denote the *tangible music recording universe* by $\mathcal{R}_\Omega$. A member of $\mathcal{R}_\Omega$ is a recording of an element of $\omega \in \Omega$. A *recording* is a tangible object, such as a printed CD or score. Denote one recording of $\omega \in \Omega$ as $r_\omega \in \mathcal{R}_\Omega$. There might be many recordings of an ω in $\mathcal{R}_\Omega$. We say the music ω is *embedded* in r_ω; it enables for a listener an indirect sense of ω. For instance, one can hear a live or recorded performance of ω, and one can read a printed score of ω. The acknowledgment of and distinction between intangible music and tangible recordings *of* music is essential since systems cannot work with intangible music, but only tangible recordings.

2.2 Music Description and the Use Case

Denote a *vocabulary*, $\mathcal{V}$, a set of symbols or tokens, e.g., "Baroque", "piano", "knock knock", scores employing common practice notation, the set of real numbers $\mathbb{R}$, other music recordings, and so on. Define the *semantic universe* as

$$\mathcal{S}_{\mathcal{V},A} := \{s = (v_1, \dots, v_n) | n \in \mathbb{N}, \\ \forall 1 \le i \le n [v_i \in \mathcal{V}] \wedge A(s)\} \quad (1)$$

where $A(\cdot)$ encompasses a semantic rule, for instance, restricting $\mathcal{S}_{\mathcal{V},A}$ to consist of sequences of cardinality 1. Note that the *description* s is a sequence, and not a vector or a set. This permits descriptions that are, e.g., time-dependent, such as envelopes, if $\mathcal{V}$ and $A(\cdot)$ permit it. In that case, the order of elements in s could be alternating time values with envelope values. Descriptions could also be time-frequency dependent.

We define *music description* as pairing an element of Ω or $\mathcal{R}_\Omega$ with an element of $\mathcal{S}_{\mathcal{V},A}$. The *problem of music description* is to make the pairing *acceptable with respect to a use case*. A *use case* provides specifications of Ω and $\mathcal{R}_\Omega$, $\mathcal{V}$ and $A(\cdot)$, and success criteria. *Success criteria* describe how music or a music recording should be paired with an element of the semantic universe, which may involve the sanity of the decision (e.g., tempo estimation must be based on the frequency of onsets), the efficiency of the decision (e.g., pairing must be produced under 100 ms with less than 10 MB of memory), or other considerations.

To make this clearer, consider the following use case. The music universe Ω consists of performances by Buckwheat Zydeco, movements of Vivaldi's "The Four Seasons", and traditional Beijing opera. The tangible music recording universe $\mathcal{R}_\Omega$ consists of all possible 30-second digital audio recordings of the elements in Ω. Let the vocabulary $\mathcal{V} = \{$"Blues", "Classical"$\}$; and define $A(s) := [|s| \in \{0, 1\}]$. The semantic universe is thus, $\mathcal{S}_{\mathcal{V},A} = \{(),$ ("Blues"), ("Classical")$\}$. There are many possible success criteria. One is to map all recordings of Buckwheat Zydeco to "Blues", map all recordings of Vivaldi's "The Four Seasons" to "Classical", and map all recordings of traditional Beijing opera to neither. Another is to map no recordings of Buckwheat Zydeco and Vivaldi's "The Four Seasons" to the empty sequence, and to map any recording of traditional Beijing opera to either non-empty sequence with a probability less than 0.1.

2.3 Recorded Music Description Systems

A *recorded music description system* (RMDS) is a map from the tangible music recording universe to the semantic universe:

$$\mathcal{S} : \mathcal{R}_\Omega \to \mathcal{S}_{\mathcal{V},A}. \quad (2)$$

Building an RMDS means making a map according to well-specified criteria, e.g., using expert domain knowledge, automatic methods of supervised learning, and a combination of these. An instance of an RMDS is a specific map that is already built, and consists of four kinds of components [21]: algorithmic (e.g., feature extraction, classification, pre-processing), instruction (e.g., description of $\mathcal{R}_\Omega$ and $\mathcal{S}_{\mathcal{V},A}$), operator(s) (e.g., the one inputting data and interpreting output), and environmental (e.g., connections between components, training datasets). It is important to note that $\mathcal{S}$ is not restricted to map any recording to a single element of $\mathcal{V}$. Depending on $\mathcal{V}$ and $A(\cdot)$, $\mathcal{S}_{\mathcal{V},A}$ could consist of sequences of scalars and vectors, sets and sequences, functions, combinations of all these, and so on. $\mathcal{S}$ could thus map a recording to many elements of $\mathcal{V}$.

