The Knowledge Engineering Review, Vol. 30:2, 140–156. © Cambridge University Press, 2015
doi:10.1017/S0269888914000265

Personalization and rule strategies in data-intensive intelligent context-aware systems

PHILIP T. MOORE[1] and HAI V. PHAM[2]

[1]*Ubiquitous Awareness and Intelligent Solutions Lab (UAIS), School of Information Science and Engineering, Lanzhou University, Lanzhou, P. R. China; Shandong Normal University, Jinan City, P. R. China;*
e-mail: philip.moore@mail.bcu.ac.uk;
[2]*Information Systems Department, School of Information Technology and Communication, Hanoi University of Science and Technolgy, Hanoi, Vietnam; Soft Intelligence Lab, Graduate School of Science and Engineering, Ritsumeikan University, Shiga, Japan;*
e-mail: hai@spice.ci.ritsumei.ac.jp

Abstract

The concept of personalization in its many forms has gained traction driven by the demands of computer-mediated interactions generally implemented in large-scale distributed systems and *ad hoc* wireless networks. Personalization requires the identification and selection of entities based on a defined profile (a context); an entity has been defined as a person, place, or physical or computational object. Context employs contextual information that combines to describe an entities current state. Historically, the range of contextual information utilized (in context-aware systems) has been limited to identity, location, and proximate data; there has, however, been advances in the range of data and information addressed. As such, context can be highly dynamic with inherent complexity. In addition, context-aware systems must accommodate constraint satisfaction and preference compliance.

This article addresses personalization and context with consideration of the domains and systems to which context has been applied and the nature of the contextual data. The developments in computing and service provision are addressed with consideration of the relationship between the evolving computing landscape and context. There is a discussion around rule strategies and conditional relationships in decision support. Logic systems are addressed with an overview of the open world assumption versus the closed world assumption and the relationship with the Semantic Web. The event-driven rule-based approach, which forms the basis upon which intelligent context processing can be realized, is presented with an evaluation and proof-of-concept. The issues and challenges identified in the research are considered with potential solutions and research directions; alternative approaches to context processing are discussed. The article closes with conclusions and open research questions.

1 Introduction

The concept of personalization in its many forms has gained traction driven by the demands of computer-mediated interactions generally implemented in *large-scale distributed systems* (LSDS), which may include *ad hoc* wireless networks. Personalization requires the identification and selection of entities based on a defined profile (a *context*). While research into personalization has generally addressed individuals, investigations are increasingly considering the application of context to a diverse range of entities; an entity has been defined as: 'a person, place, or physical or computational object' (Dey & Abowd, 1999). Recent developments have shown that the underlying motivation for the use of context is *personalized service provision* (PSP) to a range of entities in an increasingly broad and diverse range of domains and systems.

Historically, investigations into context have predominantly focused on two domains: (1) office applications and (2) tourist guides; the primary focus being on the use of *location*, *identity*, and *proximate* data (Moore *et al.*, 2010b). Investigations are, however, seeking to expand the domains addressed along with the range of data (contextual information) utilized. These advances are exemplified in the growing body of research that has investigated the application of context in domains including pedagogic systems, health monitoring, memory aids, field research tools, motion capture, group interaction support, pervasive computing, adaptation, *ad hoc* wireless networks, and group-based decision-support systems (Moore *et al.*, 2010b; Moore & Pham, 2012).

Documented research has addressed context as it applies to information systems (IS). Space restricts a detailed discussion on IS, however, a review of related research with a synopsis of the contextual information used and the approaches to implementation can be found in Moore *et al.* (2010b). There is no commonly agreed definition of the term IS or what constitutes an IS (Alter, 2008); in fact any system which operates on captured data which can be codified, digitized, and is machine processable to produce useful information can be viewed as an IS.

The traditional view of information resulting from data processing has been extended by Checkland and Holwell (1997) to include an intermediate stage termed *Capta* which is defined as 'data selected, created, or to which attention is paid' and 'involves the selection of data for processing into information useful to users'. This is analogous to the concept of relevance (Hildreth, 1998; Coppola & Della, 2004). The observations of Checkland and Holwell (1997), Hildreth (1998), and Coppola and Della (2004) clearly identify the synergy between relevance and PSP. The objectives for PSP are: the provision of relevant and appropriate services to users concomitant with the mitigation of the potential to deliver irrelevant services. PSP implemented in intelligent context-aware decision-support systems has the potential to increase the relevance of PSP, mitigate the issues surrounding information overload (Moore *et al.*, 2010a), and accommodate constraint satisfaction (CS) and preference compliance (Moore *et al.*, 2010d).

The approach proposed in this article to enable the realization of PSP is an event-driven fuzzy rule-based decision-support system with *ontology-based context modeling* (OBCM) (Moore *et al.*, 2007) implemented using Semantic Web technologies (W3C, 2012) with the Jena API (Jena, 2012); a discussion on rule-based systems (RBS) and the Semantic Web can be found in Moore *et al.* (2011). Research into RBS has investigated rules and rule strategies in diverse domains including (1) database systems, (2) data mining and warehousing, (3) fuzzy logic and fuzzy rule-based systems (FRBS), (4) decision trees, (5) artificial neural systems, and (6) logical reasoning and inference (Moore *et al.*, 2010a, 2010b). A discussion on RBS and rule strategies can be found in Hayes-Roth (1985), Gonzales and Dankel (1993), Mitchell (1997), Moore *et al.* (2011); a comprehensive exposition on FRBS is presented in Berkan and Trubatch (1997). In summary, the literature demonstrates the efficacy of RBS and FRBS in a broad range of applications in diverse domains, systems, and technologies (Moore *et al.*, 2011).

An analysis of the domains addressed and the methodologies used to capture and process contextual data clearly demonstrates (1) the reliance on location, identity, and proximate data in the case of the early context-aware systems and (2) the growing innovation in the usage of the available data in the provision of PSP. Important conclusions drawn are the diverse nature of context, its inherent complexity, and the difficulty in defining dynamically a user's context (Moore, 2009; Moore & Pham, 2011, 2012). Addressing these limitations demands PSP and the approach proposed in this article, in applying intelligent context processing (CP), has been shown to provide an effective basis upon which computational intelligence can be implemented and the inherent complexity of context leveraged.

