Automated Inference of Musical Sophistication From User Behaviour

Author: Joel Coffey
Supervisor: Prof. Stephen Barrett

A dissertation submitted to the University of Dublin, in partial fulfilment of the requirements for the degree of Master in Computer Science.

Submitted to The University of Dublin, Trinity College, May, 2014

Declaration

I, Joel Coffey, declare that the following dissertation, except where otherwise stated, is entirely my own work; that it has not previously been submitted as an exercise for a degree, either in Trinity College Dublin, or in any other University; and that the library may lend or copy it or any part thereof on request.

Signature

Date

Acknowledgements

I would like to thank my supervisor, Prof. Stephen Barrett, for his tireless enthusiasm and insightful advice. I also wish to thank my parents for their unquestioning support. And finally Marina, whose belief and encouragement were a source of constant inspiration.

Summary

Recommender systems are widely used across many domains with the goal of providing timely, accessible decision making support to users. Collaborative filtering is a class of recommender system that exploits similarities between users and items in order to select suitable items for recommendation. In a music application, a reliance on listening or rating co-occurrence as a means of assessing user similarity can result in loss of diversity and novelty in the result set. This research proposes an alternative similarity metric, unreliant on co-occurrence, with the aim of increasing diversity, or serendipity, in such applications.

Musical sophistication is a prevalent and well researched concept in the field of music psychology. Working from the assumption that this characteristic is a significant determinant in musical taste, this research describes a methodology by which it can be inferred from user interaction with a music software application. It is proposed that this indicator be utilised as an alternative, or complementary, similarity measure, thereby mitigating the aforementioned homogeneity exhibited by some music recommender systems.

The development of a model of musical sophistication is described, as it pertains to the behavioural tendencies of a user of a music software application. The model is derived from a knowledge base that takes the form of a set of natural language propositions. A primary goal of the research is the development of a methodology by which a knowledge base such as this, provided by a domain expert with little technical expertise, can be populated and evaluated given a chronological log of user activity. In this sense the methodology described is not specific to the context of the current work, but is presented as a generalisable means of capturing user characteristics from the information latent in a detailed interaction log.

The natural language knowledge base is transformed into an instance of a computational argumentation framework using techniques informed by both fuzzy logic and defeasible reasoning. The framework consists of a hierarchy of user attributes, and a set of relationships or rules which dictate an aggregation process that ultimately converges on 4 broad characteristics that comprise the top layer of the model. The final musical sophistication score attributed to the user is a weighed arithmetic mean of these characteristics.

The methodology is design-driven and stands in contrast to statistical techniques designed to capture correlations and patterns from pre-existing datasets. A feedback loop from the evaluation stage back to the design stage provides an iterative mechanism by which accuracy can be improved, and omissions and misconceptions in the initial model identified.

A scoped evaluation is performed over an idealised population of exemplar users represented by a set of simulated activity logs. These logs are generated from an open source

music application that was instrumented to emit appropriate metrics as it is used. The system is found to identify these exemplars accurately, however the task of evaluation over a non-idealised population is left for future work, as a use case study is judged beyond the current scope.

Without significant further investment, neither the true efficacy of the derived implementation, nor the potential utility of a recommender system that implements musical sophistication as a similarity rating, can be unequivocally assessed. In acknowledgement of the limitations of the scoped evaluation strategy employed, the design of a large scale use case based study is described and left for future implementation. Nevertheless, satisfactory levels of performance were exhibited within the bounds of the evaluation that was performed, and the proposal is ultimately judged to show some promise.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Recent work has highlighted a focus on accuracy, over diversity and novelty, as a common oversight in the design and evaluation of recommender systems (RS). In response to this perceived deficit, *serendipity* has been proposed as an alternative evaluation metric, to express the degree to which a user is positively surprised by a suggestion (McNee et al., 2006; Murakami et al., 2008).

To illustrate the deficiency of accuracy as the predominant metric of success, consider a music recommender system that recommends exclusively songs by artists already highly rated by the user. Such a system unquestionably exhibits high levels of accuracy. However, it is unlikely to lead to high user satisfaction as the value provided is minimal beyond what can be achieved with standard search facilities.

An approach common to many RSs is the identification of similarities between users and items through the use of graph theory or matrix based analysis of access histories, stated preferences, or both. This research describes a new method to identify user similarity in a music RS which does not rely on listening or rating history.

The *musical sophistication* (MS) of the user is inferred from their interactions with a host application that provides music listening, discovery and management functionality. It is envisioned that suggestions for a given user be drawn from those with similar sophistication levels but low listening or rating co-occurrence. That those suggestions will exhibit greater serendipity than that achievable through a more traditional approach is an assumption that underlies the motivation for this research.

1.2 Research Aims

1.2.1 Literature Review

A prerequisite to the main research aims, is a survey and critical analysis of related work, drawn from the literature of those disciplines deemed relevant to the research tasks.

1.2.2 Model

MS is a well recognised concept in the field of music psychology. Nevertheless it is an inherently ambiguous term and has been subject to varied definitions and classifications according to the context of the research (Ollen, 2006). The standard approach to assessment is the administration of an interview or questionnaire, sometimes accompanied by a perceptual test. One aim of this research is to develop a methodology for modelling the outward behavioural expression of internal characteristics, particularly MS as it pertains to music consumption patterns.

1.2.3 Methology

The primary aim of the work is to design a methodology for the design and construction of a system that is capable of evaluating an interaction-based model such as that described above, given a log of software interactions. The resultant methodology should allow the specification of the model by a designer with high levels of domain expertise but that lacks statistical or data analysis expertise.

1.2.4 Evaluation

The final goal of the work is to evaluate the model, and implementation, over an idealised population. This evaluation should utilise a real world music application to generate the activity logs which serve as input to the system, and the degree to which the system can appropriately identify a set of exemplar users should be assessed.

1.2.5 Dissertation structure

The work is organised as follows:

Chapter 2 - Background & Related work

Chapter 2 consists of a multidisciplinary survey of relevant literature which provides a theoretical grounding for much of the work that follows. MS as a formalised concept is discussed in the context of music psychology. A review of defeasible reasoning, a form of argumentation theory grounded in law and psychology, provides an insight into techniques that enable reasoning in the presence of uncertain or incomplete information. An examination of fuzzy logic then provides instruction as to the methods by which a natural language knowledge base can be translated into numerical form for manipulation within an argumentation framework. Finally, the proposals of this research are contextualised within the domain of RS technologies. A survey of the field touches on statistical techniques related to data mining and data analysis, with an emphasis on both how they inform the current work, and how the design driven approach of this work contrasts and compares to those strategies that rely on purely statistical techniques.

Chapter 3 - Methodology

Chapter 3 outlines the development of a model of MS as it pertains to user behaviour within an MSA. The design and construction of a system that emits the required metrics, aggregates them according to the specification of the model, and ultimately outputs a single numerical MS index is described.

Chapter 4 - Evaluation

Chapter 4 describes the execution of an evaluation strategy scoped in accordance with the resources available. A set of exemplar users are described. The activity of these exemplars is then simulated and the system executed over the resultant logs. The ability of the system to accurately identify users given such an idealised population is assessed. An alternative evaluation strategy that would provide more conclusive validation of the proposals of this work is discussed and presented as an opportunity for further research.

Chapter 5 - Conclusion

Chapter 5 includes a summary of the main findings of the work and a discussion of the potential for future research surrounding the topic.

Chapter 2

Background & Related Work

This chapter contains a survey of related work in the fields of music psychology, argumentation theory, and RS technology. The chapter begins with an examination of the concept of MS in section 2.1, with the aim of establishing and adopting a conception of the term appropriate to the work that follows. Section 2.2 contains a survey of the literature of defeasible reasoning and fuzzy logic, intended to establish a firm theoretical foundation from which a computational system capable of reasoning about natural language propositions can be constructed. Finally, in section 2.3, the field of RS technology is reviewed with the aim of appropriately contextualising the proposals of the current work.

2.1 Musical Sophistication

Musical Sophistication (MS), also termed *musical ability*, *musicality* or *musical intelligence*, is a prevalent concept in the field of music psychology. In a survey of the literature undertaken by Ollen (2006) it was identified as a covariate in 57% of the reviewed studies. While some disagreement exists as to the precise meaning of the term, it does have a considerable research history and definitions tend towards a combination of aural skill and performing ability.

Hallam (2010) assessed perceptions of MS across a range of musical and non-musical amateurs and professionals. It was found to be associated most strongly with a sense of rhythm, but also

> the ability to understand and interpret music, express thoughts and feelings through sound, being able to communicate through sound, motivation to engage with music, personal commitment to music, and being able to successfully engage musically with others.

Hallam and Prince (2003) note that development of means to capture this ability began as early as 1883 when Carl Stumpf developed a set of tests to preselect students for musical training. A later attempt to develop a standardised assessment, primarily through appraisal

of compositional comprehension, was undertaken by Révész (1953, pg.132), who adopted the term *musicality*, and defined it as

> the need and the capacity to understand and to experience the autonomous effects of music and to appraise musical utterances on the score of their objective quality.

Seashore (1915) believed the concept to encompass a number of loosely related characteristics which should comprise a profile of musicality rather than a single measurement. These were pitch, rhythm, timbre, loudness and tonal memory, as well as emotional reaction to music and vocal performance.

More recently, Ollen (2006) developed a technique for the assessment of MS through a ten question survey, intended to aid in the preliminary assessment of research participants in music psychology studies. Her work defines the term along three categories of interaction:

1. *Aural skills:* The ability to differentiate subtleties in tone, timbre and pitch, and to recognise musical structure.
2. *Receptive responses:* The ability to listen to, understand, appreciate and evaluate music.
3. *Generative skills:* The ability to play, sing, read, compose and improvise music.

Müllensiefen et al. (2014) developed a standardised assessment termed the Gold-MSI that was administered through the BBC to over 190,000 voluntary participants. The assessment utilises a mix of self-assessment and perceptual tests. They broadly concur with the conceptualisation of the term put forward by Hallam and Prince (2003) and Ollen (2006), however, place a stronger emphasis on musical skills not related to performance ability. Similar to Ollen, they define the term along three categories of behaviour:

1. Higher frequencies of exerting musical skills or behaviours.
2. Greater ease, accuracy or effect of musical behaviours when executed.
3. Greater and more varied repertoire of musical behaviour patterns.

While it is agreed that compositional and performance ability are an integral part of what constitutes MS, these categories of skill are not deemed directly relevant to the work at hand. Furthermore, they are not expressed through interaction with music listening software, which makes measurement an impossibility. Accordingly, the conception of the term adopted by this work draws from those aspects of the literature that refer to aural skills, compositional comprehension, varied repertoire and frequency of engagement. Müllensiefen et al. (2014) and Ollen (2006) are judged to most accurately describe the term as it pertains to the work at hand, and a selective synthesis of those works is judged to best capture those facets of behaviour of most relevance. The interpretation of the term ultimately adopted for this work is constrained to the following aspects, which are described above in more detail:

1. aural skills,
2. receptive responses,
3. frequency of engagement,
4. and variation of repertoire.

2.2 Formal Argumentation

2.2.1 Defeasible Reasoning

The methodology developed over the course of this research exploits defeasible reasoning techniques to aggregate attributes until they ultimately converge on a single numerical estimation of MS. This section includes an introduction to the concept of defeasible reasoning and the basic structure of an argumentation framework through which it can be implemented. Nute (1988) offers an informal description of this class of reasoning:

> It is the kind of "other things being equal" reasoning that proceeds from the assumption that we are dealing with the usual or normal case. Conclusions based on this kind of reasoning may be defeated if we find that the situation is not usual or normal.

When a rule supporting a conclusion may be defeated by new information, it is said that such reasoning is defeasible. Defeasible reasoning is a form of non-demonstrative reasoning. This is a type of reasoning that produces a contingent statement or claim, which represents a form of best guess given the available knowledge. This knowledge may be incomplete or contradictory, and the conclusion may be weakened or retracted as more evidence becomes known. It is a form of non-monotonic reasoning, which means that a claim may be reversed or altered upon discovery of new pieces of knowledge. Non-monotonic reasoning has been likened to a form of common sense reasoning which often occurs when presented with incomplete or partially consistent information. It is also a form of default reasoning.

Argumentation theory provides a framework within which arguments can be represented, supported or discarded in a defeasible reasoning process (Toni, 2010). It implements non-monotonic reasoning, and allows for modular, intuitive construction of an argument, and the incorporation of new evidence as it is received.

An argumentation framework consists of a set of arguments, characterised by Fox et al. (1993) as “tentative proofs for propositions”, and binary relations between them. The relations represent conflict between arguments. They are binary and are classified as either attack or defeat relations. Arguments, and the relations between them, are usually constructed by a designer. They are particular to the specific domain, the goals of the system, and a weighting, or *preferentiality*, assigned to the individual components by the designer. This research adopts a simplified argumentation framework, based upon that described by Longo (2012), within which reasoning occurs. An argument is defined as a set of premises $(P_1, ..., P_n)$, a claim C, and a tentative inference function $\rightarrow$, which links the premises to the claim.

2.2.2 Fuzzy Logic

One goal of this research is that the resultant system be accessible to a designer who is a domain expert but not necessarily technically skilled. It is envisaged that the designer specify a knowledge base in the form of a set of natural language propositions from which the system implementer can construct a technical implementation with relative ease. To this end the techniques of fuzzy logic are employed to translate the natural language specifications provided by the designer into a set of formal defeasible arguments that can be evaluated computationally.

Longo (2012) proposes the use of fuzzy set theory and degrees of truth, as described by Zadeh (1965) and Zadeh et al. (1996), to translate a natural language knowledge base into a defeasible argumentation structure. The knowledge base is initially expressed as a set of natural language *arguments*. For both the premises and claims, a process known as *fuzzification* is used to transform numerical values into grades of membership for linguistic terms. A function, known as a *membership function*, is described for each premise and each claim. This function, which accepts a measurement as input, describes the degree of confidence in that measurement as a numerical expression of the linguistic term the function is associated with. The output of a membership function is termed *degree of truth*, or *confidence*, and is represented as a numerical index between 0.0 and 1.0. The technique is best illustrated with an example and a complete transformation from argument to a single numerical indicator of confidence in the argument's claim follows.

Example 2.2.1
We are given a knowledge base consisting of the single natural language proposition "A user that listens to full albums frequently is exhibiting focus, but not if the average volume over that period is low". Two numerical measurements represent whole album frequency and average volume respectively. The task is to computationally model the logic of the knowledge base over the provided measurements.

The proposition, or argument, is first broken into its constituent parts and an inference function specified which relates the premises to the claim:

- P_1*: The user listens to full albums frequently.*
- P_2*: The average volume over the period is low.*
- C*: The user is exhibiting focus.*
- $\rightarrow$*:* $Confidence_{P1}(1.0 - Confidence_{P2})$ *(See below for an explanation of confidence)*

Two membership functions are defined which calculate the degree of truth, or confidence, for each of the premises. These functions accept a measurement as input and return a confidence value between 0.0 and 1.0. A third membership function describes how the aggregation of the premises impinges on confidence in the claim of the argument itself. See figure 2.1 for an illustration of the three functions with values that demonstrate the reasoning process.

In figure 2.1 the whole album measurement is 0.6 and a linear relationship exists between that frequency and confidence in the statement "The user listens to full albums frequently".