One algorithmic component of an RMDS is a *feature extraction algorithm*, which we define as

$$\mathcal{E} : \mathcal{R}_\Omega \to \mathcal{S}_{\mathbb{F},A'} \quad (3)$$

i.e., a map from $\mathcal{R}_\Omega$ to a semantic universe built from the vocabulary of a feature space $\mathbb{F}$ and semantic rule $A'(\cdot)$. For instance, if $\mathbb{F} := \mathbb{C}^M$, $M \in \mathbb{N}$, and $A'(s) := [|s| = 1]$, then the feature extraction maps a recording to a single M-dimensional complex vector. Examples of such a map are the discrete Fourier transform, or a stacked series of vectors of statistics of Mel frequency cepstral coefficients. Another algorithmic component of an RMDS is a *classification algorithm*, which we define:

$$\mathcal{C} : \mathcal{S}_{\mathbb{F},A'} \to \mathcal{S}_{\mathcal{V},A} \quad (4)$$

i.e., a map from one semantic universe to another. Examples of such a map are k-nearest neighbor, maximum likelihood, support vector machine, and a decision tree.

To make this clearer, consider the RMDS named "RT GS" built by Tzanetakis and Cook [25]. $\mathcal{E}$ maps sampled audio signals of about 30-s duration to $\mathcal{S}_{\mathbb{F},A'}$, defined by single 19-dimensional vectors, where one dimension is spectral centroid mean, another is spectral centroid variance, and so on. $\mathcal{C}$ maps $\mathcal{S}_{\mathbb{F},A'}$ to $\mathcal{S}_{\mathcal{V},A}$, which is defined by $\mathcal{V} = \{$"Blues", "Classical", "Country", "Disco", "Hip hop", "Jazz", "Metal", "Pop", "Reggae", "Rock"$\}$, and $A(s) := [|s| = 1]$. This mapping involves maximizing the likelihood of an element of $\mathcal{S}_{\mathbb{F},A'}$ among ten multivariate Gaussian models created with supervised learning.

Supervised learning involves automatically building components of an $\mathcal{S}$, or defining $\mathcal{E}$ and $\mathcal{C}$, given a *training recorded music dataset*: a sequence of tuples of recordings sampled from $\mathcal{R}_\Omega$ and elements of $\mathcal{S}_{\mathcal{V},A}$, i.e.,

$$\mathcal{D} := \{(r_i, s_i) \in R_\Omega \times \mathcal{S}_{\mathcal{V},A} | i \in \mathcal{I}\} \quad (5)$$

The set $\mathcal{I}$ indexes the dataset. We call the sequence $(s_i)_{i \in \mathcal{I}}$ the *ground truth* of $\mathcal{D}$. In the case of RT GS, its training recorded music dataset contains 900 tuples randomly selected from the dataset *GTZAN* [22,25]. These are selected in a way such that the ground truth of $\mathcal{D}$ has no more than 100 of each element of $\mathcal{S}_{\mathcal{V},A}$.

2.4 Analysis of Recorded Music Description Systems

Given an RMDS, one needs to determine whether it addresses the problem of music description. Simple questions to answer are: does Ω and $\mathcal{R}_\Omega$ of the RMDS encompass those of the use case? Does the $\mathcal{S}_{\mathcal{V},A}$ of the RMDS encompass that of the use case? A more complex question could be, does the RMDS meet the success criteria of the use case? This last question involves the design, implementation, analysis, and interpretation of valid experiments that are relevant to answering hypotheses about the RMDS and success criteria [21, 27]. Answering these questions constitutes an *analysis* of an RMDS.