The remainder of this article is structured as follows: personalization in LSDS is introduced with consideration of the relationship between such systems, context, and CP. This is followed by a discussion around rule strategies and conditional relationships in decision support. Logic systems are addressed with an overview of the *open world assumption* (OWA) versus the *closed world assumption* (CWA) and their relationship with the Semantic Web. The event-driven rule-based approach (which forms the basis upon which intelligent CP can be realized) is presented with an evaluation and proof-of-concept. There is a discussion which considers the issues and challenges identified in the research with potential solutions and research directions; alternative approaches to CP are discussed. The article closes with conclusions and open research questions.

2 The computing landscape

The developments in the computing are reflected in the PSP in LSDS which are characterized by the ubiquity of mobile and wireless systems.

There is a large body of documented research addressing personalization; the focus is (generally) on accommodating user preferences to enable service mediation and content adaptation based on context. Recently, however, there has been interest in expanding the use of context in systems where PSP with decision support forms a central function. Viewed from this perspective, the concept of personalization has been expanded to address an increasingly diverse range of potential contextual information in IS; this is exemplified by the growing interest in the use of *kansei* words in applications using *kansei* engineering (Salem *et al.*, 2009).

The traditional paradigm of centralized *Internet* networks has been largely replaced by a new distributed *Internet*-based communications paradigm characterized by networks created using centralized systems and distributed *ad hoc* wireless networks that are increasingly accessing geospatial, temporal, and cloud-based systems (Kermarrec *et al.*, 2003; Nurmi *et al.*, 2009). These developments have resulted in systems in which a user's current *state* (termed *context*) (which describes and defines a user's prevailing personal, environmental, proximate, and social situation) may be static or, in mobile systems, highly dynamic. Mobile systems incorporate a diverse range of geographically diverse infrastructures with a potentially large and highly dynamic user base and a broad range of fixed and wearable mobile devices as discussed in Moore *et al.* (2010b).

The *Internet* systems paradigm is frequently characterized by LSDS (Kermarrec *et al.*, 2003; Nurmi *et al.*, 2009; Lan *et al.*, 2010) in which interactions between infrastructure components and individual users (more accurately the users devices) are inherently complex, the complexity increasing exponentially as the nodes in the system increases. This complexity places great strain on communication systems with issues in the management of the interactions and the ability to enable PSP that requires both user and network infrastructure contextual information to effectively target service provision. LSDS are inherently context aware, context performing an increasingly important role. The rule-based approach presented in this article is posited as an effective approach to enable (1) targeted service provision, (2) the processing of data from a diverse range of geographically and technologically diverse sources, (3) the capability to handle the inherent complexity of context, (4) the ability to manage CS, and (5) the ability to realize predictable decision support under uncertainty.

3 Context and context-aware systems

The issues of personalization, intelligent context and its inherent complexity, CP, and context matching (CM) with partial matching (PM), and CS have been explored in Moore (2009), Moore *et al.* (2010a, 2010b, 2010c, 2010d), and Moore and Pham (2011, 2012). This section considers the system design requirements as they relate to intelligent context-aware decision-support systems, context, and the nature of potential contextual information.

3.1 System design requirements

The system design requirements which must be accommodated can be summarized as follows:

1. The implementation of predictable decision support. Arriving at a decision requires the matching of the context properties that describe an *input* with the corresponding context properties describing an entity (the *output*). This matching process is CM in which the probability of achieving a perfect match is low (Moore *et al.*, 2010a, 2011; Moore & Pham, 2011, 2012); therefore, CM must be capable of reaching a decision where PM is the general use-case. Figure 1 graphically models the CM and PM problem.
2. For systems where PSP is the aim, the goal for CP and CM is to arrive at a Boolean decision as to the suitability of an entity (generally an individual user or his/her mobile device) for PSP.

The *input* (context properties) and therefore the corresponding *output* (context properties) (more accurately stated as the *literal values* that apply to a property as discussed in subsequent sections) are

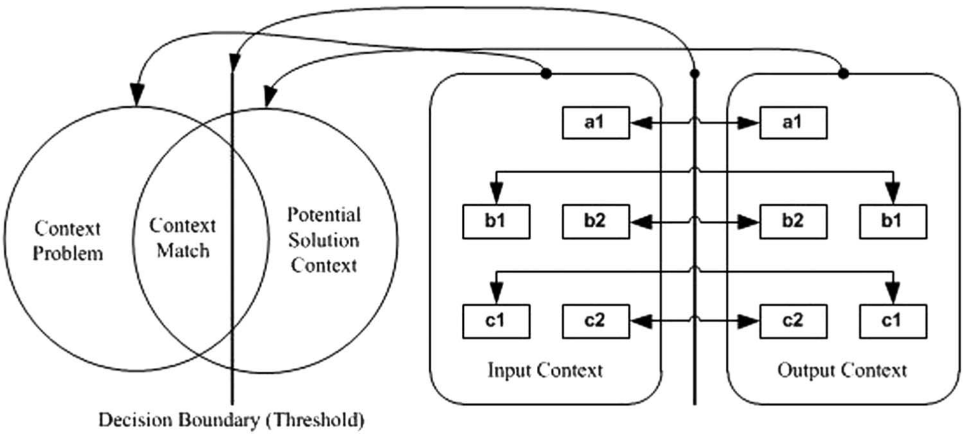

Figure 1 Context matching (CM): shown are the *input* and the corresponding *output* contexts with the CM and the decision boundary (the approach to defuzzification as discussed in Sections 4.2, 5, and 7)

potentially highly dynamic and can be extremely variable being dependent on the input which is generally unique (every use-case is new and individual). This is emphasized in the approaches considered in this article which include a broad range of applications including recent research addressing health-monitoring systems and a novel recommender system which targets the identification of stock-market trading strategies based on context-aware group-based decision support (Pham *et al.*, 2011).

The implementation of CM with PM, CS, and decision support are not independent processes but are closely interconnected. To address these requirements the novel context-processing algorithm (CPA) has been developed as articulated in this article.