The membership function for the second premise is slightly more complex however. In the presence of a measurement of 15 or lower the designer has expressed certainty in the statement "The average volume over the period is low". Given an output of 50 or above however, the opposite is true and the statement has been judged unequivocally false. Between those two thresholds is a linear curve representing the fuzzy boundaries inherent to natural language. In this particular example, the measurement is 40, which translates to a confidence level of 0.25.

The third membership function describes how confidence in P_1 and P_2, when aggregated using the inference function, in turn affects confidence in the statement "the user is exhibiting focus". In this particular case a simple linear curve is judged to capture the relationship.

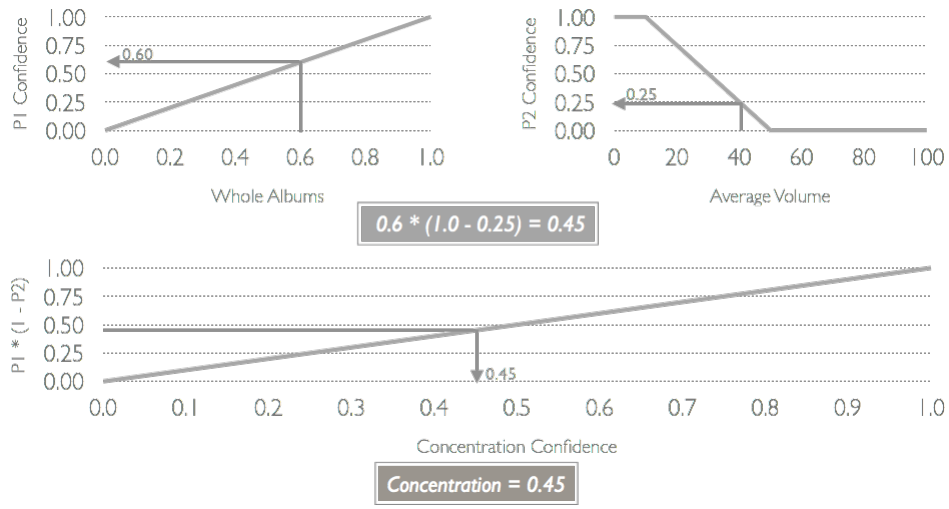

Fig. 2.1 The fuzzification process: a membership function calculates confidence in each of the premises from the provided measurements, and another, confidence in the claim given the output of the inference function.

2.3 Recommender Systems (RS)

The goal of a recommender system (RS) is to provide easily-accessible, high quality, and timely decision making support to users. Their use has become ubiquitous across many domains and they are utilised widely in shopping, news and multimedia applications. While the research described in this work does not directly describe an RS, nor does it draw significantly on the literature of the field, a primary motivation for the work is the assumption that the framework developed would have a place in such a system. Consequently an overview of RS technology and the space that such a hypothetical system would occupy within it follows.

2.3.1 Recommender System Taxonomy

Jannach et al. (2011) describe four widely accepted categories of approach to the generation of recommendations:

Collaborative filtering (CF)

Collaborative filtering (CF) is a widely implemented approach and is based upon the principle that users that exhibit commonalities in past consumption patterns can often serve as predictors for each other. A pure CF approach relies solely on relationships between users and items, and does not exploit knowledge about the items themselves. An advantage of this strategy is that the maintenance of a catalogue of metadata is not required. Pure CF systems are also unobtrusive and can improve over time through automated analysis of relationships and access patterns. This technique does suffer from what is termed the "cold start" problem, as described by Schein et al. (2002), whereby new users and items suffer from a paucity of relationship data and consequent obscurity. Fleder and Hosanagar (2009) note that systems that utilise this approach can also be susceptible to a "rich get richer" effect, essentially a feedback loop caused when the activity of the RS itself increases the visibility of candidate suggestions within the network. CF is the category within which a system such as that proposed by this research would exist, and an examination of some specific techniques follows in section 2.3.1.

Content-based filtering (CB)

Content-based approaches (CB) exploit metadata, or automated content analysis, in order to predict future preferences given a user's history. In the context of a music RS, CB could utilise a pre-existing taxonomy based on genre, or analysis of audio content for qualities such as tempo or timbre. This strategy is particularly suited to environments with low user density where CF would fail to produce high accuracy recommendations due to a dearth of relationship data in the system as a whole. A further advantage over content-agnostic approaches is that new items are viable and equal candidates immediately upon entry to the system. In some cases appropriate metadata may be available in a structured format such as an online music catalogue, however in others the necessitation of manual input, and consequent expense, can make this a less suitable approach.

Knowledge-based recommendations

A prerequisite for CF or CB is the existence of a relevant user history. In certain decision making situations, such as prior to an isolated major purchase, the user is unlikely to possess an exploitable history. In these circumstances a third type of approach is required, termed a knowledge-based approach (KB), in which means-end knowledge is utilised to determine optimum recommendations. In KB systems, the user is often directly queried as to their requirements or preferences. As an example, a system designed to aid in the purchase of a car may query the user as to their relative preference for fuel efficiency over acceleration speed. This approach requires significant context-specific knowledge and can impose unacceptable cognitive load on the user at a critical moment in a retail transaction. When executed well, however, KB can yield significant value in the form of both user satisfaction and increased

revenue (Burke, 2002).

Hybrid systems

In certain situations a hybrid of two or more of the above approaches may be appropriate. Preliminary CB may be used to narrow the scope of the network within which a CF system operates in order to increase accuracy or improve efficiency. In some cases CB is used to solve the cold start problem, and the system may transition to CF once sufficient relational data is available. An example of a hybrid approach is the Netflix system[1], in which users are initially queried as to their preferred genres of movie (KB), but as they use the system on an ongoing basis a CF system exploits their access histories and ratings to improve accuracy.

2.3.2 Collaborative filtering

Collaborative filtering is the RS category under which a system such as that proposed by the current research would fall. Consequently, an examination of some of the more widely utilised techniques follows. This includes a sampling of traditional data mining techniques as they apply to RS technologies, and highlights some purely statistical alternatives to the design driven methodology adopted for this research.

CF techniques are often further categorised as either memory-based or model-based (Jannach et al., 2011). The distinction relates essentially to the use of preprocessing. A memory-based approach maintains the access history or ratings database in memory and performs all relational computations at run time. A model-based approach, such as that proposed by this work, performs offline preprocessing of data to build a model, which is subsequently exploited at runtime to make predictions.

There follows a representative sampling of the wide range of CF techniques to be found within the literature of the domain, and in widespread commercial use across industry.

Matrix factorisation

Matrix factorisation techniques comprise a class of model-based approach that can be used to extract latent factors from the historical data within a system. In a music context, such factors may correspond to obvious groupings such as genre, or geographical origin, but they can also be difficult, or impossible to interpret.

One of the earliest examples of the application of this range of techniques, Singular value decomposition (SVD), was conceived by Deerwester et al. (1990) as a method of information retrieval using semantic analysis of documents. The technique, as originally proposed, involves the reduction of both documents and queries to a representative vector of terms, which are then analysed for co-occurrence. The key technique in this approach that distinguishes it as model-based rather than memory-based, is the preprocessing stage in which the matrix of document vectors is collapsed according to the semantic meaning of the constituent terms.

[1] `http://www.netflix.com`

Sarwar et al. (2000b) expanded on the dimensionality reduction techniques of SVD and applied them successfully to the task of product recommendations. This approach captured the latent relationships between customers and products, rather than queries and documents. Canny (2002) applied a similar approach to movie recommendations, with a particular emphasis on the additional privacy that factor analysis can afford over memory-based CF approaches.

Matrix factorisation RS techniques generally involve the decomposition of a single user-item matrix of ratings or access histories into two separate matrices that represent the most significant entries of the original. These individual matrices, representing users and items respectively, are then projected in two-dimensional space and analysed for clustering and spatial relationships.

Nearest neighbour

One of the earliest RS strategies is termed *nearest neighbour recommendation* and is still in widespread use. It is a memory-based CF approach that utilises matrix methods to generate recommendations based on user or item similarity.

A user-item matrix is constructed and maintained. This matrix contains user ratings, either explicitly supplied, or implicitly harvested by the system from the access history and behaviour of users. The similarity between users, or between items, is calculated using one of many possible similarity measures. Those with the highest similarity score to a given user, or to items rated highly by the user, serve as predictors.

A number of methods have been proposed to evaluate similarity in a nearest neighbour algorithm. Herlocker et al. (2004) performed an evaluation of the most widely used and found the *Pearson Correlation Coefficient* to be the most effective measure with respect to *user* similarity, while *cosine similarity* was preferable in techniques that compare *items*.

The Pearson Coefficient is a measure of linear correlation between two variables and gives a value between +1.0 and –1.0, where a negative value expresses negative correlation. It is particularly useful as a measure of user similarity because it accounts for differences in the way users interpret the rating scale and so measures correlation in trends rather than absolute terms. Formula 2.3.1 defines the Pearson Coefficient between two users a and b.

$$pearson(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \overline{r_a})(r_{b,p} - \overline{r_b})}{\sqrt{\sum_{p \in P}(r_{a,p} - \overline{r_a})^2 \sqrt{\sum_{p \in P}(r_{b,p} - \overline{r_b})}}}$$

where $\overline{r_a}$ = the average rating of user a

Formula 2.3.1 Pearson's Coefficient based similarity measurement between users a and b.

Cosine similarity, presented in formula 2.3.2, measures the similarity between two n-dimensional vectors based on the angle between them and gives a value between 0.0 and 1.0.

When the vectors are populated with ratings assigned to the two items by users, this provides a measure of similarity between the items.

$$cosine(a,b) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}| * |\vec{b}|}$$

Formula 2.3.2 Cosine similarity measure between two items a and b.

Association rule mining

Association rule mining is another model-based CF technique that is typically used to detect patterns in sales transactions but is also applied to content recommendations (Jannach et al., 2011). Sarwar et al. (2000a) describe the technique:

> Let us denote a collection of m products $\{P_1, ..., P_2, P_m\}$ by P. A transaction $T \subseteq P$ is defined to be a set of products that are purchased together. An association rule between two sets of products X and Y, such that $X, Y \subseteq P$ and $X \cap Y = 0$, states that the presence of products in the set X in the transaction T indicates a strong likelihood that products from the set Y are also present in T.

Two measures of quality are generally calculated with respect to such rules, and context-dependent minimum thresholds are applied:

$$support = \frac{\text{number of transactions containing } X \cup Y}{\text{number of transactions}}$$

$$confidence = \frac{\text{number of transactions containing } X \cup Y}{\text{number of transactions containing } X}$$

Logistic regression

Others have suggested alternative approaches to CF grounded in techniques drawn from statistics and probability. Logistic regression (Hastie et al., 2009) is a general linear model used to predict probabilities. Vucetic and Obradovic (2004) propose a system that is trained by experts who describe the relationships between items. A logistic regression is then applied in order to predict the preferences of users who have rated some but not all of the training items. Another example of an approach based on logistic regression is that proposed by Zheng et al. (2011), which considers CF as an inference problem on a bipartite graph where the links represent ratings from users to items. In contrast to that proposed by Vucetic

and Obradovic (2004), this is a non-supervised learning approach which applies an ordered logistic regression to estimate similarities among both users and items.

2.4 Conclusion

This chapter established a theoretical foundation for the design and implementation outlined in Chapter 3. The field of music psychology was surveyed in order to establish a firm basis from which to approach the concept of musical sophistication. Strategies for defeasible reasoning were examined with a view to the establishment of a formal system of argumentation upon which an estimation of musical sophistication can be constructed. A brief sampling of the discipline of fuzzy logic provided instruction as to how the ambiguities of a natural language knowledge base can be transformed into a numerical representation for manipulation within that argumentation system. Finally, the field of recommender systems was surveyed in order to establish a context for the motivation underlying this work, and also to highlight some alternative approaches that rely on data mining and statistical techniques.

Chapter 3

Methodology

This chapter describes the development of both a model of the behaviour of a musically sophisticated user, and a system that populates and evaluates that model. The nature of the iterative design-driven process which was undertaken is that design and implementation are closely intertwined. Consequently the two are described together as intrinsic parts of the same process.

The chapter begins with an overview of the technical approach in section 3.1. Section 3.2 follows, with a discussion of the range of relevant user interactions and corresponding metrics. Section 3.3 contains an analysis of 4 characteristics identified as key components of MS, and how they relate to the natural language knowledge base from which the model was derived. Section 3.4 lists each of a set of independent processing components designed to extract single units of information from the activity log. The chapter closes with an outline of the reasoning process in section 3.5.

3.1 Overview

An initial model of the behaviour of a musically sophisticated user was developed through analysis of music software and services in popular use, and of the various facets of MS identified in section 2.1. An integral part of the development process, however, was a feedback loop from the evaluation stage back to the design stage which enabled the verification of assumptions, identification of omissions, and an iterative optimisation of the model. The particular model developed over the course of this research is presented not as a definitive model of musically sophisticated behaviour, but as an archetype for the kind of interaction-based model the class of system described is intended to support. One rationale for the design driven approach outlined in this chapter was the research aim that the system and methodology be accessible to a model designer who possesses domain expertise but lacks advanced statistical or data analysis skills.

A hierarchical model was constructed, comprised of layers of discrete units of information, inferences, and tentative conclusions which are drawn from the relationships between them. A set of reasoning rules was developed, which specify the procedure by which be-

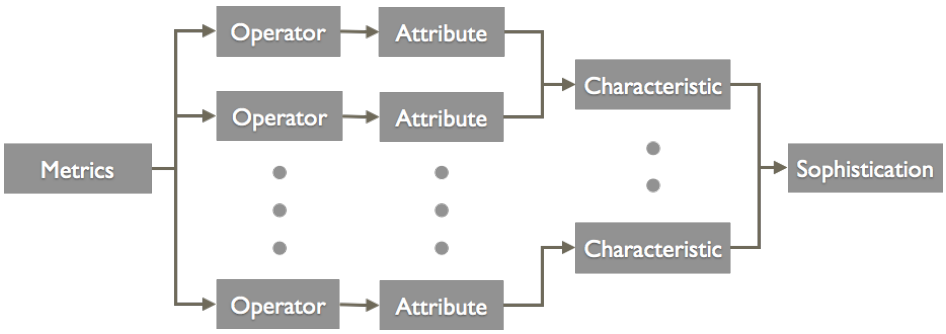

Fig. 3.1 An overview of the technical approach.

haviour traits are aggregated and interpreted to ultimately produce a single numerical index. This index represents a tentative estimation of MS given the data available to the system at runtime. This index is liable to change as new information becomes available, and could be extended to represent the MS of a population as a whole, or of a particular target demographic.