Absent explicit success criteria of a use case, a standard approach for evaluating an RMDS is to compute a variety of *figures of merit* (FoM) from its "treatment" of the recordings of a testing $\mathcal{D}$ that exemplify the input/output relationships sought. Examples of such FoM are mean classification accuracy, precisions, recalls, and confusions. An implicit belief is that the correct output will be produced from the input only if an RMDS has learned criteria relevant to describing the music. Furthermore, it is hoped that the resulting FoM reflect the real world performance of an RMDS. The *real world performance* of an RMDS are the FoM that result from an experiment using a testing recording music dataset consisting of *all* members in $\mathcal{R}_\Omega$, rather than a sampling of them. If this dataset is out of reach, statistical tests can be used to determine significant differences in performance between two RMDS (testing the null hypothesis, "neither RMDS has 'learned better' than the other"), or between the RMDS and that of picking an element of $\mathcal{S}_{\mathcal{V},A}$ independent of the element from $\mathcal{R}_\Omega$ (testing the null hypothesis, "The RMDS has learned nothing"). These statistical tests are accompanied by implicit and strict assumptions on the measurement model and its appropriateness to describe the measurements made in the experiment [2, 8].

As an example, consider the evaluation of RT GS discussed above [25]. The evaluation constructs a testing $\mathcal{D}$ from the 100 elements of the dataset *GTZAN* not present in the training $\mathcal{D}$ used to create the RMDS. They treat each of the 100 recordings in the testing $\mathcal{D}$ with RT GS, and compare its output with the ground truth. From these 100 comparisons, they compute the percentage of outputs that match the ground truth (accuracy). Whether or not this is a high-quality estimate of the real world accuracy of RT GS depends entirely upon the definition of Ω, $\mathcal{R}_\Omega$, $\mathcal{S}_{\mathcal{V},A}$, as well as the testing $\mathcal{D}$ and the measurement model of the experiment.

There are many serious dangers to the interpretation of the FoM of an RMDS as reflective of its real world performance: noise in the measurements, an inappropriate measurement model [2], a poor experimental design and errors of the third kind [14], the lack of error bounds or error bounds that are too large [8], and several kinds of bias. One kind of bias comes from the very construction of testing datasets. For instance, if the testing dataset is the same as the training dataset, and the set of recordings in the dataset is a subset of $\mathcal{R}_\Omega$, then the FoM of an RMDS computed from the treatment may not indicate its real world performance. This has led to the prescription in machine learning to use a testing dataset that is disjoint with the training dataset, by partitioning for instance [13]. This, however, may not solve many other problems of bias associated with the construction of datasets, or increase the relevance of such an experiment with measuring the extent to which an RMDS has learned to describe the music in Ω.

2.5 Summary

Table 1 summarizes all elements defined in our formalization of the problem of music description, along with examples of them. These are the elements that must be explicitly defined in order to address an instance of the problem of music description by algorithms. Central to many of these are the definition of a use case, which specifies the music and music recording universe, the vocabulary, the desired semantic universe, *and* the success criteria of an acceptable system. (Note that "use case" is not the same as "user-centered.") If the use case is not unambiguously specified, then a successful RMDS cannot be constructed, relevant and valid experiments cannot be designed, and the analysis of an RMDS cannot be meaningful. Table 1 can serve as a checklist for the extent to which an instance of the problem of music description is explicitly defined.

3. APPLICATION

We now discuss two additional published works in the MIR literature in terms of our formalism.

3.1 Dannenberg et al. [7]

The use cases of the RMDS employed by Dannenberg et al. [7] are motivated by the desire for a mode of communication between a human music performer and an accompanying computer that is more natural than physical interaction. The idea is for the computer to employ an RMDS to describe the acoustic performance of a performer in terms of several "styles." Dannenberg et al. circumvent the need to define any of these "styles" by noting, "what really matters is the ability of the performer to consistently produce intentional and different styles of playing at will" [7]. As a consequence, the use cases and thus system analysis are centered on the performer.