3.2 Context and contextual information

Context awareness describes a concept in which the profile of an entity is defined by its context, an entity being defined in Dey and Abowd (1999) as 'a person, place, or a physical or computational object'. Context awareness employs context to identify individuals and implement PSP. Context is highly domain and application specific requiring the identification of domain-specific function(s) and properties (Moore *et al.*, 2010a, 2010b, 2010c) and therefore must reflect a user's current dynamic state (Moore *et al.*, 2010a). Location is central to context; context, however, includes more than just location (Moore *et al.*, 2010d). As identified in Sheth and Ramakrishnan (2003) and Moore *et al.* (2010a, 2010b), a broad and diverse range of context factors combine to form a context definition, in fact, almost any information available at the time of an interaction can be viewed as contextual information including:

1. the variable tasks demanded by users;
2. the diverse range of mobile devices and the associated service infrastructure(s) in LSDS incorporating *Wide Area Networks*, *Local Area Networks*, and *Personal Area Networks* (PAN) implemented in *Internet* and *Internet* systems;
3. resource availability (connectivity, battery condition, display, network, and bandwidth, etc.);
4. nearby resources (accessible devices and hosts including I/O devices;
5. the physical situation (temperature, air quality, light, and noise level, etc.);
6. the social situation (who you are with, people nearby, etc.—proximate information);
7. spatial information (location, orientation, speed and acceleration. etc.);
8. temporal information (time of the day, date, and season of the year);
9. physiological measurements (blood pressure, heart function—electrocardiography (ECG or EKG from the German *Elektrokardiogramm*), cognitive functions related to brain activity (EEG from *Electroencephalography*), respiration, galvanic skin response, and motor functions including muscle activity);
10. cognitive and abstract contextual information such as an individual's emotional responses, intuition, feelings, and sensibilities expressed in terms of semantic terminologies.

Context-aware systems have been developed in recent years to improve user interaction in applications and services which operate in highly dynamic environments which are characterized by adaptable contextual situations (Hong *et al.*, 2009). Contextual information (captured context data processed into information useful in CP) is fundamental in the development of context-aware applications where improved interactions between computers, devices, and people in LSDS is a system requirement. In the modeling of context the requirements for dynamic environments include scalability, technology, mobility, tolerance, robust systems, security, CS, and decision support. When dealing with context, context-aware systems and applications may run multiple computing devices and sensors that define and describe the evolving *states* that characterize the dynamic nature of context. CP as proposed in this article has the capability to enhance a context-aware systems performance when sharing data between services and clients in LSDS.

Context modeling is concerned with the modeling of data (on a conceptual level) and includes the metadata and the contextual information (the context properties and their *literal values*) that describes and defines a context where data are captured from a range of sources. A good context-modeling formalism should (1) be human and machine readable (Moore *et al.*, 2010b, 2010c) and (2) reduce the complexity (of a context-aware system) with the aim of improving performance.

Contextual data are (generally) gathered by a set of autonomous sensors. When gathering data the process of context data generation consists of filtering, aggregation, fusion (CM as discussed in this article extends the concept of data fusion), and may additionally include inference and reasoning. Creating a context requires contextual information which describes the resources available by the computing infrastructure and an entities (generally an individual's or an individual's mobile device) context. In intelligent context-aware systems context middleware is used to detect and adapt to the changing context(s) and implement PSP. Thus, context-aware systems should facilitate the adaptation of the behavior among services, accommodate dynamic environments, and enable CS. Context modeling should also provide an effective basis upon which inference and reasoning (generally using subsumption and entailment) can be implemented in complex situations.

In semantic intelligent context-aware systems, contexts are dynamically influenced by user intuitions and preferences. An appropriate method called *kansei* engineering has been developed to deal with human feelings, demands, and impressions of context-aware applications. *Kansei* is a Japanese term meaning *sensibility*, *impression*, and *emotion* (Salem *et al.*, 2009). *Kansei* words are given by adjectives describing human *emotion* (more accurately emotional response), *sensibility*, and *impression*; there is no equivalent term in English, the nearest applicable word is possibly *intuition*. *Kansei* evaluation is commonly used for evaluation methods to quantify impressions.

For *kansei* evaluation, we have determined adjective pairs called *kansei* words in pairs: (*Synonym and Antonym*) and (*Synonym and Not Synonym*). For instance, the pairs of adjectives (*good and bad*) and (*successful and unsuccessful*) are *kansei* words. This approach is discussed in Pham *et al.* (2011) where *kansei* words are used in quantifying trader sensibilities about trading decisions, market conditions with uncertain risks in a *context-aware-based group decision making*. By aggregating user preferences and selecting alternatives, a group of individuals enhance potentially optimal solutions in contextual information. The use of *kansei* evaluation shows that if data can be captured, measured, codified, and digitized it can be considered to be contextual information.

The range of contextual information identified along with the dynamic nature of the data demonstrates the inherent complexity of context, its domain-specific nature, and the difficulty in defining and measuring it. This difficulty is exemplified in the need to accommodate two general types of context: a *static* context and a *dynamic* context (Moore & Hu, 2007; Moore *et al.*, 2010b, 2010c). A static context relates to a use-case in which a user has an element of control in context creation; a dynamic context relates to a condition in which the user is seen as being passive, or at least somewhat less in control; in such a use-case the system monitors, analyses, and reacts dynamically to a user's behavior and 'state'. The two types of context are reflected in the two principal ways context is used, these are: (1) as a retrieval indicator (a static context) and (2) to tailor system behavior to match users system usage patterns (a dynamic context). The static and dynamic context also uses to provide relevant information, matching with users in uncertain environments.

4 Rules and rule strategies

Rules are generally conditional specifications that instruct, permit, trigger, and inhibit processes, functions, and actions. Many approaches have emerged to create RBS's; the approaches, however, share certain key properties (Hayes-Roth, 1985) including (1) the capability to incorporate practical human knowledge in conditional IF-THEN rule strategies, (2) the degree of accuracy improves at a rate proportional to the growth in the size of the knowledge base, (3) the capacity to solve diverse and complex problems, and (4) the ability to determine the best execution sequence.

Rules fall into two general types: (1) reasoning and inference rules which are generally used where inference is drawn to identify conditional relationships such as familial or peer-to-peer and business-to-business relationships and (2) trigger rules which generally implement actions either in the form of a Boolean decision or the firing of another rule. Rule strategies fall into four general classifications (Moore *et al.*, 2011): (1) derivation or deduction rules, (2) integrity constraints, (3) transformation rules, and (4) reaction or event condition action (ECA) rules:

1. *Derivation rules*: generally relate to conditional relationships. For example, IF one statement or set of statements is true THEN a second statement or set of statements is also true.
2. *Integrity constraints*: in general use this rule type relates to validation checking predicated on truth by negation (if it is not true then conclude it is an error).
3. *Transformation rules*: relate to truth in, for example, one knowledge base or ontology with truth in another. For example, in a modified form (when used in combination with integrity constraints) transformation rules have been used in the matching and merging of ontologies (Lera *et al.*, 2010).
4. *Reaction rules*: relate to the notion of ⟨*action*⟩ where the IF component evaluates the rule ⟨*condition*⟩ resulting in an ⟨*action*⟩. The outcome can be either (1) a Boolean decision or (2) the firing of another rule.