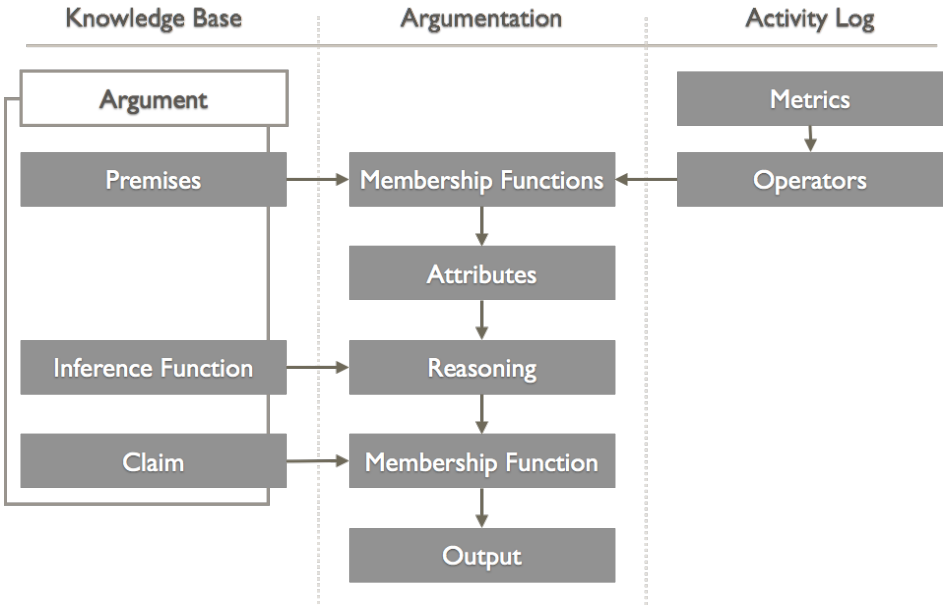

Fig. 3.2 The first stage of the argumentation process.

The lowest layer of the model is comprised of interactions. An open source music software application (MSA) was instrumented so as to emit a detailed log of user interaction. This is termed the *activity log*, and its individual entries are termed *metrics*. A set of independent processing units, termed *operators*, was developed. Each operator maintains state as it is fed metrics and can be queried to provide an isolated piece of information about the user's behavioural tendencies. These isolated, context-free, indicators are termed *attributes*.

Defeasible reasoning and fuzzy logic techniques are exploited to aggregate a hierarchy of these indicators computationally until they converge into the four *characteristic* indicators that represent the top layer of the model. See figures 3.2 and 3.3 for a graphical representation of this process. These indicators then serve as inputs to the final stage of reasoning, the output of which is a bounded numerical index between 0.0 and 1.0 that represents an estimation of

the user's MS. Four characteristics were identified, each of which captures a specific facet of musically sophisticated activity. These are *focus*, *curiosity*, *specificity* and *organisation*, and are described in detail in section 3.3.

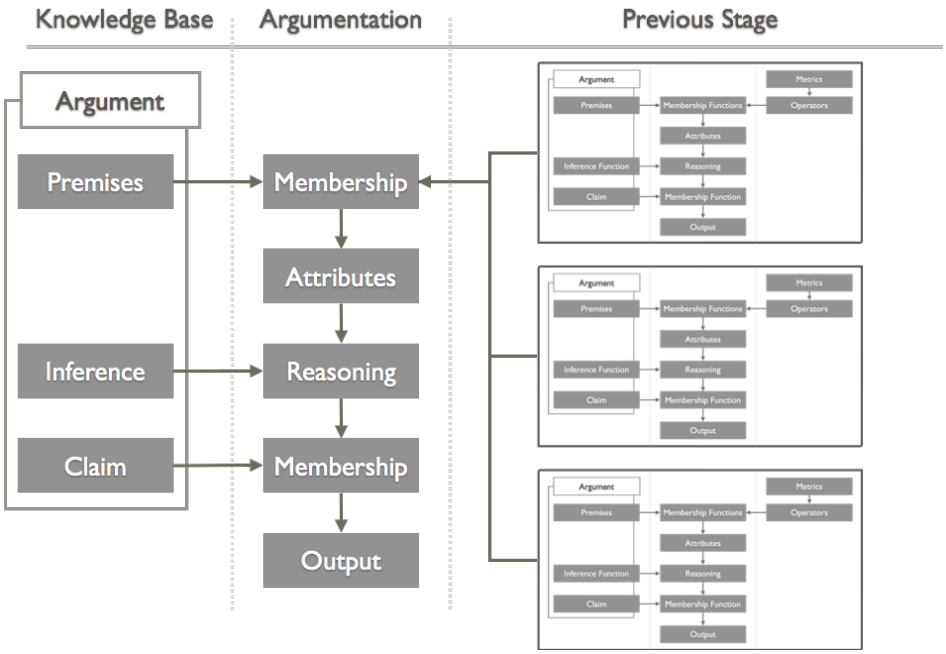

Fig. 3.3 Subsequent stages of the argumentation process.

3.2 Metrics

An analysis of user behaviour and the functionality provided by widely used music services and software revealed three distinct categories of intent that underly most interactions with an MSA. At any given time a user may be engaged in fulfilling one or more of these in parallel. This multitasking behaviour results in a serially interleaved, chronological log of metrics which must subsequently be deinterleaved, or statistically analysed, in order to accurately interpret intent. A description of these 3 categories follows, in sections 3.2.1 (listen), 3.2.2 (discover), and 3.2.3 (manage).

3.2.1 Listen

This category of interaction can serve as an indicator of the user's direct engagement with the music being output by the software at any given time. A user's behaviour as they listen to music can indicate direct engagement with the material, passivity, disinterest or possibly even the physical absence of the user. Table 3.1 lists the metrics recorded as the user listens to music.

Metric	Action	Additional Data
Load	Song loaded for playback	Metadata: *Title* *Artist* *Album* *Track* *Genre* *Year* *Popularity*
		Source of song: *Spotify* *File* *Radio*
		Bitrate at which song is encoded.
		Flag indicating whether song belongs to same album as last song loaded (*same_album*).
		Flag indicating causal event (*instigator*): *First*: Song is first in an album or playlist. *Manual*: User skipped manually to song. *Auto*: Song next in album or playlist.
Play	Play button activated	
Pause	Pause button activated	
Stop	Stop button activated	
Seek	Seek function activated	New song position
Skip	Skip song function activated	Previous position
		Current position
Volume	Volume altered or song loaded	Previous volume
		Current volume

Table 3.1 Metrics recorded in the *listen* category.

3.2.2 Discover

Discovery is a key function of many modern music services and encompasses features such as online catalog search, personalised radio, recommender services, discovery applications, and subscriptions to the activity feeds of other users. The manner in which a user acquires new music and expands their listening history can serve as an indicator of their existing musical knowledge, their openness to new music and the degree to which they drive their

own education or rely on others to curate their discovery experience. Table 3.2 lists the metrics recorded as the user performs activities related to music discovery.

Metric	Action	Additional Data
Search	Online search executed	Search term
LoadPlaylist	Playlist loaded for playback	Comma separated list of unique artists
		Comma separated list of unique albums
		Comma separated list of unique songs
RadioPlay	Song loaded from online radio playlist	*[See Load]*

Table 3.2 Metrics recorded in the *discover* category.

3.2.3 Manage

Management of a personalised collection of music is a category of interaction that offers little immediate stimulatory reward. The work outlay required to tag songs, edit metadata, construct and maintain playlists, and acquire artist, label and period collections, points to a user who is consciously invested in the domain. Table 3.3 lists metrics related to library management.

Metric	Action	Additional Data
EditMetadata	Song metadata edited	Title Update
		Artist Update
		Album Artist Update
		Genre Update
		Art Update
		Year Update
SavePlaylist	Playlist saved to disk	Comma separated list of unique artists
		Comma separated list of unique albums
		Comma separated list of unique song years

Table 3.3 Metrics recorded in the *manage* category.

3.3 Characteristics

The MS model is built around four broad characteristics which can be inferred from user behaviour. These are focus, curiosity, organisation, and specificity.

The user causes the generation of metrics while servicing intent. It is the patterns of intent that ultimately indicate the presence or absence of these higher level indicators. The final MS index is computed as a weighted arithmetic mean of characteristic indicators.

The development of these characteristics was driven by the range of latent information available in the activity log. The initial rules and metrics of which they are comprised were formulated through intuitive reasoning and analysis of the concept of MS adopted by this research. A feedback loop from the evaluation stage enabled the refinement of the model empirically over time.

Future development could conceivably lead to significant refinement and revision of the model, for example an empirical study whereby brain activity and other physical indicators of focus or curiosity are monitored and correlated with user behaviour. The nature of the implementation allows for such updated assumptions to be reflected through adjustment to the operators and aggregation functions, essentially adjustments to the parameterisation of the model. In this respect, what is described is a framework to which any interaction-based model can be applied, the specifics of which may vary from one context to another given the metrics available and the current state of knowledge with regard to user behaviour in the domain.

A description of each of the 4 characteristics follows.

3.3.1 Focus

A high focus score indicates a tendency towards immersion in the work of a particular artist, genre or time period. A user with low levels of focus tends to skip from one artist to another, play single tracks, and listen to radio or pre-existing playlists or streams. In contrast, a user with high focus shows signs of thoroughly absorbing an album, and listens to it repeatedly over a given time period. They exhibit this behaviour not just in relation to single units such as albums, but also to artists, genres and periods. Their listening patterns exhibit a conscious attempt to integrate an album or artist into their musical knowledge. A high focus score indicates purpose and intent behind choice of listening material. It indicates that music selection is not primarily reactive or in response to transitive moods and emotions, but is approached systematically as a form of project. Time periods and artists are methodically investigated and internalised. A user with a high score in this category may not necessarily have a broad range of knowledge, but in the areas in which they take interest they display a depth of knowledge and intimate familiarity with particular albums, genres, artists and time periods. Table 3.4 lists the natural language knowledge base that describes the behaviour of a focused user.

3.3.2 Curiosity

Whereas focus can be broadly described as a measure of depth, curiosity is a measure of breadth. A high score in this category indicates a willingness to sample new music, a broad

The user is exhibiting focus if they...	Premises
Tend to listen to whole albums, but not if they listen at a very low volume	Listens to whole albums
	Listens at very low volume
Maintain playlists, but only if those playlists exhibit clustering around particular artists, genres or time periods (in descending order of significance).	User maintains playlists
	Playlists exhibit artist focus
	Playlists exhibit genre focus
	Playlists exhibit period focus
Show continuity with respect to song, album or artist on session resume (in descending order of significance).	Sessions exhibit song continuity
	Sessions exhibit album continuity
	Sessions exhibit artist continuity
Exhibit variance with respect to albums in long term listening patterns, but repetition in short term listening patterns.	Exhibits long term album variance
	Exhibits short term album variance
Tend to listen to multiple unique albums from single artists (completism).	Ratio of albums to artists is high
Tend not to utilise radio	Listens to radio

Table 3.4 The natural language knowledge base that describes $focus$.

range of tastes, frequent exploratory searches, and experimentation. A user with a high curiosity score expends significant effort in finding new music. They have a broad range of knowledge, that encompasses many musical styles, time periods and genres. In contrast to concentration, this indicator represents a willingness to experiment with multiple genres and periods rather than maintain a singular focus on existing domains of knowledge. Crucial to this category though is evidence of engagement and intent. For example, while use of radio may contribute to a curiosity score, the contribution is magnified significantly if evidence of engagement is present. In this example, engagement could be exhibited by a tendency to save tracks from the radio to a playlist, or subsequent search and investigation of artists discovered through the feature.

3.3.3 Organisation

Organisation captures activities related to the management and maintenance of a collection of music. A user with a high organisation score expends the time and effort required to maintain correct metadata about their library. Their collection shows little evidence of missing metadata or album covers. The genre tag is edited frequently as the user imposes a personalised taxonomy upon their collection. Spelling errors within the metadata are infrequent.

The user is exhibiting curiosity if they...	Premises
Do not consistently listen to high popularity material	Song choices exhibit high popularity
Utilise radio but only if they sometimes pursue discoveries afterwards	Utilises radio
	Exhibits discovery pursual
Exhibit variety in choice of artist, album and genre	Variety in artist choice
	Variety in album choice
	Variety in genre choice
Search frequently, but only if search terms are varied	Searches frequently
	Search terms exhibit variance

Table 3.5 The natural language knowledge base that describes *curiosity*.

Extensive playlists may be maintained. The album artist field[1] is utilised to correctly group compilation or collaborative albums. Prior to the digitisation of the domain, an exemplar of this class of user would have maintained well organised, compact disc or vinyl record collections in immaculate physical condition. Table 3.6 lists the natural language knowledge base that describes organisation.

The user is exhibiting organisation if they...	Premises
Edit critical metadata or collection exhibits complete metadata (*artist*, *album*, *title*).	Edits critical metadata
	Critical metadata usually present
Frequently create and save playlists.	Frequently creates playlists
	Frequently saves playlists
Frequently edit genre tag or existing genre classification is consistent.	Frequently edits genre
	Genre tags exhibit consistency

Table 3.6 The natural language knowledge base that describes *organisation*.

3.3.4 Specificity

Specificity captures specific intent on the part of the user. It is another indicator of engagement over passivity, and a high rating indicates the user has strong preferences, maintains an awareness of what they are listening to and is ready to intervene to ensure a satisfactory listening experience. A user with a high specificity score is likely to skip particular tracks or skip to particular sections of tracks. Their search patterns exhibit precision, and they are more likely to search for a particular album or song rather than an artist or genre. They may

[1]The *album artist* field is distinct from the *artist* field and allows an album comprised of songs by multiple artists to be grouped under a single artist name in a library.

search for a particular edition of an album, or version of a song. This class of user has well defined taste and is unwilling to tolerate music that falls outside of those bounds. Table 3.7 lists the natural language knowledge base that describes specificity.

The user is exhibiting specificity if they...	Premises
Use seek functionality	Uses seek functionality
Use skip functionality	Uses skip functionality
Do not consistently begin playback from first track	Consistently begins playback from first track
Favour high bitrate encodings	Favours high bitrate encodings
Search for song or albums more frequently than artists	Searches for songs frequently Searches for albums frequently Searches for artists frequently

Table 3.7 The natural language knowledge base that describes *specificity*.

3.4 Attributes & Operators

Each of the argument premises extracted from the knowledge base, as described in the previous section, describes a specific, isolated behavioural trait, the presence or absence of which is taken into consideration in the reasoning stage. A set of processing units, termed *operators*, was design and developed over the course of this work. Each produces a measurement related to one or more of premise from the information available in the activity log. These measurements are transformed into numerical indicators that represent the system's confidence that the premise is true or false. This is achieved through a set of membership functions which were developed as part of the model design process. The output of the membership functions are termed *attributes* and each is a numerical representation of an argument premise, which enables it to be computationally manipulated and reasoned about. See figure 3.4 for an illustration of the procedure by which premises are transformed into attributes.

There follows a listing of each of the operators.

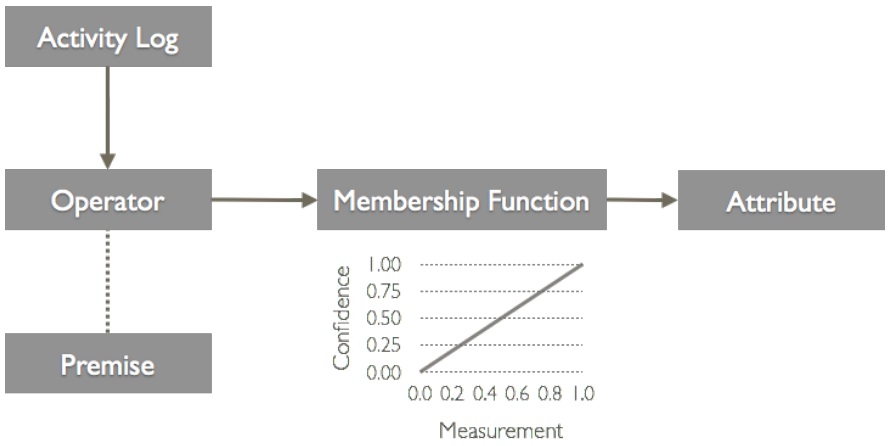

Fig. 3.4 The fuzzification process by which a premise is transformed to an attribute.