One use case considered by Dannenberg et al. defines $\mathcal{V} = \{$"lyrical", "frantic", "syncopated", "pointillistic", "blues", "quote", "high", "low"$\}$, and the semantic rule $A(s) := [|s| \in \{1\}]$. The semantic universe $\mathcal{S}_{\mathcal{V},A}$ is then all single elements of $\mathcal{V}$. The music universe Ω is all possible music that can be played or improvised by the specific performer in these "styles." The tangible music recording universe $\mathcal{R}_\Omega$ is all possible 5-second acoustic recordings of the elements of Ω. Finally, the success criteria of this particular problem of music description includes the following requirements: reliable for a specific performer in an interactive performance, classifier latency of under

Element (Symbol)	**Definition**	**Example**
music universe (Ω)	a set of (intangible) music	{“Automatic Writing” by R. Ashley}
tangible music recording universe ($\mathcal{R}_\Omega$)	a set of tangible recordings of all members of Ω	{R. Ashley, “Automatic Writing”, LCD 1002, Lovely Music, Ltd., 1996}
recording (r_ω)	a member of $\mathcal{R}_\Omega$	a 1-second excerpt of the 46 minute recording of “Automatic Writing” from LCD 1002
vocabulary ($\mathcal{V}$)	a set of symbols	{“Robert”, “french woman”, “bass in other room”, “Moog”} $\cup [0, 2760]$
semantic universe ($\mathcal{S}_{\mathcal{V},A}$)	$\{s = (v_1, \ldots, v_n) \mid n \in \mathbb{N}, \forall 1 \leq i \leq n[v_i \in \mathcal{V}] \wedge A(s)\}$, i.e., the set of all sequences of symbols from $\mathcal{V}$ permitted by the semantic rule $A(\cdot)$	{(“Robert”, 1), (“Robert”, “Moog”, 4.3), (“french woman”, 104.3), (“french woman”, “Moog”, 459), ...}
semantic rule ($A(s)$)	a Boolean function that defines when sequence s is “permissible”	$A(s) := \big[(\lvert s\rvert \in \{2,3,4,5\}) \wedge (\{v_1, \ldots, v_{\lvert s\rvert-1}\} \subseteq \{$“Robert”, “french woman”, “bass in other room”, “Moog”$\} \cup \{\}) \wedge (v_{\lvert s\rvert} \in [0, 2760])\big]$
music description	the pairing of an element of Ω or $\mathcal{R}_\Omega$ with an element of $\mathcal{S}_{\mathcal{V},A}$	label the events (character, time) in recording LCD 1002 of “Automatic Writing” by R. Ashley
the problem of music description	make this pairing acceptable with respect to the success criteria specified by the use case	make this pairing such that F-score of event “Robert” is at least 0.9
use case	specification of $\Omega, \mathcal{R}_\Omega, \mathcal{V}, A(s)$, and success criteria	see all above
system	a connected set of interacting and interdependent components of four kinds (operator(s), instructions, algorithms, environment) that together address a use case	system created in the Audio Latin Genre Classification task of MIREX 2013 by organizer from submission “AP1” and fold 1 of LMD [18]
operators	agent(s) that employ the system, inputting data, and interpreting outputs	Audio Latin Genre Classification organizer of MIREX 2013
instructions	specifications for the operator(s), like an application programming interface	MIREX 2013 input/output specifications for Train/Test tasks; “README” file included with “AP1”
algorithm	a finite series of well-defined ordered operations to transduce an input into an output	“Training.m” and “Classifying.m” MATLAB scripts in “AP1”, etc.
environment	connections between components, external databases, the space within which the system operates, its boundaries	folds 2 and 3 of LMD [18], MIREX computer cluster, local MATLAB license file, etc.
recorded music description system (RMDS) ($\mathcal{S}$)	$\mathcal{S} : \mathcal{R}_\Omega \rightarrow \mathcal{S}_{\mathcal{V},A}$, i.e., a map from $\mathcal{R}_\Omega$ to $\mathcal{S}_{\mathcal{V},A}$	“RT GS” evaluated in [25]
feature extraction algorithm ($\mathcal{E}$)	$\mathcal{E} : \mathcal{R}_\Omega \rightarrow \mathcal{S}_{\mathbb{F},A'}$, i.e., a map from $\mathcal{R}_\Omega$ to an element of a semantic universe based on the feature vocabulary $\mathbb{F}$ and semantic rule $A'(s)$	compute using [19] the first 13 MFCCs (including zeroth coefficient) from a recording
feature vocabulary ($\mathbb{F}$)	a set of symbols	$\mathbb{R}^{13}$
classification algorithm ($\mathcal{C}$)	$\mathcal{C} : \mathcal{S}_{\mathbb{F},A'} \rightarrow \mathcal{S}_{\mathcal{V},A}$, i.e., a map from $\mathcal{S}_{\mathbb{F},A'}$ to the semantic universe	single nearest neighbor
recorded music dataset	$\mathcal{D} := (\{r_\omega \in \mathcal{R}_\Omega, s \in \mathcal{S}_{\mathcal{V},A}\}_i)_{i \in \mathcal{I}}$, i.e., a sequence of tuples of recordings and elements of the semantic universe, indexed by $\mathcal{I}$	*GTZAN* [22, 25]
“ground truth” of $\mathcal{D}$	$(s_i)_{i \in \mathcal{I}}$, i.e., the sequence of “true” elements of the semantic universe for the recordings in $\mathcal{D}$	in *GTZAN*: {“blues”, “blues”, ... , “classical”, ... , “country”, ...}
analysis of an RMDS	answering whether an RMDS can meet the success criteria of a use case with relevant and valid experiments	designing, implementing, analyzing and interpreting experiments that validly answer, “Can RT GS [25] address the needs of user A?”
experiment	principally in service to answering a scientific question, the mapping of one or more RMDS to recordings of $\mathcal{D}$, and the making of measurements	apply RT GS to *GTZAN*, compare its output labels to “ground truth”, and compute accuracy
figure of merit (FoM)	performance measurement of an RMDS from an experiment	classification accuracy of RT GS in *GTZAN*
real world performance of an RMDS	the figure of merit expected if an experiment with an RMDS uses all of $\mathcal{R}_\Omega$	classification accuracy of RT GS