The four rule strategies are based on the IF-THEN logic structure in the form: IF ⟨*condition*⟩ THEN ⟨*action*⟩; the context-processing rules (CPR) and fuzzy event condition rules algorithm (FECA) rules strategy is based on the ECA rule strategy using the IF-THEN logic structure. In actuality, RBS generally utilize a number of rule strategies. Viewed from an intelligent context-processing perspective, the final output is a Boolean decision; this clearly calls for a trigger rule type with a reaction rule strategy. However, as discussed in Moore *et al.* (2010d) CS requires validation using integrity constraints to address, for example, implementation of security and access control policies. Where context modeling utilizes OBCM (Moore *et al.*, 2007), transformation or derivation rules strategies may be used.

RBS using the ECA rule strategy with the IF-THEN logic structure have been employed in diverse domains (Moore *et al.*, 2010b) including FRBS as discussed in Berkan and Trubatch (1997). There are five principal fuzzy rule composition strategies identified in Berkan and Trubatch (1997): (1) competitive rules, (2) weighted rules, (3) prioritized rules, (4) hierarchical rules, and (5) adaptive rules. Given the inherent complexity of context (Moore, 2009) and the demands of CM with CS (Moore *et al.*, 2010d) no single fuzzy rule composition strategy fulfills the requirements of CM (see Section 5 below). The FECA rules strategy (Moore *et al.*, 2011) employs strategic functions drawn from a number of fuzzy rule composition strategies; for example, thresholds (competitive rules) and weighting (weighted rules). It is therefore clear that to address the domain-specific demands of RBS no one rule strategy is capable of addressing all RBS, each problem requiring a domain and problem-specific rules strategy based on the use of conventional logic with fuzzy logic using fuzzy extensions to reflect the need to address cases where conventional logic is not an adequate solution.

4.1 The Semantic Web and rule-based systems

There is an ongoing debate in Semantic Web circles surrounding the use of rules (Horrocks *et al.*, 2003, 2004; Sheth & Ramakrishnan, 2003; Parsia *et al.*, 2004) and in addressing the Semantic Web Rules Language (W3C, 2012) it has been observed that 'it is commonly acknowledged that rules are next' (Parsia *et al.*, 2004). An interesting aspect of this debate is the approach to reasoning and inference, Ontology Web Language (OWL), and the OWA versus the CWA. The CWA is predicated on *negation-as-failure* (i.e. ‘if it is not provably true, conclude that it’s false’) (Sheth & Ramakrishnan, 2003). The Semantic Web is predicated on the OWA; however, the CWA is also useful in certain applications (Horrocks *et al.*, 2003, 2004; Sheth & Ramakrishnan, 2003). A discussion on the topic is beyond the scope of this article; however, an exposition with extensive references can be found in Sheth and Ramakrishnan (2003) and Sheth *et al.* (2005). In summary, the proposed approach implemented using the CPA is predicated on the CWA to enable effective CM and CS with predictable decision support.

4.2 Logic systems and context

Conventional logic is generally characterized by notions based on a clear numerical bound (the crisp case) (Klir & Yuan, 1995) in which an element is defined as a member of a set based on numerical parameters in the discrete range {1,0}. Traditionally, logic (also referred to as Boolean logic) consists of three elements (Berkan & Trubatch, 1997): (1) truth values, (2) linguistic connectors, and (3) reasoning types. In Boolean logic truth values are in the discrete range {1,0} (*truth* or *false*, respectively).

In fuzzy logic truth is a matter of degree with truth values in the continuous range [1,0] (Klir & Yuan, 1995; Berkan & Trubatch, 1997). Fuzzy set theory enables a variable measure of membership of a set (expressed in terms of degrees of membership) (Klir & Yuan, 1995; Berkan & Trubatch, 1997) defined using a fuzzy distribution function (often termed a *membership function* in the literature) (Berkan & Trubatch, 1997) based on normalized values in the continuous range [1,0]. These mapping assumptions along with the related approach to defuzzification as discussed in Berkan and Trubatch (1997), Moore and Pham (2012), and subsequent sections of this article are central to the FECA rules strategy.

To illustrate the need for variable degrees of membership of a set in the posited approach consider CM where the likelihood of a perfect match between the input and corresponding output context property values has a low probability (Moore *et al.*, 2010a, 2010b, 2011; Moore & Pham, 2012). This introduces the need for PM (Moore *et al.*, 2010a, 2011) where a matched context mapped to a normalized value of, for example {0.80}, has a defined degree of membership. CM with PM imposes issues similar to those encountered in decision support under uncertainty, which is possibly the most important category of decision problem (Shackle, 1961) and represents a fundamental issue for decision support.

Uncertainty is associated with vagueness (Klir & Yuan, 1995) and is considered in relation to FRBS by Berkan and Trubatch (1997) where fuzzy logic, as it applies to decision support, is discussed. Space limits a discussion on the topic, however, a detailed exploration of fuzzy sets and fuzzy logic can be found in Klir and Yuan (1995); a comprehensive discussion on fuzzy system design principles can be found in Berkan and Trubatch (1997), where a number of classes of decision problem are identified and discussed. In summary, the CPA which is predicated on an FRBS, a strategy has been shown to provide the ability to arrive at decisions under uncertainty with high levels of predictability (Moore *et al.*, 2011; Moore & Pham, 2012).

4.3 Rule-based systems design

The use of an RBS raises questions in two (general) areas: (1) RBS design when viewed from a parametric design perspective with respect to context-aware systems and (2) the use of a rules engine—arguably a similar question with a different emphasis; however, the questions raised extend the RBS design parameters.

The use of a rule-based approach clearly involves the design and development of a suitable rule set serialized in a machine (and ideally human) readable formalism. In addition, an RBS is domain specific requiring domain-specific design where the primary considerations are the domain of interest and the problem being addressed; the domain-specific nature of RBS is shared with context-aware systems and can be summarized as form-follows-function. From a design perspective a significant potential issue for rule-based solutions is the temptation to apply an inappropriate solution to a problem. To address this issue the question Is a rule-based solution a suitable option? must be answered.