Notation

Symbols utilised throughout this section include:

L_{value}	An activity log entry of type *value*.
$\{value\}$	The set of all log entries of type *value*.
$\|S\|$	The cardinality of set S.
$map(f \to S)$	The set of results derived from the application of function f to each member of the set S.

3.4.1 Whole albums

Whole_Albums is an indicator of a preference for whole albums over individual songs. It is expressed as the proportion of occurrences of the *same_album* flag to total number of songs loaded.

$$whole\_albums = \frac{|\{same\_album\}|}{|\{song\_load\}|}$$

3.4.2 Density

The *Density* operator expresses the average number of unique values for a particular meta tag over a specified number of sequential song loads. The operator is parameterised with two values:

1. A numeric constant, a, which specifies the maximum size of a first in, last out set (A) of most recent *song_load* log entries.
2. A meta tag identifier, *tag_name*, which specifies the meta tag of interest (e.g. *album*, *artist*, *genre*).

The operator produces a measure of the average number of unique values of *field* for every a songs loaded.

$$density(tag\_name, a) = (\sum_{i=a}^{n} \frac{|unique(map(tag\_value : tag\_name \rightarrow A_i))|}{i}) * \frac{1}{n-a}$$
$$\text{where } n = |\{song\_load\}|$$
$$A_i = \text{The set of } a \text{ most recent } L_{song\_load}$$

3.4.3 Focus

The *Focus* operator captures a tendency to focus on an evolving subset of a larger collection of albums, artists or genres. A low *Density* expresses this tendency but it is vulnerable to false positives in the case of a user with a limited overall collection because their listening history will naturally exhibit low variety. In order to mitigate the impact of this vulnerability, the *Density* operator is executed twice: once with a lower value of a, and again with a significantly higher value. The *Focus* operator's output is the ratio of the result of the first execution to the result of the second. A high ratio indicates that smaller subsets of plays exhibit a low number of unique tag values in relation to number of plays, however over a longer time period wider variety is present. Essentially a clustering of values with respect to time is detected.

$$focus(tag, a, b) = \frac{density(tag, a)}{density(tag, b * a)}, b \in \mathbb{N}_{>1}$$

3.4.4 Average Volume

The *Average_Volume* operator calculates the average volume of audio output with a granularity of one second.

$$average\_volume = \frac{(\sum_{i=1}^{S} V_i)}{S} * \frac{1}{100}$$
$$\text{where } S = \text{the number of seconds of audio played}$$
$$V_i = \text{the volume at second } i$$

3.4.5 Session Continuity

The *Session_Continuity* operator calculates the probability that a user will return to the same artist, album or song after a session is interrupted. Software exits and resumes are analysed in order to determine the frequency with which this behaviour occurs:

$$session\_continuity(f) = \frac{R}{n-1}$$

$$\text{where } R = \sum_{i=2}^{n} \begin{cases} 1 & f(S_i) == f(E_{i-1}) \\ 0 & \text{otherwise} \end{cases}$$

$$S_i = \text{The first song loaded after } session\_start_i$$

$$E_i = \text{The last song loaded before } session\_end_{i-1}$$

$$n = |\{session\_start\}|$$

$$f = \text{a function that accepts an } L_{song\_load} \text{ and returns a unique identifier for one of album, genre, artist or song associated with the entry.}$$

3.4.6 Playlist Focus

The *Playlist_Focus* operator measures the proportion of artists, periods or genres to songs in a playlist. It is parameterised with a tag name (e.g. *artist*).

$$playlist\_focus(tag\_name) = (\sum_{i=1}^{n} \frac{|unique(map(f \rightarrow songs(playlist_i)))|}{|songs(playlist_i)|}) * \frac{1}{n}$$

$$\text{where } n = |\{playlist\_load\}|$$

$$f(song) = tag\_value : tag\_name(song)$$

3.4.7 Playlist Maintenance

The *Playlist_Maintenance* operator records frequency of $L_{playlist\_save}$ as a proportion of L_{song_load} frequency.

$$playlist\_maintenance = \frac{|\{playlist\_save\}|}{|\{song\_load\}|}$$

3.4.8 Metadata Maintenance

The *Metadata_Maintenance* operator produces a measure of the frequency with which the user edits standard meta tags. It counts updates to multiple tags applied in a single operation by the software as a single edit.

$$metadata\_maintenance = \frac{|\{meta\_update\}|}{|\{song\_load\}|}$$

3.4.9 Tag Maintenance

The *Tag_Maintenance* operator records updates to particular meta tags and is parameterised with a tag name.

$$tag\_maintenance(tag\_name) = \frac{|\{meta\_update : tag\_name\}|}{|\{song\_load\}|}$$

3.4.10 Missing Metadata

The *Missing_Metadata* operator records empty meta tag values or the presence of common placeholder text such as “untitled track”, “untitled song” or “unknown artist”.

$$missing\_metadata(tag\_name) = \frac{|\{missing\_data\}|}{|\{song\_load\}|}$$

3.4.11 Skip

The *Skip* operator records the ratio of songs skipped to songs loaded.

$$skip = \frac{|\{song\_skip\}|}{|\{song\_load\}|}$$

3.4.12 Seek

The *Seek* operator expresses the frequency with which the user manually repositions playback in a song as a ratio to the total number of songs loaded.

$$seek = \frac{|\{song\_seek\}|}{|\{song\_load\}|}$$

3.4.13 Bitrate

The *Bitrate* operator calculates the average bitrate at which songs files are encoded.

$$bitrate = (\sum_{i=1}^{n} bitrate(song_i)) * \frac{1}{n}$$
$$\text{where } n = |\{song\_load\}|$$

3.4.14 Completism

The *Completism* operator measures the number of unique albums relative to the number of unique artists.

$$completism = \frac{|unique(map(tag\_value : album \rightarrow songs))|}{|unique(map(tag\_value : artist \rightarrow songs))|}$$

3.4.15 Search Frequency

The *Search_Frequency* operator expresses total search executions as a ratio of total song loads.

$$search\_frequency = \frac{|\{search\}|}{|\{song\_load\}|}$$

3.4.16 Discovery

The *Discovery* operator measures the use of online discovery services such as recommendations, discovery applications and radio. The only metric available in this category is online radio play due to limitations of the software in use.

$$discovery = \frac{|\{radio\_load\}|}{|\{song\_load\}|}$$

3.4.17 Pursual

The *Pursual* operator calculates the probability that the user will search for an artist or album after they encounter it when using online radio. A first in, first out set of artists and albums played through the radio service is maintained and compared against subsequent search terms. The operator is parameterised with the size of the set and the meta tag name to be compared.

$$pursual(tag\_name, a) = \frac{M}{n}$$

$$\text{where } M = \sum_{i=1}^{n} \begin{cases} 1 & term(L_{search_i}) \in map(tag\_value : tag\_name \rightarrow A) \\ 0 & \text{otherwise} \end{cases}$$

$$A = \text{The set of } a \text{ most recent } L_{radio\_load}$$

$$n = |\{search\}|$$

3.4.18 Variety

Variety is a parameterised operator that accepts a tag name and returns the average number of unique corresponding tag values per song load.

$$variety(tag\_name) = \frac{|unique(map(tag\_value : tag\_name \rightarrow \{song\_load\}))|}{|\{song\_load\}|}$$

3.4.19 Popularity

Popularity information is available when a song is loaded from Spotify[2]. It is represented on an integer scale from 0 to 100. The *Popularity* operator calculates the average popularity rating for those songs for which the metric is available.

$$popularity = \frac{|map(popularity \rightarrow \{song\_load\})|}{S}$$

$$\text{where } S = \sum_{i=1}^{n} \begin{cases} 1 & source(L_{song\_load_i}) == \text{“}spotify\text{”} \\ 0 & \text{otherwise} \end{cases}$$

$$n = |\{song\_load\ \}|$$

3.5 Reasoning

Attributes, numerical representations of the premises of a particular argument, are computationally manipulated and aggregated in the reasoning stage. The output of each reasoning unit is input to a membership function which produces a single numerical indicator. This indicator represents confidence in the claim of the argument.

The argumentation framework resultant from the activities described in the previous sections mirrors the logic of the natural language knowledge base from which it is derived. As a result of the rich descriptive nature of the membership functions which produce the inputs to the reasoning stage, the reasoning process itself tends towards a set of simple mathematical operations. These consist primarily of:

- *multiplication*: $\boldsymbol{X * Y}$,
- *weighted mean*: $\overline{\boldsymbol{(X_{w1}, Y_{w2})}}$,
- and *attack (1.0 - X)*: $\boldsymbol{X!}$.

A set of inference functions was developed which accept either one or more attributes, or the output of the preceding stage of reasoning, as input. The initial formulation of these functions was established through intuitive reasoning about the domain, and about the relationships between the arguments that comprise the knowledge base. This is a feature of a design-driven methodology which allows a system to be developed incrementally through consultation with a domain expert and negates the requirement for large datasets or advanced statistical techniques, particularly in the early stages of development. As the system was evaluated, the inference functions were modified through a refinement mechanism intended

[2]Spotify is a widely used online music subscription service (`http://www.spotify.com`).

to improve accuracy, identify omissions, and shape the model towards a closer representation of the characteristic being assessed. This stands in contrast to many traditional data mining techniques, some of which were described in section 2.3.1, that are often designed to capture correlations and patterns from an existing dataset.

See figures 3.5, 3.6, 3.7, 3.8 and 3.9 for illustrations of the reasoning logic that leads to focus, curiosity, specificity, organisation and sophistication, respectively. The output of each of these reasoning units is subsequently input to a membership function which spreads or constrains it over a bounded confidence range (0.0 to 1.0).

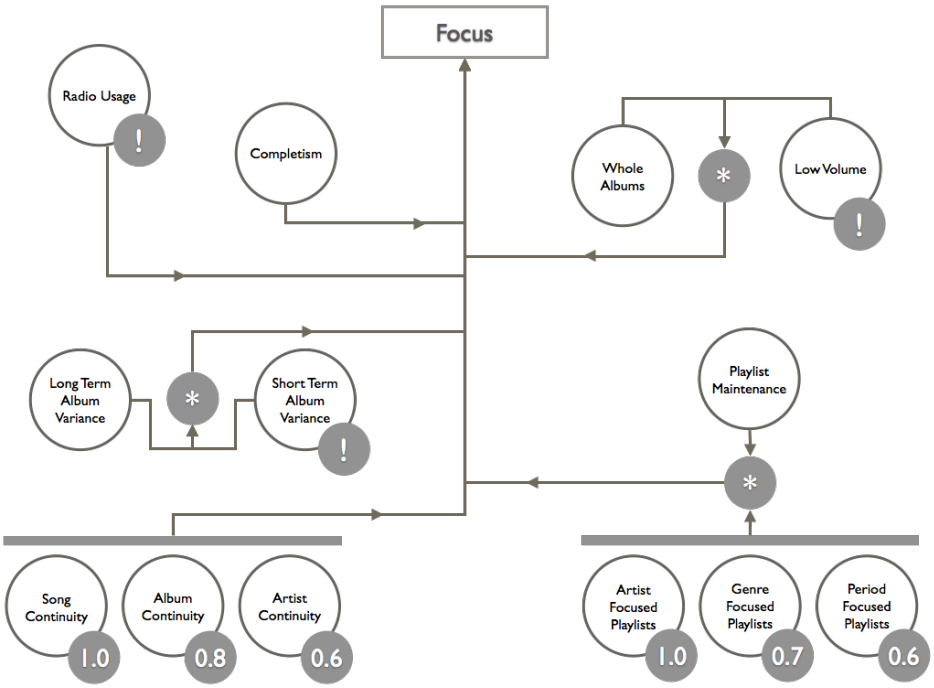

Fig. 3.5 The reasoning hierarchy by which focus is calculated.

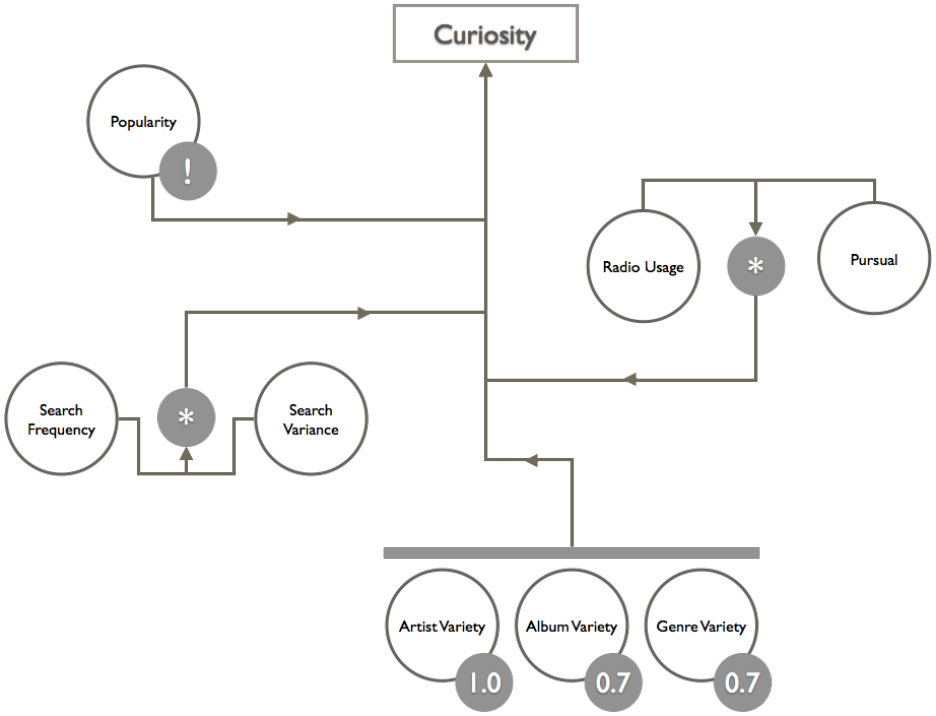

Fig. 3.6 The reasoning hierarchy by which curiosity is calculated.