Table 1. Summary of all elements defined in the formalization of the problem of music description, with examples.

5 seconds. The specific definition of “reliable” might include high accuracy, high precision in every class, or only in some classes.

Dannenberg et al. create an RMDS by using a training dataset of recordings curated from actual performances, as well as collected in a more controlled fashion in a laboratory. The ground truth of the dataset is created with input from performers. The feature extraction algorithm includes algorithms for pitch detection, MIDI conversion, and the computation of 13 low-level features from the MIDI data. One classification algorithm employed is maximum likelihood using a naive Bayesian model.

The system analysis performed by Dannenberg et al. involve experiments measuring the mean accuracy of all systems created and tested with 5-fold cross validation. Furthermore, they evaluate a specific RMDS they create in the context of a live music performance. From this they observe three things: 1) the execution time of the RMDS is under 1 ms; 2) the FoM of the RMDS found in the laboratory evaluation is too optimistic for its real world performance in the context of live performance; 3) using the confidence of the classifier and tuning a threshold parameter provides a means to improve the RMDS by reducing its number of false positives.

3.2 Turnbull et al. [24]

Turnbull et al. [24] propose several RMDS that work with a vocabulary consisting of 174 unique "musically relevant" words, such as "Genre–Brit_Pop", "Usage-Reading", and "NOT-Emotion–Bizarre_/_Weird". $A(s) := [|s| = 10 \wedge \forall i \neq j(v_i \neq v_j)]$, and so the elements of $\mathcal{S}_{\mathcal{V},A}$ are tuples of ten unique elements of $\mathcal{V}$. The music universe Ω consists of at least 502 songs (the size of the CAL500 dataset), such as "S.O.S." performed by ABBA, "Sweet Home Alabama" performed by Lynyrd Skynyrd, and "Fly Me to the Moon" sung by Frank Sinatra. The tangible music recording universe $\mathcal{R}_\Omega$ is composed of MP3-compressed recordings of entire music pieces. The RMDS sought by Turnbull et al. aims "[to be] good at predicting all the words [in $\mathcal{V}$]", or "produce sensible semantic annotations for an acoustically diverse set of songs." Since "good", "sensible" and "acoustically diverse" are not defined, the success criteria is ambiguous. Ω is also likely much larger than 502 songs.

The feature extraction algorithm in the RMDS of Turnbull et al. maps a music recording to a semantic universe built from a feature vocabulary $\mathbb{F} := \mathbb{R}^{39}$, and the semantic rule $A'(s) := [|s| = 10000]$. That is, the algorithm computes from an audio recording 13 MFCC coefficients on 23ms frames, concatenates the first and second derivatives in each frame, and randomly selects 10000 feature vectors from all those extracted. The classification algorithm in the RMDS uses a a maximum a posteriori decision criterion, with conditional probabilities of features modelled by a Gaussian mixture model (GMM) of a specified order. One RMDS uses expectation maximization to estimate the parameters of an 8-order GMM from a training dataset.