To attempt to provide an answer to this question the following design parameters have been identified to test the suitability of an RBS approach: (1) Is human problem-solving knowledge being replicated in the solution? (2) Is the problem-solving knowledge heuristic in nature? (3) Does the knowledge (or expertise) periodically change? (4) If expertise is involved, is the expertise fairly well understood and accepted? (5) Is the problem well understood? (6) Are the input data always complete and correct? (7) Can the problem be (better) solved by conventional database or programming solutions? (8) Does it pass the telephone test? (this test asks whether an expert, by speaking with someone over the telephone, can gather sufficient information to solve the problem). (9) Is the decision-making process merely the matching of data? (or are rules applied in the matching and decision-making process). (10) Is speed of operation a prime requisite for the system? (an RBS is arguably slower at runtime than conventional programming solutions). (11) Are partial solutions acceptable in the identification of individuals?

The design parameters and issues 1–8 have been derived from Gonzales and Dankel (1993), parameters 9–11 representing three additional parameters which are relevant to intelligent context-aware systems (Moore *et al.*, 2010a, 2011).

4.4 Rule-based systems and context processing

The use of a rule-based approach clearly involves the creation of rules and a suitable data structure. In the posited approach the rules are encoded in Java; the rules updating and accessing a Jena TDB data set using SPARQL and SPARQL Update (SPARQL, 2007; Jena, 2012; SPARQL Update, 2012; W3C, 2012). The data structure is a semantic context-modeling ontology created using OWL DL (W3C, 2012) based on OBCM as discussed in Moore *et al.* (2007). Space restricts consideration of the semantic context-modeling ontology, however, the ontology is presented with a discussion, illustrative implementations, and proof-of-concept in Moore *et al.* (2010b). In summary, to define the context properties and the related data values the RDF/S triple structure $\langle S\rangle$ $\langle P\rangle$ $\langle O\rangle$ (the *subject*, *predicate*, and *object* (the *literal value*), respectively) is used. RDF/S and OWL serialization can be cryptic and is not particularly human friendly, however, the availability of development tools such as Protege 3.4 and 4.1 (Protege, 2012) which automatically creates the RDF/XML serialization generally resolves this issue.

The design parameters 1–8 have been derived from Gonzales and Dankel (1993); parameters 9–11 representing three further design considerations relevant to the CM problem under investigation. A comprehensive discussion on the design and engineering of knowledge-based systems can be found in Gonzales and Dankel (1993), however, in summary, a positive response to parameters (1, 2, 3, 4, 5, 8, 11) indicates that an RBS does potentially provide an effective solution. A positive response to parameters (6, 7, 9, 10), however, implies that a rule-based approach is not potentially a viable solution. Applying the design parameters to the CM problem points to the conclusion that an RBS does provide a viable and effective basis for realizing a context-aware decision support in an intelligent context-aware system.

5 Context processing

Context processing utilizes the *context-processing algorithm* that implements two algorithms: (1) the *context-processing rules* and (2) the *fuzzy event condition rules algorithm*, which is an extension of the CPR algorithm. The CPR and FECA algorithms (Moore *et al.*, 2010c, 2011; Moore & Pham, 2012) are designed to implement CM with PM and CS. Figure 1 graphically models the CM problem; shown is the input and output context(s) with the PM problem and the implementation of the membership function (the decision threshold) the FECA algorithm is designed to facilitate (Moore *et al.*, 2011); see Section 8 (below) where there is an overview of the design for the membership function. In certain cases, for example, context-aware health-monitoring systems as discussed in Moore *et al.* (2010d) and Ashford *et al.* (2010), the proposed approach has the capability to implement multiple decision thresholds (Moore & Pham, 2012) where there are number of possible decisions relating to a patient's medical condition and prognosis.

5.1 The context-processing algorithm

The CPR implemented in the CPR and FECA algorithms are predicated on three elements: (1) event, (2) condition, and (3) action. The basic structure of the rules is $\{ON\ \langle event\rangle\ IF\ \langle condition\rangle\ THEN\ \langle action\rangle\}$. An *event* relates to an input which can be either (1) the $\langle event\rangle$ and the context data or (2) the output resulting from the firing of a rule (which triggers a subsequent rule). A *condition* (a rule antecedent/body) relates to the evaluation of the context properties in which comparisons are drawn between the *input* property(s) and the *output* context properties (a potential solution properties and related property values derived from the data structure) as discussed in the context-matching algorithm. An *action* (a rule consequent/head) is the result derived from the processing of the rule condition(s) (the rule antecedent/body) and results in either (1) the triggering of a related simple rule, a complex or composite rule, or another rule set or (2) a Boolean decision based on CM arrived at using CP. In (1) there is a pre-defined sequence incorporating built in stopping criteria setting out termination and/or CS criteria. In (2) the decision is based on the relationship between the context match and a decision threshold.

5.1.1 The context-processing rules algorithm

Using the illustrative context properties $a1$, $b1$, $b2$, $c1$, $c2$ modeled in Figure 1 the CPR algorithm is as follows where:

- e is the numerical evaluation of an individual property match {1,0} (*true*/*false*, respectively);
- n is the sum of the individual property evaluation matches;
- x is the value for a perfect match; that is, if all context matches are *true* and the computation equates to a normalized value of [1];
- m is the normalized value for the context match: ($m = n/x$);
- t is the preset threshold value in the range [0…1].

Applying these functions the algorithm process is as follows:

- *Step 1*: evaluate the context match (e) for each individual context property [1,0] (*true*, *false*).
- *Step 2*: sum the values for (n) derived from the matching process for each individual property (e).
- *Step 3*: calculate the value (x) for a perfect match.
- *Step 4*:calculate the normalized value for (m) in the range [0…1].
- *Step 5*: using the preset threshold (t) value (in the range [0…1]) determine if the output (potential solution) context is a suitably qualified match with the input (problem) context.

5.1.2 Evaluation and proof-of-concept

An example implementation of the CPR algorithm where: $e(a1) = [1]$, $e(b1) = [0]$, $e(b2) = [1]$, $e(c3) = [1]$, $e(c4) = [0]$ and given that: $x = [5]$ results in a normalized value for m of {0.60}. As (m) is less than the preset threshold value ($t = 0.75$) the context match is *false* and the individual is therefore *not* a suitably qualified individual.