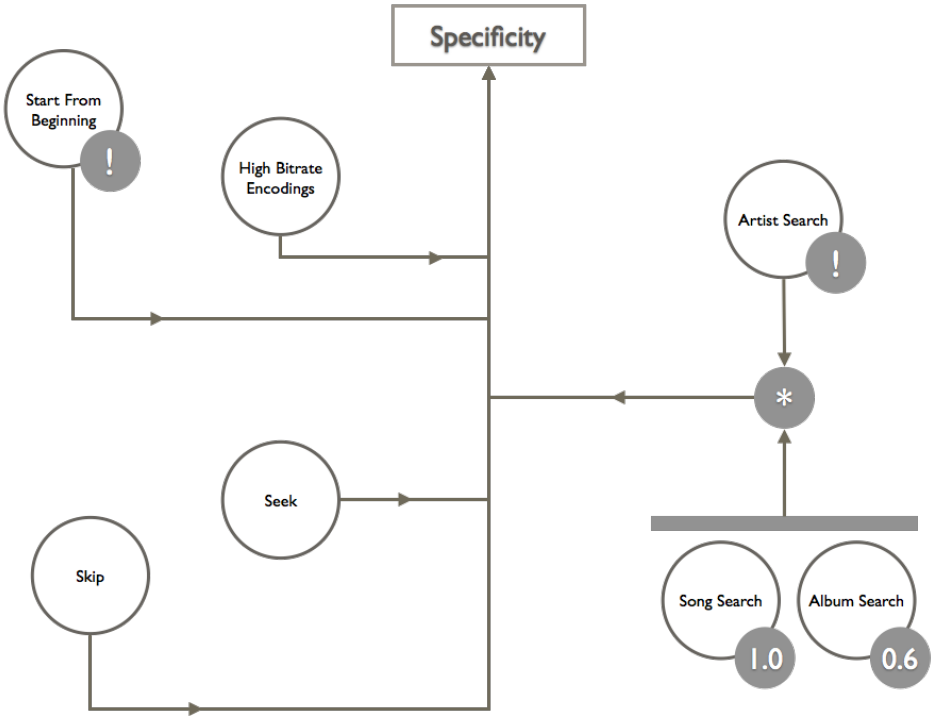

Fig. 3.7 The reasoning hierarchy by which specificity is calculated.

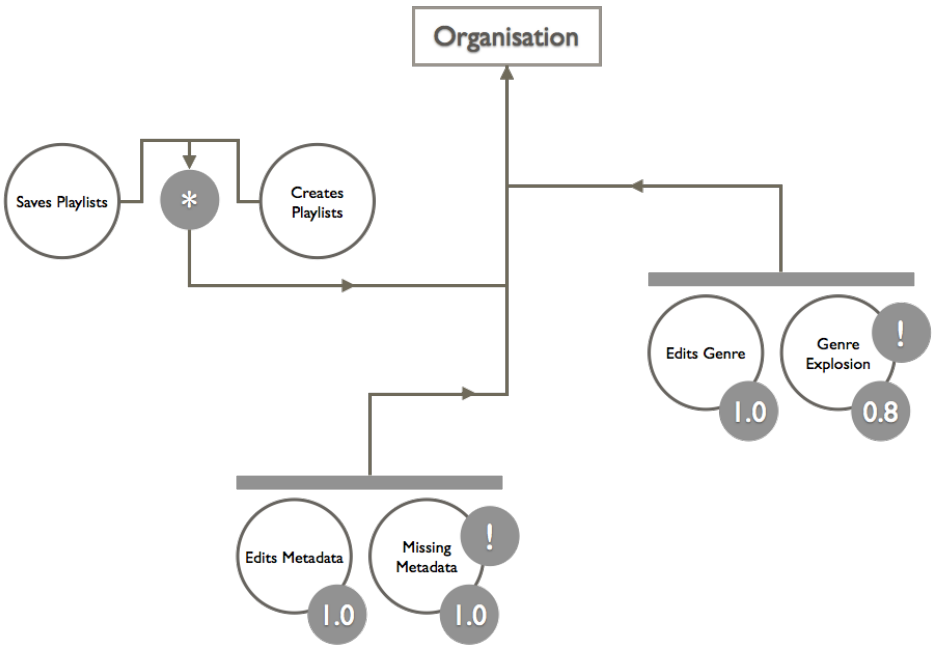

Fig. 3.8 The reasoning hierarchy by which organisation is calculated

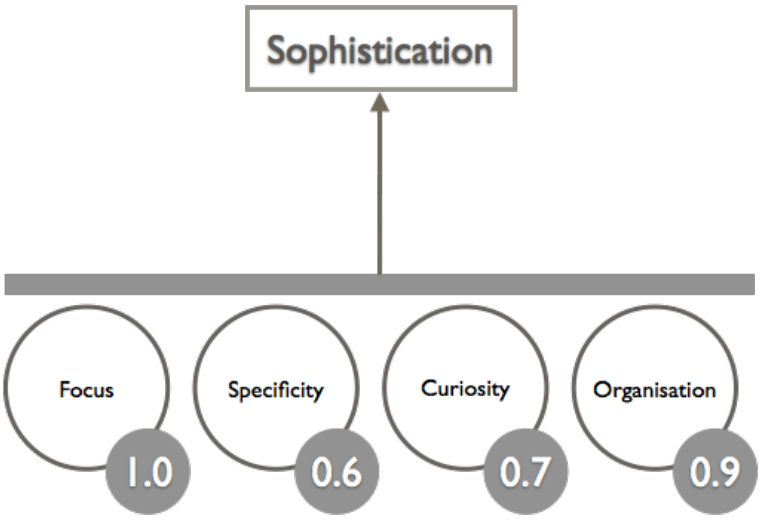

Fig. 3.9 The reasoning hierarchy by which sophistication is calculated.

3.6 Conclusion

The preceding sections, wherein the methodology developed over the course of this research was outlined in detail, comprise an account of the core activities upon which much of the contribution of the current work rests. The methodology described enables the transformation of a knowledge base, provided by a domain expert in the form of a set of natural language propositions, into a computational argumentation framework populated by a log of software interactions. The initial derivation of the model from the knowledge base, and the design-driven approach by which it was populated, evaluated and iteratively refined, is a process independent of the concept of MS, however. The resultant methodology is appli-

cable to any problem domain wherein user characteristics must be inferred from a record of interactions with a software application, and it is this methodology that is presented as the main contribution of the work. As a secondary contribution, the model itself was described in depth, both in natural language form, and as an instance of the proposed argumentation framework.

Chapter 4

Evaluation

An evaluation strategy, scoped commensurately to the resources available, was designed and executed. This chapter begins with a delineation of this scope in section 4.1, which outlines the aims and limitations that were identified from the outset of the process. The evaluation activities undertaken are then described in sections 4.2 through 4.5, followed by an analysis of the results obtained in section 4.6. A design for a future, more conclusive, study is proposed in section 4.7, before the chapter closes with some discussion of the evaluation as a whole.

4.1 Scope

4.1.1 Aims

The aim of the evaluation undertaken was twofold. Firstly, to assess the ability of the system to identify exemplar users given simulated activity logs that represent an idealised population. Secondly, to assess the model itself, and to what degree the individual components of which it is comprised accurately capture the characteristics intended. Without real world data to exploit, a strategy of systematic isolation of individual model components was pursued to this end, with the simulated activity logs described above functioning as input to the system.

4.1.2 Limitations

There are a number of limitations to the approach taken, and they form the bounds of the evalauation scope. Given the resources available, the performance of the system cannot be assessed over a non-idealised population, or over real world data. Neither can the validity of the assumption that MS is a reliable or useful measure of user similarity in an RS. The viability of integrating the measures proposed by this work with a functioning RS, operating at scale, is similarly beyond the bounds of the current work. These tasks are left for future investigation, as described in section 5.2.

4.2 Host Application

The open source music player Clementine[1] was selected as the host application for the evaluation process. Clementine was chosen for its extensive feature set which enabled the collection of metrics around all three categories of interaction:

- **Listen**: Full support for both online streaming services and local files, with integrated equaliser, skip, seek, volume and other standard controls common to most full featured music players.
- **Organise**: Extensive library organisation utilities, including full metadata access, rating capability, playlists, cover management and varied library visualisation options.
- **Discover**: Integration with the Spotify search API and support for streaming online radio.

The source code of the player was downloaded and instrumented so as to populate an activity log with the metrics described in section 3.2. The source code was compiled and executed on Ubuntu Linux 12.04. See figure 4.1 for a screenshot of the application in use, and figure 4.2 for a representative snapshot of the activity log.

Fig. 4.1 The open source music application Clementine running on Ubuntu Linux.

4.3 Output Space Partitioning

The output space of the final reasoning stage was partitioned into 3 ranges: low $(0.00-0.24)$, middling $(0.25-0.74)$, and high $(0.75-1.0)$. See figure 4.3 for a graphical representation of the partitioning scheme.

Each of the exemplars (described in section 4.4) was assigned an expected partition. These partitions then served as a general guide during design iterations subsequent to the

[1] https://www.clementine-player.org/

```
"12345",04:49:07.843,start
"12345",04:49:07.937,set_volume,25
"12345",04:49:08.224,set_volume,25
"12345",04:49:08.288,disable_equaliser
"12345",04:49:08.288,adjust_balance,0
"12345",04:49:21.002,load,"file:///mnt/hgfs/Music/Ali Farka Touré/Ali Farka Touré/01 Timbarma.mp3",0,1,0,0
"12345",04:49:21.037,set_volume,25
"12345",04:49:21.037,disable_equaliser
"12345",04:49:21.037,edit_equaliser
"12345",04:49:21.037,adjust_balance,0
"12345",04:49:21.038,play
"12345",04:49:21.459,metadata,"Timbarma","Ali Farka Touré","Ali Farka Touré","Blues, Folk, World, &
Country","","1"
"12345",04:50:15.684,seek,0,53000000000
"12345",04:50:15.848,seek,0,159000000000
"12345",04:50:15.967,seek,0,239000000000
"12345",04:50:16.084,seek,0,254000000000
"12345",04:50:16.202,seek,0,278000000000
"12345",04:50:35.242,track_end
"12345",04:50:35.269,load,"file:///mnt/hgfs/Music/Ali Farka Touré/Ali Farka Touré/02 Singya.mp3",0,0,1,1
"12345",04:50:35.272,play
"12345",04:55:56.170,metadata,"Nawiye","Ali Farka Touré","Ali Farka Touré","Blues, Folk, World, &
Country","","3"
"12345",04:56:01.225,track_end
"12345",04:56:01.228,load,"file:///mnt/hgfs/Music/Ali Farka Touré/Ali Farka Touré/03 Nawiye.mp3",0,0,1,1
"12345",04:56:01.228,play
```

Fig. 4.2 A representative snapshot of the activity log.

initial evaluation. This feedback driven refinement procedure included modification to membership functions, parameterisation of operators, and adjustment to inference functions.

The introduction of partition boundaries is intended to aid in the refinement of the model through the provision of an approximate indicator of model accuracy. There is some circularity to this evaluation process, consequent to the fact that the evaluation is itself an integral part of the development process. The model has no definitively correct form, but it can be shaped towards an accurate approximation of its target. In this early stage evaluation, the manipulation of the model until it is judged to capture the characteristics of the exemplar users can be seen as a form of initialisation procedure, designed to establish a reasonable set of initial parameters from which a large scale study can proceed in future.

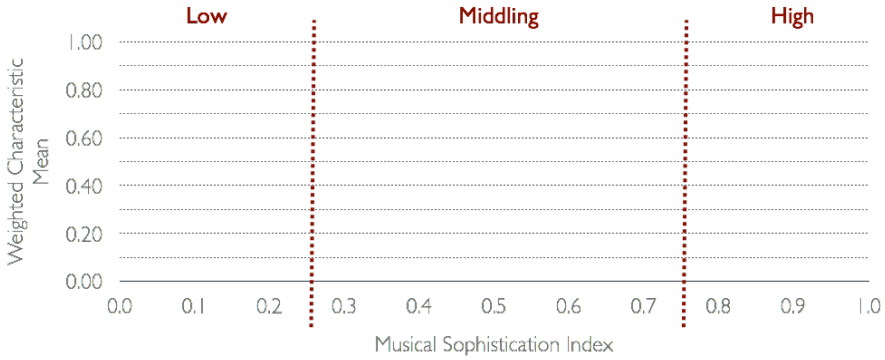

Fig. 4.3 The output space is partitioned into low, middling and high sophistication.

4.4 Exemplar Users

A set of exemplar users was developed, each designed to exercise particular facets of the model. Each exemplar was assigned an expected output partition, in which the MS index resultant from their behaviour could be expected to lie (see section 4.3). The behaviour of these users was then simulated using the host application, and the system executed over the resultant logs.

4.4.1 User A - High Sophistication

User A exemplifies a highly sophisticated user. The activities of this user indicate strong focus, specificity, organisation and curiosity. These characteristics are somewhat contradictory in nature and are not exhibited simultaneously. Periods of intense focus are interspersed with discovery and organisational activities. When this user listens, they do so with focus, however particular periods are set aside for supporting activities such as metadata maintenance and knowledge expansion. The behaviour of this user typifies that which is expected to result in high scores across all characteristics, and ultimately a high MS index.

4.4.2 User B - Low Sophistication

User B exemplifies a user with low levels of MS. This user is passive in their approach to music. Choice of listening material is dictated primarily by chance, or in response to transitive moods. Little effort is expended discovering new music, and none in tasks secondary to listening, such as metadata management, or playlist maintenance. Radio streams are absorbed passively with no indication of subsequent pursual. Overall listening patterns exhibit low variance over time, and a tendency towards popular material.

4.4.3 User C - Middling Sophistication I

User C typifies a user highly engaged in discovery of new music. This activity is to the detriment of focus however. The nature of this user's knowledge can be characterised as "breadth over depth". Frequent search executions, high listening rates, use of discovery services with subsequent pursual, and a listening history that exhibits a wide range of genres, artists and albums, are some of the traits that contribute to a positive MS score. However, low levels of focus, indicated by a lack of session continuity, a timeline with no discernible clustering and little attention to library management, all contribute negatively to an expected medium MS rating.

4.4.4 User D - Middling Sophistication II

User D stands in contrast to *User C*, as an exemplar of a highly focused user with little interest beyond the bounds of a limited set of artists and genres. Discovery is not a prime motivator in the interactions of this user. Search frequency tends towards zero. Little, or no, usage of radio and other discovery services is exhibited. However latent information in the listening patterns of this user points to a systematic, and methodical approach to music that indicates focus and concentration. Session continuity with regard to both artists and albums is high. Clustering is exhibited in listening patterns, and a subset of albums is rotated over the course of time as the user integrates the work of particular artists into their existing knowledge. The knowledge of this user is best characterised as "depth over breadth", and

their activities demonstrate an alternate pattern of behaviour that would be expected to lead to a medium MS rating.

4.5 Activity Logs

The activity of each the exemplars was simulated through extended periods of usage of the host application. Approximately 3 hours usage per exemplar resulted in an average activity log size of 97Kb. The size of the individual logs is listed in table 4.1. This relatively small time window, in comparison to that of a real world implementation, necessitated some adjustment to model parameterisation in order to reflect the different expectations that arise from briefer usage periods. This is an unavoidable consequence of the evaluation strategy, but one that does not significantly impact on the conclusions that can be drawn from the results.

User	Log Size (Kb)
User A	152
User B	54
User C	109
User D	73

Table 4.1 Simulated activity log sizes per exemplar

Computational efficiency was deemed of low priority during development, however some tentative experimentation was performed with larger log sizes in order to gauge the performance of the operators under such circumstances. This data was generated through a process of replication and partial randomisation of the individual logs. While not entirely reflective of real world data patterns, the resultant files were fit for purpose in that they were of sufficient size to exercise the operators for extended periods of time. This enabled an assessment of operator efficiency when processing large data sets, and exposed some inefficiencies in the initial architecture, which are described in section 4.6.6.