Turnbull et al. build an RMDS using a training dataset of 450 elements selected from CAL500. They apply this RMDS to the remaining elements of CAL500, and measure how its output compares to the ground truth. When the ground truth of a recording in CAL500 does not have 10 elements per the semantic rule of the semantic universe, Turnbull et al. randomly add unique elements of $\mathcal{V}$, or randomly remove elements from the ground truth of the recording until it has cardinality 10.

Turnbull et al. compute from an experiment FoM such as mean per-word precision. Per-word precision is, for a $v \in \mathcal{V}$ and when defined, the percentage of correct mappings of the system from the recordings in the test dataset to an element of the semantic universe that includes v. Mean per-word precision is thus the mean of the $|\mathcal{V}|$ per-word precisions. Turnbull et al. compare the FoM of the RMDS to other systems, such as a random classifier and a human. They conclude that their best RMDS is slightly worse than human performance on "more 'objective' semantic categories [like instrumentation and genre]" [24]. The evaluation, measuring the amount of ground truth reproduced by a system (human or not) and not the sensibility of the annotations, has questionable relevance and validity to the ambiguous use case.

4. CONCLUSION

Formalism can reveal when a problem is not adequately defined, and how to explicitly define it in no uncertain terms. An explicit definition of a problem shows how to evaluate solutions in relevant and valid ways. It is in this direction that we move with this paper for the problem of music description, the spirit of which is encapsulated by Table 1. The unambiguous definition of the use case is central for addressing an instance of the problem of music description.

We have discussed several published RMDS within this formalism. The work of Dannenberg et al. [7] provides a good model since its use case and analysis are clearly specified — both center on a specific music performer — and through evaluating the system in the real world they actually complete the research and development cycle to improve the system [27]. The use cases of the RMDS built by Tzanetakis and Cook [25] and Turnbull et al. [24] are not specified. In both cases, a labeled dataset is assumed to provide sufficient definition of the problem. Turnbull et al. suggest a success criterion of annotations being "sensible," but the evaluation only measures the amount of ground truth reproduced. Due to the lack of definition, we are thus unsure what problem either of these RMDS is actually addressing, or whether either of them is actually considering the music [23]. An analysis of an RMDS depends on an explicit use case. The definition of the use case in Dannenberg et al. [7] renders this question irrelevant: all that is needed is that the RMDS meets the success criteria of a given performer, which is tested by performing with it.

While we provide in this paper a formalization of the problem of music description, and a checklist of the components necessary to define an instance of such a problem, it does not describe how to solve any specific problem of music description. We do not derive restrictions on any of the components of the problem definition, or show how datasets should be constructed to guarantee an evaluation can result in good estimates of real world performance. Our future work aims in these directions. We will incoprorate the formalism of the design and analysis of comparative experiments [2,21], which will help define the notions of relevance and validity when it comes to analyzing RMDS. We seek to incorporate notions of learning and inference [13], e.g., to specify what constitutes the building of a "good" RMDS using a training dataset (where "good" depends on the use case). We also seek to explain more formally two paradoxes that have been observed. First, though an RMDS is evaluated in a test dataset to reproduce a large amount of ground truth, it appears to not be a result of the consideration of characteristics in the music universe [20]. Second, though artificial algorithms have none of the extensive experience humans have in music listening, description, and culture, they can reproduce ground truth consisting of extremely subjective and culturally centered concepts like genre [11].

5. ACKNOWLEDGMENTS

The work of BLS and VE is supported in part by l'Institut français du Danemark, and ARCHIMEDE Labex (ANR-11-LABX- 0033).

6. REFERENCES

[1] J.-J. Aucouturier and E. Pampalk. Introduction – from genres to tags: A little epistemology of music information retrieval research. *J. New Music Research*, 37(2):87–92, 2008.

[2] R. A. Bailey. *Design of comparative experiments*. Cambridge University Press, 2008.

[3] M. Barthet, G. Fazekas, and M. Sandler. Multidisciplinary perspectives on music emotion recognition: Implications for content and context-based models. In *Proc. CMMR*, 2012.