5.2 The fuzzy event condition action rules algorithm

The FECA rules algorithm extends the RDF/S triple structure (used in the CPR algorithm) with a weighting $\langle W \rangle$ component to give a triple plus comprising $\langle S \rangle$ $\langle P \rangle$ $\langle O \rangle$ $\langle W \rangle$. The FECA rules algorithm is predicated on a quantitative approach and is based on fuzzy logic using degrees of membership of a set as discussed in Moore *et al.* (2011) and Moore and Pham (2012) and Section 4.2 (above). In summary, the $\langle W \rangle$ component in the numerical range [0.01…1.00] is applied to each context property to implement a prioritizing bias into the CP to accommodate CS (or at least mitigate CS violations).

5.2.1 Implementation

The FECA rules algorithm is set out below with an example implementation to demonstrate CP and provide proof-of-concept. Using the illustrative context properties $\{a1, b1, b2, c1, c2\}$ modeled in Figure 1 the FECA rules algorithm is as follows:

- w [0.10…1.00] is a weight applied to each context property to reflect its relative priority.
- e {1,0} is the numerical (Boolean) evaluation for each context property match true = [1] or false [0].
- av ($e \times w$) is the computation of the actual value for each context property match evaluation in the range [0.00…1.00].

- *sav* (*av* (*a*1…*an*)) is the sum of the *av*.
- *mpv* (*av* (*a*1…*an*)) is the maximum potential value. This represents a state in which all context property matches (*av* (*a*1…*an*)) are true [1]. The *mpv* assumes that a perfect context match has been identified.
- *rv* (*sav*/*mpv*) is the resultant value. This value represents the degree to which the overall context match is *true* in the range [0.00…1.00]. The (*rv*) value is used in step 7 to test if (*rv*) is greater than (*t*).
- *t* is the value for the decision boundary (threshold) in the range [0.10…1.00]. The (*t*) is used in step 7 to test if (*rv*) is greater than (*t*). Note: the posited approach enables multiple thresholds (*t*).

Applying these functions the algorithm process is as follows:

- *Step 1*: evaluate the context match [1,0] for each individual context property, for example: (IF (*a*1 (input)).equalTo(*a*1(output)) THEN $e = [1]$ *or* IF (*a*1(input)).notEqualTo(*a*1(output)) THEN $e = 0$).
- *Step 2*: obtain the pre-defined property weighting (*w*) in the range [0.1…1.0] for each context property.
- *Step 3*: apply the weighting (*w*) to the value as derived from (2): (IF $e(a1) = \{1,0\}$ THEN $av = (e \times w)$).
- *Step 4*: sum the values derived from the matching process: ($sav = (av\ (a1) + av\ (b1) + av\ (b2) + av\ (c1) + av\ (c2))$).
- *Step 5*: compute the *mpv* for the CM process: ($mpv = (w(a1) + w(b1) + w(b2) + w(c1) + w(c2))$).
- *Step 6*: compute the *rv* for testing against threshold value (*t*): ($rv = (sav/mpv)$).
- *Step 7*: using the preset threshold value (*t*) determine if the output (potential solution) context definition is a suitably qualified match with the input context: IF (*rv*) is greater than or equal to (*t*) THEN context match = *true* [1] or IF (*rv*) is less than (*t*) THEN context match = *false* [0].

5.2.2 *Evaluation and proof-of-concept*

The following illustrative implementation of the FECA rules algorithm demonstrates the CM process. Applying the algorithm using the context properties modeled in Figure 1 and the following example values for (*w*), (*e*), and (*t*) results in a computation as follows:

- the values for (*w*) are ($w(a1) = 0.7$, $w(b1) = 0.8$, $w(b2) = 0.7$, $w(c1) = 1.0$, $w(c2) = 0.9$);
- the values for (*e*) are ($e(a1) = [1]$, $e(b1) = [0]$, $e(b2) = [1]$, $e(c1) = [1]$, $e(c2) = [1]$);
- the threshold value (*t*) is {0.75}.

Using these values the CM process is as follows: $sav = ((av(a1) = ([1] \times 0.7) = [0.70]) + (av(b1) = ([0] \times 0.8) = [0.00]) + (av(b2) = ([1] \times 0.7) = [0.70]) + (av(c1) = ([1] \times 1.0) = [1.00]) + (av(c2) = ([1] \times 0.9) = \{0.9\}))$. Thus, $sav = 3.30$.

$mpv = (av\ (a1 \ldots an)) = 4.10$.
$rv = (sav/mpv)$, $rv = (3.30/4.10)$, $rv = 0.80$.

As (*rv*) is 0.80 and $t = 0.75$ (*rv*) is greater that (*t*). The result of the CM is therefore *true* [1] and the individual is a suitably qualified individual.

An analysis of the implementation supports the following conclusions:

- The evaluation of the context properties (step 1) and their respective values will initially result in a Boolean result in the range $(e) = \{1,0\}$.
- The Boolean value for *truth* [1] or *falsity* [0] derived for (*e*) (step 1) is retained when the weight $\langle W \rangle$ applicable to a context property (*w*) is applied ($e \times w$) (step 3).
- The implementation of a prioritizing bias to reflect the relative importance of specific context properties is achieved.
- CM with PM and CS is realized using the FECA rules strategy.

In actuality, implementation of the proposed approach employs a four-stage process. The use of the $\langle W \rangle$ parameter is utilized for context properties in stages 2 and 3 of the CM process:

- *Stage 1*: the evaluation of control properties to validate the inputs, apply error checking, apply relevant constraints and policies, and implement rule assignment as discussed in Berndtsson and Lings (1995) and Moore *et al.* (2010b).

- *Stage 2*: an evaluation of the *input* context properties with the corresponding *output* context properties (CM) to identify if an *output* context (an instance of an entity where such an instance is a member of a set of entities) is a suitably qualified recipient for personalization.
- *Stage 3*: the application of the context properties that represent constraints and the implementation of priorities to avoid (or at the very least mitigate) violations of CS.
- *Stage 4*: the implementation of the decision; there are two courses of action, either (a) the delivery of the service if the CM is *true* (equates to [1]) or (b) in the event of the CM being *false* (equates to [0]) the system terminates.

5.3 The context-processing rules and fuzzy event condition rules algorithm strategy

The CPR and FECA rules strategies are predicated on the ECA rule strategy (Moore *et al.*, 2010a, 2010b, 2011; Moore & Pham, 2012) in the form: $\{ON\ \langle event\rangle\ IF\ \langle condition\rangle\ THEN\ \langle action\rangle\}$. The $\langle event\rangle$ rule component implements either (1) a system input event such as a location change or login event or (2) an event resulting from the firing of a previous rule. The $\langle condition\rangle$ (or antecedent) component of the rule employs a Boolean structure similar to Horn style rules in a simple rule structure using a single context property with input and output literal values. In operation, the condition rule component executes a conditional comparison of the input and output literal values to arrive at a Boolean result: the *truth* [1] or alternatively the *falsity* [0].