4.6 Results

This section contains a description and analysis of the results obtained when the system was executed over the exemplar logs. The final MS indexes produced for each exemplar were first compared and assessed as to their correlation with expectations. The component parts of the model were then examined in turn in order to determine the relative contribution of the characteristics in isolation. This was accomplished through a process of systematic disabling of individual characteristics, and analysis of the consequent effect on the system's inferences.

4.6.1 Musical Sophistication

Table 4.2 lists the MS index ultimately attributed to each of the exemplars, the contributory characteristic indicators, the weights associated with each characteristic, and the inputs and outputs to the membership function of the final reasoning stage. That membership function is illustrated, with values for each of the exemplar users highlighted, in figure 4.4.

	Organisation	Focus	Specificity	Curiosity	Membership Input	Membership Output
Weights	0.9	1.0	0.6	0.7		
User A	0.56	0.64	0.30	0.44	0.51	0.88
User B	0.15	0.18	0.14	0.16	0.16	0.24
User C	0.32	0.35	0.34	0.56	0.39	0.60
User D	0.33	0.71	0.42	0.27	0.45	0.71

Table 4.2 The input and output values in the final reasoning stage, including characteristic indicators, associated weights and membership function input and output values.

As seen in figure 4.4, the MS scores attributed to the exemplars fall within the expected partitions. The significance of this result should not be overstated, however, as the exemplars themselves contributed to the refinement of the model over the course of the evaluation.

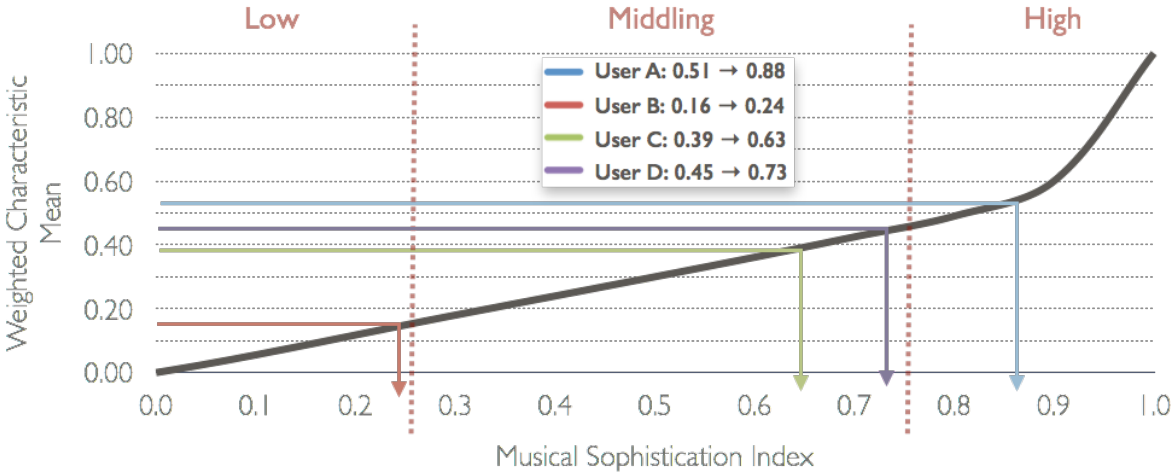

Fig. 4.4 The MS membership function with exemplar user input and output values indicated.

The curve of the membership function, and the placement of the partition boundaries, evolved out of the iterative design process described in sections 3.1 and 4.3. The changes in slope are a reflection of the constrained output space of the reasoning stage which tends to lie between 0.1 and 0.5. The membership function spreads the output of the reasoning stage across the confidence range of 0.0 to 1.0 in order to more accurately express the deductions of the system.

See figure 4.5 for an alternative visualisation of the results, in the form of a kiviat graph. Each characteristic lies on a separate axis of the graph and the scores of the 4 exemplars are

overlaid. The larger the area occupied by a user over the entirety of the graph, the higher the attributed MS index, with the caveat that the graph does not reflect the weighting of the final inference function, nor the output spread of the membership function.

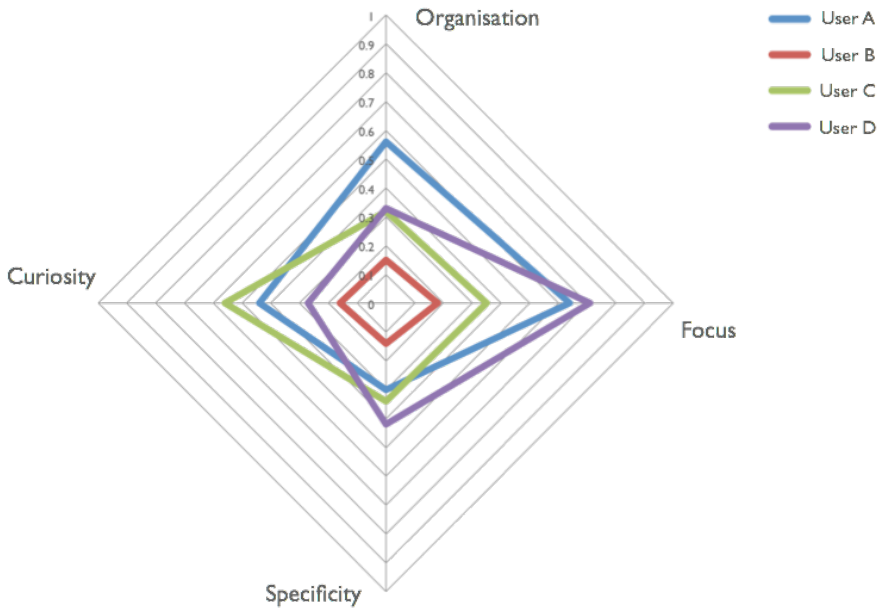

Fig. 4.5 The larger the area occupied by a user on the graph, the higher the attributed MS index.

Insofar as the exemplars fall within expected partitions, the model can be deemed to capture the MS of this particular data set. In that respect the stated aim of the evaluation, particularly to assess the efficacy of the system over an idealised population, is achieved. Whether this same ability holds true over a non-idealised population cannot be known without further investigative work, such as that described in section 4.7. It is inevitable that such work would lead to further refinement of the model, as the complexity of real world data would furnish the designer and implementer with supplementary domain specific knowledge.

The following sections, 4.6.2 through 4.6.5, contain further analysis of the individual characteristic indicators attributed to the exemplars. The attribute values of which focus, curiosity, organisation and specificity are comprised are listed in tables 4.3, 4.4, 4.5 and 4.6 respectively. Aggregation was performed according to the reasoning rules outlined in section 3.5 and ultimately produced the characteristic indicators, which are listed in the bottom row of each table. It is these values that served as input to the final stage of reasoning, as outlined above.

4.6.2 Focus

Table 4.3 details the individual attribute values that were aggregated to form the exemplar's focus indicators. Of note is that the aggregated scores, listed in the bottom row of the table, correlate quite closely with the contrived exemplar backstory in each case. The significance of this is tempered by the fact that the actions of the users were simulated in a research

environment, and guided by the backstories. A discussion of this circularity, inherent to the form of evaluation undertaken, is to be found in sections 4.1 and 4.8.

Attribute	User A	User B	User C	User D
Radio usage	0.00	0.65	0.78	0.00
Completism	0.73	0.25	0.36	0.78
Whole albums	0.63	0.24	0.42	0.89
Low volume	0.11	0.23	0.18	0.05
Long term album variance	0.73	0.64	0.89	0.78
Short term album variance	0.36	0.79	0.76	0.26
Song continuity	0.36	0.13	0.64	0.46
Album continuity	0.82	0.19	0.48	0.76
Artist continuity	0.91	0.23	0.69	0.78
Playlist maintenance	0.51	0.00	0.56	0.46
Artist focused playlists	0.92	0.00	0.43	0.86
Genre focused playlists	1.00	0.00	0.78	0.86
Period focused playlists	0.38	0.00	0.87	0.86
Focus	0.64	0.18	0.35	0.71

Table 4.3 The focus attributes calculated for the exemplars.

In order to gauge the impact of focus on the MS indexes ultimately assigned to the exemplars, the system was executed over the same data with that characteristic disabled. The results can be seen in the form of the membership function of the final stage of reasoning in figure 4.6.

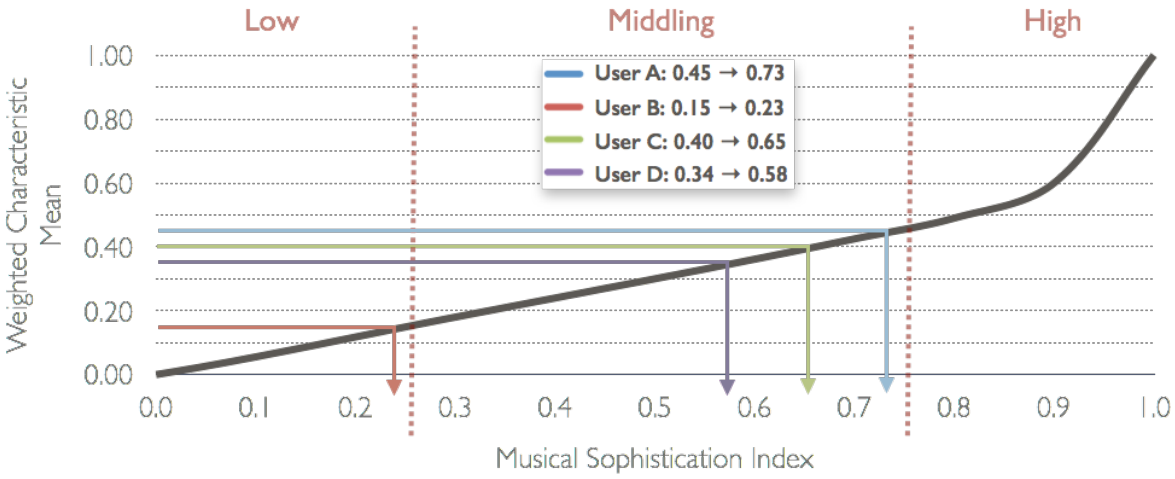

Fig. 4.6 The MS membership function when focus is disabled.

A number of points of interest arise from a comparison of the output of this function with that obtained when all four characteristics are enabled (see figure 4.4):

- User A drops from the high partition to the middling partition, but remains the highest ranked of the 4 with a decrease in score from 0.88 to 0.73. This falls just below the high partition boundary.
- User C rises from 0.60 to 0.65, above User D. This is in keeping with the associated backstory of a highly curious user who lacks focus.
- User D drops from 0.71 to 0.58. This drop is also in keeping with the respective backstory of a highly focused user . The drop is not as significant as the contrived user history would indicate however, and this may point to an opportunity for further refinement of the model.
- The ranking of User B is not altered significantly. This may be attributable to the generally low rankings of User B, with no particular characteristic expressed strongly in the associated backstory.
- In general terms, the rise in the score of User C, and the drop in that of User D, while definite, are not as pronounced as predicted by the respective backstories. This may indicate a need for modification to the slope of the membership function to further magnify the spread of the output range of the inference function.

See figure 4.6 for an alternative view of the data. This visualisation highlights curiosity as the single most significant contributor to MS when focus is disabled. This insight serves as an opportunity to demonstrate the iterative feedback process described throughout this document: An observation such as this could indicate that the curiosity component of the model is excessively sensitive or is reacting to false positives. This is also a possible explanation for the compressed output range of the inference function noted above. Theories and insights such as these are gathered in the evaluation stage, and guide the activities of subsequent design iterations. Ensuing evaluations serve to validate these theories and generate new ones in an ongoing optimisation process.

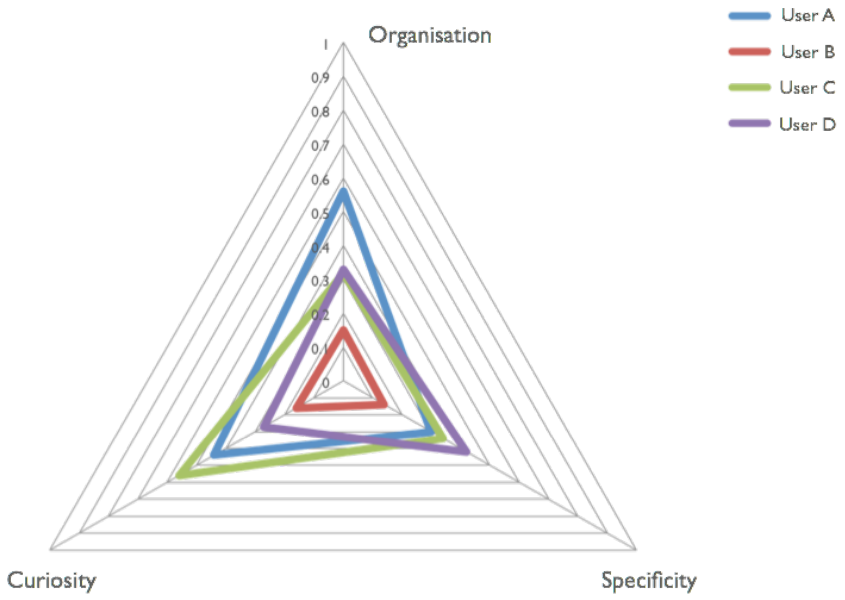

Fig. 4.7 Results when the focus component of the model is disabled.

4.6.3 Curiosity

The constituent attributes of curiosity, the characteristic with which the backstory of User C is most closely associated, are detailed in table 4.4. As in the case of focus, the aggregated scores, listed in the bottom row of the table, correlate closely with the associated backstory of each exemplar, however the same caveat noted in section 4.6.2 applies.

Attribute	User A	User B	User C	User D
Popularity	0.36	0.89	0.58	0.38
Radio usage	0.21	0.65	0.78	0.00
Pursual	0.06	0.00	0.65	0.00
Search frequency	0.68	0.38	0.86	0.18
Search variance	0.63	0.24	0.79	1.00
Artist variety	0.68	0.79	0.76	0.26
Album variety	0.69	0.13	0.64	0.38
Genre variety	0.72	0.19	0.48	0.21
Curiosity	0.44	0.16	0.56	0.27

Table 4.4 The curiosity attributes calculated for the exemplars.

See figure 4.8 for the membership function of the final stage of reasoning when curiosity is disabled. Some points of note that arise from an analysis of these results:

- User D rises to the high partition due to a significant increase from 0.73 to 0.85. This can be attributed to the low curiosity score predicted by the respective backstory.
- User C falls from 0.63 to 0.58, a definite decrease but not as significant as could be expected given the primary characterisation of the user as highly curious.
- User A increases slightly from 0.88 to 0.89 in keeping with the roughly equal contributory impact of the 4 characteristics in the associated backstory.
- User B is unchanged, again reflecting the roughly equal attribution of characteristics to this exemplar.