[4] T. Bertin-Mahieux, D. Eck, and M. Mandel. Automatic tagging of audio: The state-of-the-art. In W. Wang, editor, *Machine Audition: Principles, Algorithms and Systems*. IGI Publishing, 2010.

[5] T. Bertin-Mahieux, D. Ellis, B. Whitman, and P. Lamere. The million song dataset. In *Proc. ISMIR*, 2011.

[6] A. Craft, G. A. Wiggins, and T. Crawford. How many beans make five? The consensus problem in music-genre classification and a new evaluation method for single-genre categorisation systems. In *Proc. ISMIR*, pages 73–76, 2007.

[7] R. B. Dannenberg, B. Thom, and D. Watson. A machine learning approach to musical style recognition. In *Proc. ICMC*, pages 344–347, 1997.

[8] E. R. Dougherty and L. A. Dalton. Scientific knowledge is possible with small-sample classification. *EURASIP J. Bioinformatics and Systems Biology*, 2013:10, 2013.

[9] J. Stephen Downie, Donald Byrd, and Tim Crawford. Ten years of ISMIR: Reflections on challenges and opportunities. In *Proc. ISMIR*, pages 13–18, 2009.

[10] A. Flexer. Statistical evaluation of music information retrieval experiments. *J. New Music Research*, 35(2):113–120, 2006.

[11] J. Frow. *Genre*. Routledge, New York, NY, USA, 2005.

[12] Z. Fu, G. Lu, K. M. Ting, and D. Zhang. A survey of audio-based music classification and annotation. *IEEE Trans. Multimedia*, 13(2):303–319, Apr. 2011.

[13] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag, 2 edition, 2009.

[14] A. W. Kimball. Errors of the third kind in statistical consulting. *J. American Statistical Assoc.*, 52(278):133–142, June 1957.

[15] E. Law, L. von Ahn, R. B. Dannenberg, and M. Crawford. Tagatune: A game for music and sound annotation. In *Proc. ISMIR*, pages 361–364, 2007.

[16] E. Pampalk, A. Flexer, and G. Widmer. Improvements of audio-based music similarity and genre classification. In *Proc. ISMIR*, pages 628–233, Sep. 2005.

[17] R. Sedgewick and K. Wayne. *Algorithms*. Addison-Wesley, Upper Saddle River, NJ, 4 edition, 2011.

[18] C. N. Silla, A. L. Koerich, and C. A. A. Kaestner. The Latin music database. In *Proc. ISMIR*, 2008.

[19] M. Slaney. Auditory toolbox. Technical report, Interval Research Corporation, 1998.

[20] B. L. Sturm. Classification accuracy is not enough: On the evaluation of music genre recognition systems. *J. Intell. Info. Systems*, 41(3):371–406, 2013.

[21] B. L. Sturm. Making explicit the formalism underlying evaluation in music information retrieval research: A look at the MIREX automatic mood classification task. In *Post-proc. Computer Music Modeling and Research*, 2014.

[22] B. L. Sturm. The state of the art ten years after a state of the art: Future research in music information retrieval. *J. New Music Research*, 43(2):147–172, 2014.

[23] B. L. Sturm. A simple method to determine if a music information retrieval system is a "horse". *IEEE Trans. Multimedia*, 2014 (in press).

[24] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet. Semantic annotation and retrieval of music and sound effects. *IEEE Trans. Audio, Speech, Lang. Process.*, 16, 2008.

[25] G. Tzanetakis and P. Cook. Musical genre classification of audio signals. *IEEE Trans. Speech Audio Process.*, 10(5):293–302, July 2002.

[26] J. Urbano. *Evaluation in Audio Music Similarity*. PhD thesis, University Carlos III of Madrid, 2013.

[27] J. Urbano, M. Schedl, and X. Serra. Evaluation in music information retrieval. *J. Intell. Info. Systems*, 41(3):345–369, Dec. 2013.

[28] G. A. Wiggins. Semantic gap?? Schemantic schmap!! Methodological considerations in the scientific study of music. In *Proc. IEEE Int. Symp. Mulitmedia*, pages 477–482, Dec. 2009.

[29] Y.-H. Yang and H. H. Chen. *Music Emotion Recognition*. CRC Press, 2011.