The $\langle action\rangle$ (or consequent) rule component reacts to the result of the $\langle condition\rangle$ rule component, the output being either a Boolean decision or the firing of a subsequent rule or rule set. The condition component of an ECA rule (the antecedent) employs the IF-THEN rule strategy commonly found in reaction rules (Berndtsson & Lings, 1995) which relates to the notion of $\langle action\rangle$ where the IF component evaluates the rule $\langle condition\rangle$ resulting in an $\langle action\rangle$. The $\langle action\rangle$ in the proposed approach can be either (1) a Boolean decision or (2) the firing of another rule.

To address the PM issue the posited approach applies the principles identified in fuzzy logic and fuzzy set theory with a defined membership function (Moore & Pham, 2012) as discussed in Section 4.2 (above). Conventional logic is generally characterized using notions based on a clear numerical bound (the crisp case); that is, an element is (or alternatively is not) defined as a member of a set in binary terms according to a bivalent condition expressed as numerical parameters {1,0}. Fuzzy set theory enables a continuous measure of membership of a set based on normalized values in the range [0,1].

These mapping assumptions are central to the CM algorithm (Moore *et al.*, 2010a, 2010b, 2011; Moore & Pham, 2012). A system becomes a fuzzy system when its operations are ‘entirely or partly governed by fuzzy logic or are based on fuzzy sets’ and ‘once fuzziness is characterized at reasonable level, fuzzy systems can perform well within an expected precision range’ (Berkan & Trubatch, 1997). Consider a use-case where a matched context mapped to a normalized value of, for example [0.80], has a defined degree of membership. This measure, while interesting, is not in itself useful when used in a decision-support system (Moore & Pham, 2012); in the CPA this is addressed using a distribution function, more generally referred to in the literature as a membership function (Berkan & Trubatch, 1997) to implement the essential process of defuzzification as discussed in Moore *et al.* (2011), Moore and Pham (2012), and Berkan and Trubatch (1997).

6 Context-processing and large-scale distributed systems

As discussed in Section 2 (above) the computing landscape is frequently characterized by LSDS; such systems are dynamic, complex, inherently context aware, and generally function under uncertainty. Addressing these often conflicting characteristics places high demands on context middleware and the processing of data derived from diverse local and distributed sources including ‘mashups’ (Lopez *et al.*, 2011) and cloud-based systems (Rimal *et al.*, 2009). The posited approach to CP has been developed as a generic system capable of processing contextual information from a diverse range of distributed data sources with implementation in a broad range of domains, systems, and technologies. As such, the CPA incorporates the ability to manage the inherent complexity that characterizes context, context-aware systems, and LSDS. In addition, the data structure employed in the posited approach and implemented using the Jena Java API (Jena, 2012) incorporates the ability to handle both *in memory* and *persistent* data storage along with updating using distributed data processing employing local processing in, for example, a mobile device using a thin client and centralized processing in a server.

Decentralized systems will increasingly feature local processing of captured data into information useful in CP. For example, (1) there may be a preprocessing stage in the overall processing of data captured from a sensor(s) in which processing capacity may be incorporated into a sensor(s) (Thomas *et al.*, 2012) before submission to the context middleware or (2) in an assisted living environment (in, e.g., a health-monitoring system; Ashford *et al.*, 2010; Moore & Pham, 2012) local processing of data in an intelligent context-aware decision-support system using a combination of sensors and mobile technologies in a PAN may be employed. Space precludes a detailed discussion on the topic; however, the relative merits and demerits, while complex, are well covered in the literature and range from technical considerations (Thomas *et al.*, 2012) to cryptography using digital aggregate signature schemes with bilinear mapping (Boneh *et al.*, 2003a, 2003b).

In summary, the design issues are twofold: (1) data storage and (2) context processing. For general location and identity-based systems a distributed approach using a thin client solution with local data storage has potential advantages. The increased complexity of intelligent context-aware systems, however, presents increased data storage and processing demands with related computational overhead and tractability issues. These issues, when taken with the multiple data capture modes in LSDS, wireless *ad hoc* networks, and mobile systems present significant challenges in the area of data security. While the centralized server approach may offer greater potential security the growing ubiquity of wireless mobile systems demands a domain-specific approach to implementation.

Saha and Mukherjee (2003) in addressing pervasive computing have noted that all the basic component technologies exist today in hardware, we have mobile devices, sensors, and even smart appliances. Thomas *et al.* (2012) concur observing components such as sensors, wireless mesh architectures, cloud services, and data brokerage/processing are all currently available and widely researched. The challenges lie in the development of context middleware; the posited approach incorporates the ability to manage these challenges in LSDS.

7 Discussion

This article has considered personalization, PSP, and context with consideration of the domains and systems to which context has been applied and the nature of the contextual data. An overview of the evolving computing landscape has been presented with its relationship to context. Rules and rule strategies have been discussed with logic systems and an overview of the OWA versus the CWA and the relationship with the Semantic Web. The design parameters for RBS have been considered; the discussion supports the conclusion that an RBS represents a suitable solution to enable effective context processing. The event-driven rule-based approach (which forms the basis upon which intelligent context processing can be realized) has been presented with an evaluation and proof-of-concept.

It is posited that the proposed approach has the capability to implement intelligent CP with predictable decision support. In addition, the reported results in Moore *et al.* (2010a, 2010b, 2011) and Moore and Pham (2012) when taken with the evaluation presented in this article supports the conclusion that the posited approach provides an effective and predictable basis upon which CS can be accommodated (or at least the potential violations of CS can be mitigated). A discussion on the CPA, logic systems, and rule strategies with the related conditional relationships for intelligent context-aware systems can be found in Moore *et al.* (2011) and Moore and Pham (2012). For an in-depth discussion on CM and PM with example implementations and a data set evaluation, see Moore (2009) and Moore *et al.* (2010b).

While the research has resolved many crucial issues, a number of outstanding issues and challenges have been identified (Moore *et al.*, 2010a, 2010b, 2010c, 2011; Moore & Pham, 2012) including (1) the design of the membership function, (2) data capture and data description, (3) persistent storage, (4) centralized versus distributed systems, (5) the approach to the setting of the $\langle W \rangle$ parameter, and (6) alternative approaches to context processing.