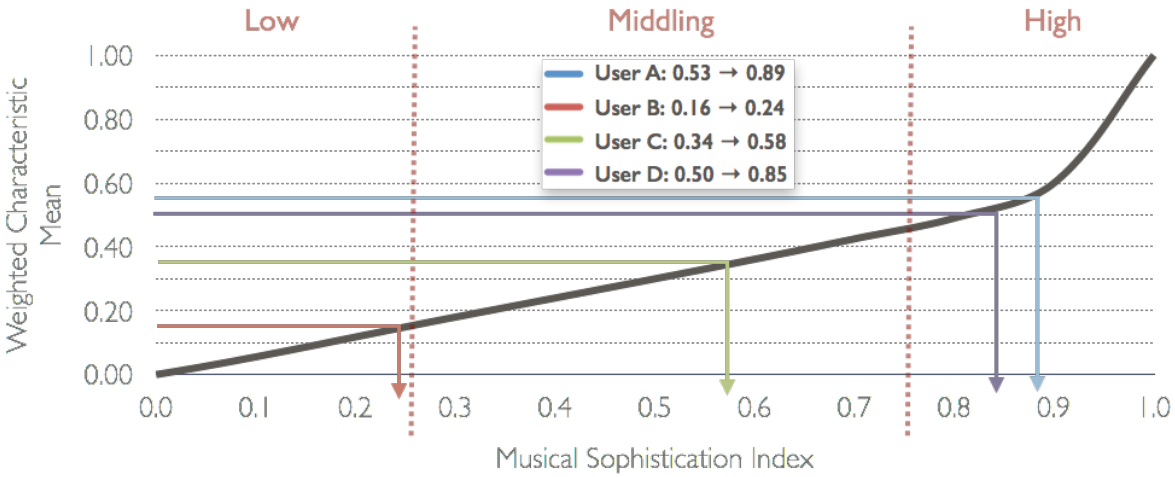

Fig. 4.8 The MS membership function when curiosity is disabled.

See figure 4.9 for the kiviat representation of the results. An examination of this visualisation can provide some indication why the drop in the score of User C was relatively moderate, despite the removal of the primary associated characteristic. The triangle formed by the user on the kiviat graph exhibits roughly equal lengths along each side, indicating equal contributions from the 3 other characteristics. This may indicate that either the lack of focus described in the backstory was not sufficiently expressed in the simulated activity, or that the focus component in the model requires adjustment in the proceeding design iteration. While this highlights a deficiency in the system, it also serves as another example of the type of feedback based, iterative refinement process that is integral to this methodology.

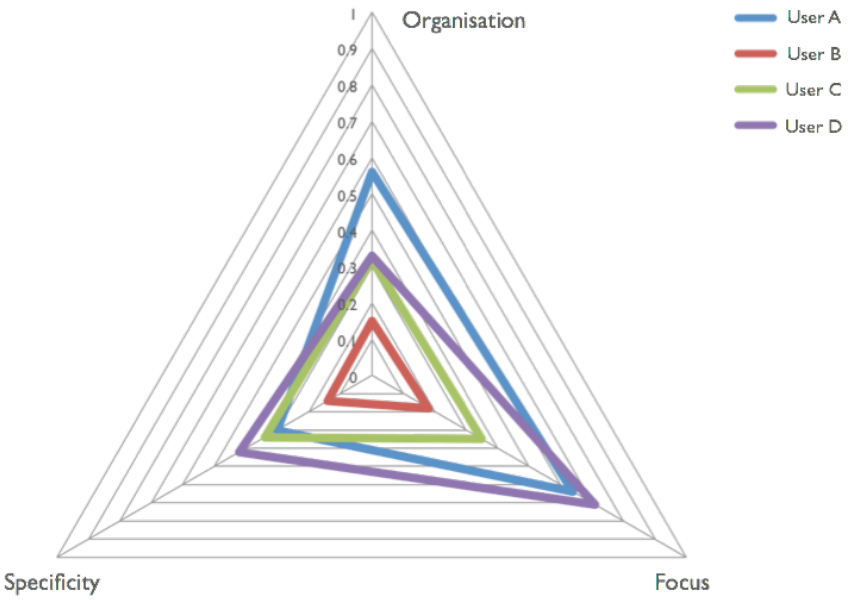

Fig. 4.9 Results when the curiosity component of the model is disabled.

4.6.4 Organisation

In adherence to the structure established in the preceding sections, the measurements that comprise the organisation scores of the exemplars are detailed in table 4.5, and the member-

ship function and kiviat graph that result when the characteristic is disabled, in figures 4.10, and 4.11 respectively.

In contrast to focus and curiosity, no exemplar was assigned particular organisational tendencies in the associated backstories. Of note when the characteristic is disabled:

- User D increases significantly from 0.73 to 0.84. This increase is equal in magnitude to that of the same user when curiosity was disabled, which was specifically noted as present in low measure in the associated backstory. This anomaly may be attributable to a curiosity of the simulated activity, or it may indicate false positives in the organisation component. Further investigation is required.
- The remainder of the users are largely unchanged with the exception of User A with a moderate increase. This is in keeping with expectations due to the lack of detail with respect to the characteristic in the backstories.

Attribute	User A	User B	User C	User D
Saves playlists	0.58	0.00	0.56	0.38
Creates playlists	0.44	0.00	0.56	0.54
Edits genre	0.68	0.00	0.00	0.20
Genre explosion	0.26	0.56	0.68	0.35
Edits metadata	0.73	0.10	0.23	0.51
Missing metadata	0.27	0.38	0.79	0.26
Organisation	0.56	0.15	0.32	0.33

Table 4.5 The organisation attributes calculated for the exemplars.

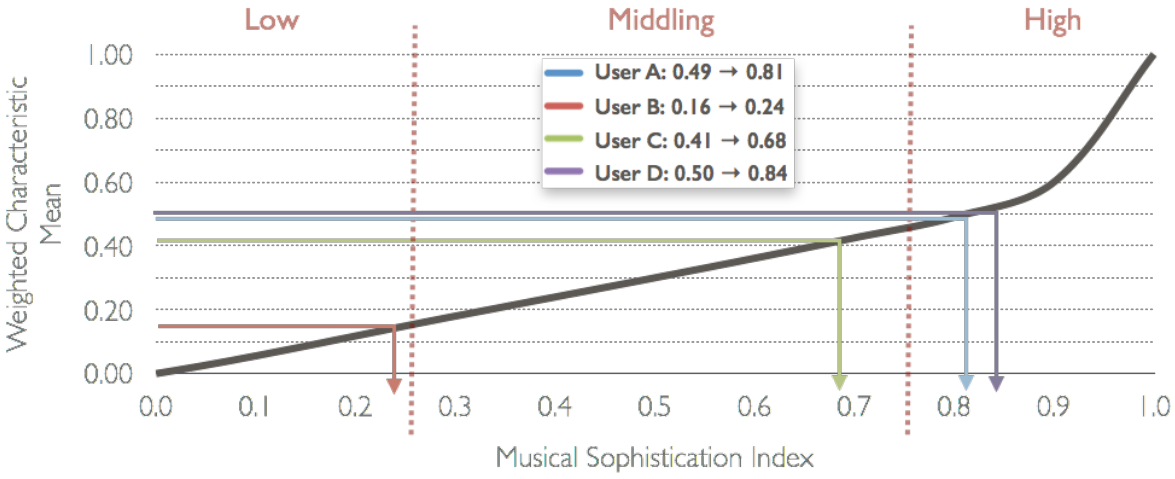

Fig. 4.10 The MS membership function when organisation is disabled.

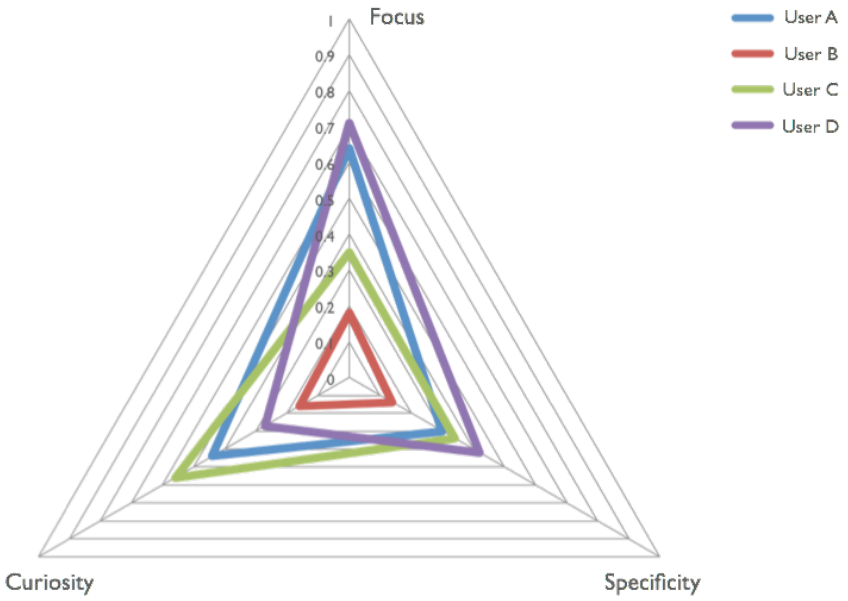

Fig. 4.11 Results when the organisation component of the model is disabled.

4.6.5 Specificity

The measurements that comprise the specificity scores of the exemplars are detailed in table 4.6, and the membership function and kiviat graph that result when the characteristic is disabled, in figures 4.12 and 4.13 respectively.

Attribute	User A	User B	User C	User D
Start from beginning	0.70	0.78	0.36	0.00
High bitrate encoding	0.64	0.15	0.24	0.63
Artist search	0.48	0.66	0.67	0.69
Song search	0.20	0.23	0.89	0.00
Album search	0.73	0.00	0.47	0.41
Seek	0.00	0.00	0.26	0.14
Skip	0.36	0.26	0.33	0.28
Specificity	0.30	0.14	0.34	0.42

Table 4.6 The specificity attributes calculated for the exemplars.

Of note in table 4.6 is the generally low aggregated score across all users. This may point to a need for an increase in sensitivity to the specificity component, possibly through adjustments to membership functions at one or more stages of the reasoning process. The introduction of an additional exemplar, designed to exercise this aspect of the system more thoroughly could elicit further insight in this respect. The generally low indicators attributed for this characteristic are also reflected in the membership function, seen in figure 4.12. An alternate explanation is that the characteristic is not in fact a meaningful component of

musical sophistication in isolation, and should be merged with one or more of the other characteristics. The results outlined in this section are the most dissatisfactory of those obtained, and further work is required in regard to this characteristic.

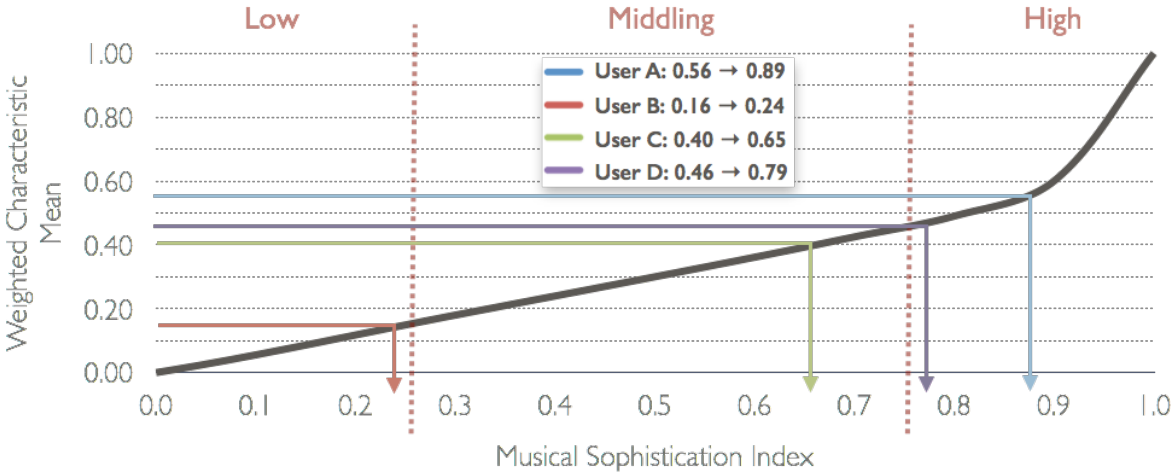

Fig. 4.12 The MS membership function when specificity is disabled.

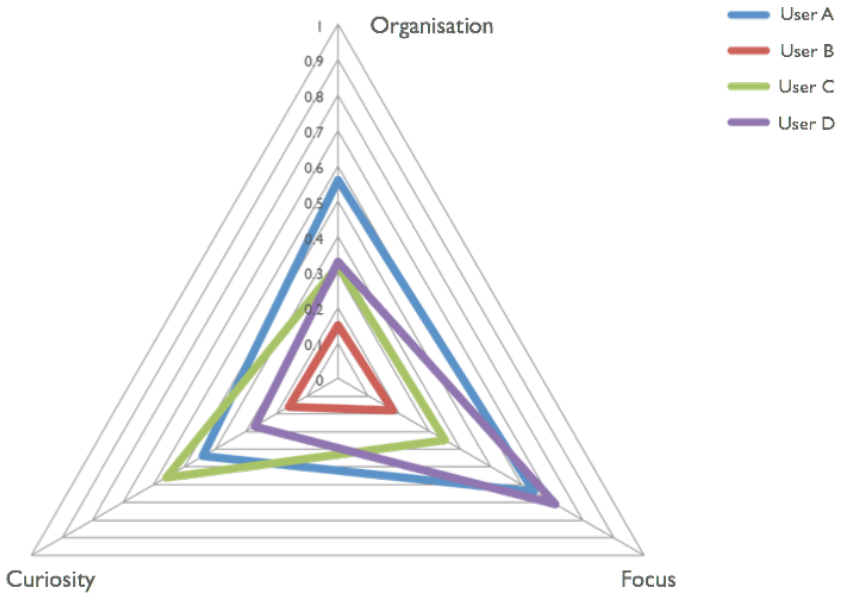

Fig. 4.13 Results when the specificity component of the model is disabled.

4.6.6 Computational Efficiency

Computational efficiency was not deemed a development priority. Nevertheless some tentative assessment of the system's ability to process larger data sets was undertaken. The average log size resultant from 3 hours of simulated exemplar usage informed an approximation of that which could be expected to result from 1 day, from 3 days and from 1 week of user activity. Commensurately sized logs were then generated through a process of replication and partial randomisation of the exemplar data. Performance figures for each of the those logs are provided in table 4.7.

Time Window	Log Size (Mb)	Processing Time (Minutes)
1 day	~0.75	~1
3 days	~2.25	~3
1 week	~5.25	~8

Table 4.7 Performance over larger log sizes.

While the data that was utilised for this assessment is not truly representative of the patterns of real world interactions, it was sufficient to exercise the operators at an appropriate load over an extended period of time. The performance of the current implementation would not scale to a real world system, however processing time increases roughly linearly with log size, which indicates optimisation and parallelisation could deliver an acceptable implementation.

The evaluation highlighted some inefficiencies in the initial system design and led to modification of the log processing architecture. In the original architecture each operator maintained an independent log file handle, with the intention that its speed be bounded only by the complexity of its computational logic, and independently of other operators. This was found to cause thrashing due to cache complications introduced when multiple processes accessed neighbouring disk areas simultaneously. Consequently the design was modified so as to synchronise the operators at each log entry, and a coordinator was introduced to manage a single handle to the log file. See figure 4.14 for an illustration of these modifications.

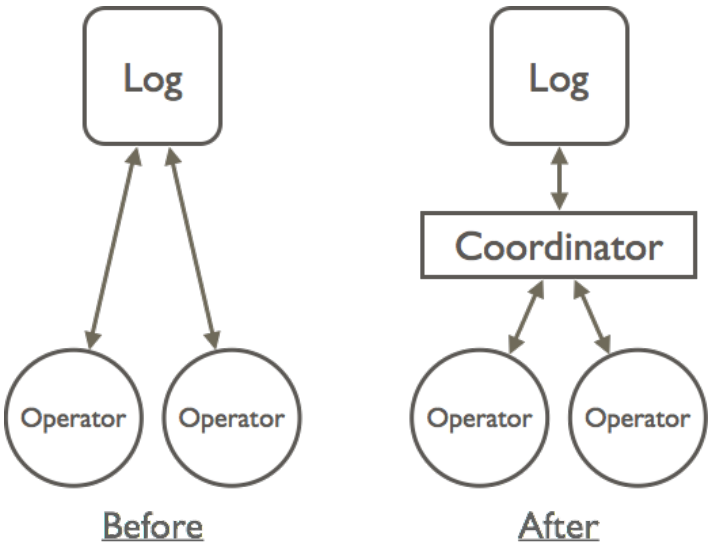

Fig. 4.14 Evaluation of computational efficiency led to some modifications to the processing architecture.

4.7 Proposal for Future Evaluation

An optimal evaluation strategy would include a large scale use case study with the aim of assessing the efficacy of the system over a non-idealised population. Time and resource

constraints preclude the execution of such a study within the scope of the current work. In this section, the outline of a design for such a study is proposed, the implementation of which is left for a later work.

4.7.1 Aims & Limitations

The primary aim of a large scale use case study would be to assess the feasibility of inferring MS from user behaviour, or at least to produce predictions that correlate with the results of alternative assessment procedures such as those discussed in section 2.1. As a consequence of the design-driven methodology employed, evaluation activities advance a secondary goal of continued development of the model, and an integral component of the evaluation strategy described in section 4.7.2 is a refinement of the initial model over the course of the study.

The proposed strategy does not, and could not, aim to validate the assumption underlying the motivation for this work, specifically the usefulness of MS as a measure of user similarity. Neither would it assess the viability of implementing the techniques of this work in an RS operating at scale. These are left as open tasks, as detailed in section 5.2.

4.7.2 Methodology

Host application

It is suggested that an open source music application, such as that described in section 4.7.2, be instrumented with the logging capabilities outlined in section 3.2, and then compiled and distributed to a set of study participants.

Participant selection & partitioning

It is suggested that participants be divided into 2 groups of equal size: *group 1*, selected so as to represent as broad a cross section of typical users as practical; and *group 2*, a random selection of subjects from the available pool. This partitioning of participants is intended to mitigate any subconscious effects of the initial assessment necessitated by the selection process, and also to provide a control group to aid in the assessment of iterative revisions to the model performed as the study progresses. In order to ensure a wide range of MS levels in group 1, a mixture of self-assessment and standardised tests, such as those described in section 2.1 is advised. However the administration of such assessments could potentially impact the subsequent behaviour of the participants by drawing their attention to the behavioural traits of interest. In order to mitigate this potential adulteration of the sample set, it is advised that participants assigned to group 2 undergo no assessment prior to, or during, the usage period.

Usage period

An extended usage period is recommended, during which time participants agree to use the host application as their primary music listening and management application. There are 2 options by which the length of the usage period could be governed. The first is a log size threshold, whereby the actual time period would vary by participant but would result in the same quantity of data for each. Alternatively, a more practical but less normalised strategy would employ a predetermined window of time applied equally to all participants.

It is advised that the activity logs of group 1 be harvested at regular intervals during the usage period for assessment by the system, and analysis of the results fed back into additional design iterations. Any modifications to the host application necessitated by these additional iterations should be distributed to all study participants.

Evaluation

After the conclusion of the usage period all participants, irrespective of assigned group, should be administered the same battery of assessments as administered to group 1 prior to, and throughout, the usage period. The final set of activity logs should be collected from all participants and subsequently assessed according to the model derived in the final design iteration. Some suggestions for analysis of the resultant data include:

- Identification of correlations between final assessment results and system predictions, where higher rates of correlation would indicate higher accuracy in the final model.
- Comparison of the accuracy of system predictions with respect to group 1 and group 2. This analysis would highlight the degree to which a model refined with a small dataset can be generalised to a larger population.
- Analysis of the effect of design iterations on the predictions with regard to group 2. This analysis would aid in the further assessment of the effectiveness of the feedback loop from evaluation to design. Incremental improvements in accuracy would point to an effective model optimisation mechanism.

4.8 Conclusion

This chapter began with section 4.1 which constituted a clear delineation of scope and set forth the bounds of the evaluation strategy undertaken. Sections 4.2, 4.4 and 4.6 detailed the evaluation process itself, the results obtained, and the information revealed in an analysis of those results.

The undertaking was ultimately judged successful within the bounds of the stated aims. The system successfully rated exemplars within expected ranges, although the caveat was noted that the evaluation itself informed the design of the model. A systematic isolation and assessment of each of the model components yielded results indicating value in each component part, albeit with some reservations as to the role of specificity. The methodology,

comprised in part of the evaluation process itself, can be judged independently of model accuracy. In this respect the research was judged a success by merit of the fact that the design, development, evaluation and subsequent iterative optimisation, proved a viable and intuitive means of model development in this context.

There is an inherent circularity to the evaluation of a project such as the current undertaking. Without a use case study, unequivocal evaluation results are an impossible goal. However without an initial research project to assess the merits of the idea itself, an undertaking on the scale required for a meaningful use case study will rarely be justified economically. In acknowledgement of this status as an early stage preliminary piece of research, section 4.7 described an alternative evaluation approach less encumbered by the circularity imposed by a lack of real world data. It is hoped that a future implementation of that proposal will lead to more definitive conclusions with regard to both the model itself and the viability of the methodology as a whole.

Chapter 5

Conclusion

This chapter begins with a summary of the main findings of the work in section 5.1. A number of tasks remain open as opportunities for future research and these are outlined in section 5.2 before the work concludes with a discussion of this piece of multidisciplinary research from a holistic perspective in section 5.3. This section includes some final thoughts on the questions that remain open, the conclusions that can legitimately be drawn from the results, and an assessment of the success of the project as a whole.

5.1 Summary of Results

A methodology is presented in this work for the construction of a computational model of the outward expression of internal characteristics that are relevant to the interactions between a human and a software application. The system accepts a natural language description of expected behaviour and a log of user activity, and generates a single numerical index representing its confidence in the presence of the characteristic. Additionally, a model of musically sophisticated behaviour and a system designed to detect it, are presented, as constructed through the application of the described methodology.

The resultant system is assessed against an idealised population and found to achieve its stated aims. A set of exemplar users are identified correctly by the system given log files produced through the simulation of exemplar behaviour. These results indicate a successful application of the methodology which was developed over the course of the work, and the resultant system stands as a proof of concept from which future research can proceed.

5.2 Future Work

This work is presented as an initial step towards the ultimate goal of a finely tuned, accurate model of MS, evaluated solely from user behaviour, and enabling an RS to generate suitably challenging, serendipitous recommendations. Constraints with respect to both time and resources necessitated a scope significantly limited in nature by comparison to the scale of this goal. Consequently, a number of opportunities for future research arise.

5.2.1 Use Case Based Evaluation

The most pressing requirement prior to further development of the proposal is a large scale user study to assess the level of correlation between the results produced by this system and those of alternative MS assessments. Despite the positive results of the current work, the possibility nevertheless exists that the extraction of a complex and loosely defined characteristic such as MS from real world data is simply not a viable proposition. A use case study is required in order to assess the applicability of the findings of this work outside of a controlled research environment. Such a study would also indicate the potential of the system to operate at a scale beyond a single user, and provide opportunities for optimisation of the data analysis process. One potential design for such a study is suggested in section 4.7.

5.2.2 Validation of Assumptions

A further opportunity is presented by the assumption that underlies the motivation for the work, albeit one most suited to the field of music psychology. A participatory study to evaluate correlation between MS levels and subjective response to individual pieces of music would allow the potential utility of any RS that were to utilise the proposed technique to be appraised.

5.2.3 Recommender System Integration

A final task beyond the scope of the current work, and dependent on the successful validation and evaluation of the proposal as described above, is the integration of the described techniques into an operational RS environment. The design and implementation of an RS capable of effectively utilising MS as a similarity measure is a task of significant scale, and encompasses integration with existing RS technologies, and development of a scalable parallelised activity log processing architecture among its many component parts. Beyond such details of implementation, the evaluation of RS technology is a subject of considerable research in itself, and is particularly problematic when serendipity is a primary goal. While others, such as McNee et al. (2006) and Zhou et al. (2010), have suggested evaluation strategies that take account of diversity and serendipity, these techniques are less well established than standard accuracy measures. Within this context any potential implementation would benefit from further work towards a clear evaluation strategy appropriate to the project goals, and meaningful, pre-established metrics with which to assess the effectiveness of any results.

5.3 Final Words

A multidisciplinary literature review set the context for the work in chapter 2. The concept of MS, as it is described in the literature of music psychology, was introduced in section 2.1. A number of alternative conceptions of MS were compared and contrasted with a view to revealing those facets of its nature that most relate to the aims of this research. In particular,

those components that relate to aural skills and subjective response to music were judged to be of significance, and aided in the adoption of an appropriately bounded interpretation of the term. Analysis of this body of preceding work provided a solid theoretical grounding from which to establish an initial model of the behavioural expression of MS, and was necessitated by the perceived ambiguity inherent to the concept amongst those not versed in the literature of music psychology.

Section 2.2 contained an investigation of the techniques of defeasible reasoning and fuzzy logic. This investigation provided direction in the task of reasoning computationally from a set of fuzzily defined heuristics expressed in natural language.

A review of recommender system technology in section 2.3 served to situate the proposals of this work within an established field of research, and to highlight alternative techniques currently in widespread commercial use.

Chapter 3 detailed the core research activities undertaken. Key activities described include the development of a particular model of musically sophisticated behaviour, and the derivation of a computational argumentation framework from that model, which was initially described as a set of natural language propositions. A description of, and rationale for, the design-driven approach taken was outlined in depth. Due to the close coupling between design and implementation resultant from this strategy, the two were described as one in this chapter, in a structure that mirrored the mixture of top-down, bottom-up design and development that characterised the activities of the work.

Chapter 4 described the activities undertaken to evaluate the model and methodology described in the preceding chapter. A set of exemplar users was developed, and the degree to which the system accurately identified them from a set of simulated activity logs was assessed. The relative contribution of the each of the components of the model was then evaluated through a process of isolation and individual analysis. In acknowledgement of the limitations enforced by the research context, an alternative evaluation strategy was proposed and suggested as a design for future implementation with more conclusive evaluation results.

The nature of the research project undertaken is such that unequivocal conclusions as to model accuracy and real world performance were beyond scope from the outset. Nevertheless, as an initial investigation into the viability of the concept, the work has proved successful in its aims. A system and methodology were ultimately produced that serve as a positive proof of concept, and demonstrate that the recognition of internal characteristics as they are expressed outwardly through interactions with a piece of software is not an infeasible goal. The ability of the system to identify exemplars in an idealised population indicates the methodology itself is sound, and that with sufficient refinement of the model the automated inference of musical sophistication from user behaviour may indeed prove a viable objective.

References

Robin Burke. Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, 2002.

John Canny. Collaborative Filtering with Privacy via Factor Analysis. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 238–245, New York, NY, USA, 2002. ACM.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal Of The American Society For Information Science*, 41(6):391–407, 1990.

Daniel Fleder and Kartik Hosanagar. Blockbuster Culture's Next Rise or Fall: The Impact of Recommender Systems on Sales Diversity. March 2009.

John Fox, Paul Krause, and Morten Elvang-Gøransson. Argumentation as a general framework for uncertain reasoning. In *Proceedings of the Ninth international conference on Uncertainty in artificial intelligence*, pages 428–434. Morgan Kaufmann Publishers Inc., 1993.

S Hallam. 21st century conceptions of musical ability. *Psychology of Music*, 38(3):308–330, June 2010.

Susan Hallam and Vanessa Prince. Conceptions of musical ability. *Research Studies in Music Education*, 20(1):2–22, 2003.

Trevor Hastie, Robert Tibshirani, Jerome Friedman, T Hastie, J Friedman, and R Tibshirani. *The elements of statistical learning*, volume 2. Springer, 2009.

Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen, and John T Riedl. Evaluating Collaborative Filtering Recommender Systems. *ACM Trans. Inf. Syst.*, 22(1):5–53, January 2004.

Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. *Recommender Systems An Introduction*. Cambridge University Press, 2011.

Luca Longo. *Formalising Mental Workload as a Defeasible Computational Concept*. PhD thesis, Trinity College Dublin, Dublin, 2012.

S M McNee, J Riedl, and J A Konstan. Being accurate is not enough: how accuracy metrics have hurt recommender systems. *CHI'06 extended abstracts on Human ...*, 2006.

Daniel Müllensiefen, Bruno Gingras, Jason Musil, and Lauren Stewart. The musicality of non-musicians: an index for assessing musical sophistication in the general population. *PloS one*, 9(2):e89642–23, 2014.

Tomoko Murakami, Koichiro Mori, and Ryohei Orihara. Metrics for Evaluating the Serendipity of Recommendation Lists. pages 40–46, 2008.

Donald Nute. Defeasible Reasoning: A Philosophical Analysis in Prolog. 1(Chapter 9): 251–288, 1988.

Joy E Ollen. A criterion-related validity test of selected indicators of musical sophistication using expert ratings. pages 1–261, 2006.

G Géza Révész. *Introduction to the psychology of music*. London : Longmans, Green and Co, 1953.

Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Analysis of Recommendation Algorithms for e-Commerce. In *Proceedings of the 2Nd ACM Conference on Electronic Commerce*, pages 158–167, New York, NY, USA, 2000a. ACM.

Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Application of dimensionality reduction in recommender system-a case study. Technical report, 2000b.

Andrew I Schein, Alexandrin Popescul, Lyle H Ungar, and David M Pennock. Methods and Metrics for Cold-start Recommendations. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 253–260, New York, NY, USA, 2002. ACM.

Carl E Seashore. The Measurement of Musical Talent. *The Musical Quarterly*, 1(1):129–148, January 1915.

F Toni. Argumentative agents. In *Computer Science and Information Technology (IMCSIT), Proceedings of the 2010 International Multiconference on IS* -, pages 223–229, 2010.

Slobodan Vucetic and Zoran Obradovic. Collaborative Filtering Using a Regression-Based Approach. *Knowledge and Information Systems*, 7(1):1–22, February 2004.

L A Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.

L A Zadeh, G J Klir, and B Yuan. *Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers*. Advances in fuzzy systems : applications and theory. World Scientific, 1996.

Shanshan Zheng, Tao Jiang, and J S Baras. A Robust Collaborative Filtering Algorithm Using Ordered Logistic Regression. In *Communications (ICC), 2011 IEEE International Conference on*, pages 1–6. IEEE, 2011.

T Zhou, Z Kuscsik, J G Liu, and M Medo. Solving the apparent diversity-accuracy dilemma of recommender systems. In *Proceedings of the . . .* , 2010.