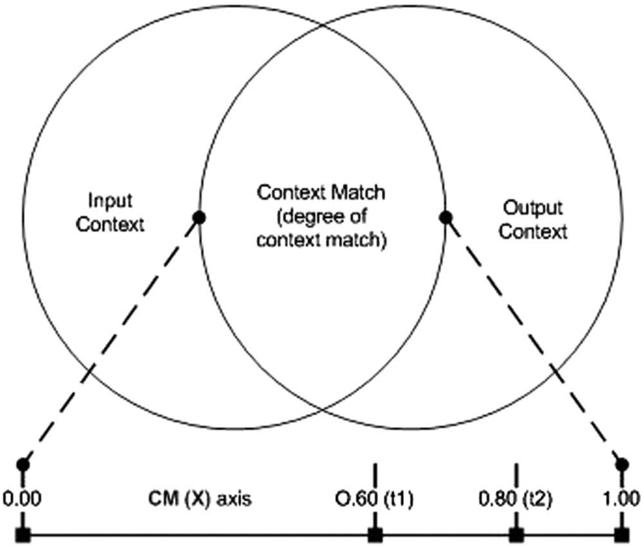

Figure 2 A conceptual model of the context matching (CM) mapped onto the solution space. Shown are the decision boundaries (thresholds) *t*1 and *t*2 used in the defuzzification process as discussed under the design of the membership function and in Sections 4.2 and 5

Item 4 has been discussed in Sections 2 and 6 (above); the conclusions drawn are that the posited approach provides an effective basis upon which the inherent complexity and data-intensive nature of LSDS can be accommodated. Items 2 and 3 have been addressed in Moore *et al.* (2010a, 2010b, 2011). Items 1, 5, and 6 are considered in the following sections.

7.1 The design of the membership function

The design of a membership function is domain and application specific (Berkan & Trubatch, 1997; Moore & Pham, 2012). The membership function currently uses a *decision boundary* (or threshold) solution to enable defuzzification in CM as discussed in Sections 4.2 and 5 (above). This approach, while effective, is recognized as a suboptimal solution. Current research is investigating (1) using *semantic* representation and (2) how heuristics are applied using the available *a priori* knowledge related to the domain of interest and the problem being addressed. For a comprehensive discussion on the design of the membership function with consideration of the relative impact of points (1) and (2) on the location of the decision boundary(s) (*t*1 *t*2), see Moore and Pham (2012). Figure 2 graphically models the CM and PM (the degree of membership as discussed in Section 5 (above)) mapped onto the solution space (shown as the *CM* (*X*) axis).

Points (1) and (2) represent important design parameters for a membership function. These parameters when taken with the relative contribution each context property (in a set of context properties that combine to create a context) makes to the overall context definition (Moore & Pham, 2012) are being used in ongoing research to attempt clarify the approach to the identification for the optimal values for the decision boundaries. The results derived from the research will be analyzed to attempt to identify the optimal solution for the membership function design generalizable to a range of domains and applications. The identification of the optimal solution represents an open research question.

7.2 Setting the weight value for the $\langle W \rangle$ *component*

Currently, setting the literal value for the $\langle W \rangle$ parameter value is set manually (to reflect the domain of interest and the problem being addressed). Ideally, the setting of the $\langle W \rangle$ value would be initially set manually using default values and updated dynamically as the context-aware system gathers information on users' system usage and changing context. However, a potential side effect of dynamic updating for the $\langle W \rangle$ parameter is the induction of unknown and emergent dynamic change to the preset design parameters and policy priorities; this may impact the ability to enable CS with predictability and consistency. Identifying solutions to the dynamic updating issue represents an open research question.

7.3 Alternative approaches to context processing

The current approach proposed in the article can be viewed in terms of a deterministic search for a suitably qualified user (u) who is a member of a set of users (U). This process is essentially a linear ‘brute-force’ search of the hypothesis space (U), to improve the efficiency of context processing a number of alternative approaches have been investigated including Bayesian methods to enable inferred conclusions to be drawn (i.e. the Boolean truth) as to a user(s) (u) suitability for personalization.

7.3.1 Bayes theorem and the context-matching problem

Bayesian reasoning provides a probabilistic approach to inference. However, a practical issue (when using Bayesian methods) is that they typically require knowledge of prior probabilities. When these probabilities are not known they are (generally) estimated based on empirical background information, previously available data, and assumptions about the underlying distributions. A detailed discussion on Bayes theorem and its fundamental principles with examples and applications can be found in Mitchell (1997). In summary, Bayes theorem applies *a priori* probabilities as set out below. In actuality, Bayes theorem is quite general and can be applied to any set of mutually exclusive hypothesis (h) in a set of hypotheses (H) whose probabilities sum to 1.

- P(h) denotes the initial probability that hypothesis (h) holds. P(h) is generally termed the prior probability of (h). In problems where no prior knowledge of probabilities exists the same probability may be assigned to each potential hypothesis (h).
- P(D) denotes the prior probability that the training data will be observed independent of knowledge relating to (H) where (H) is a set of hypothesis (h).
- P(D – h) denotes the probability that (D) will be observed given some world in which (h) holds.
- P(h – D) denotes that (h) holds given (D) and is generally termed the posterior probability of (h).

Intuitively, there is an observed similarity between Bayes theorem hypothesis where (h) is a member of a set of hypotheses (H) and the CM problem where a user (u) is a member of a set of users (U). An analysis of Bayesian methods when applied to the CM problem has, however, identified a number of issues. Before addressing these issues consider the following relationships: (1) an instance of an individual (u) in the CM problem generally correlates with an hypothesis (h) in the Bayes theorem and (2) the input context data (C) in the CM problem generally equates to the data (D) in the Bayes theorem. Given these relationships and that according to Bayes theorem the sum of the probabilities must sum to [1]; consider the following.

In the CM problem the input data (C) (which describes, e.g., a resource being distributed or a requested collaboration query) will always hold to be true independent of (u) as the input data (C) will in all cases be correct. The prior probability P(C) is therefore estimated to be universally [1]. All output data (context properties that describe an individual(s) (u) context) will always hold to be true independent of (C) as the data for (u) will in all cases be correct. The prior probability P(u) for (u) is therefore estimated to be universally 1. In the CM problem the probability of a true (or alternatively a false) context match P(u – C) is unknown before CP given the highly dynamic and variable nature of the potential inputs (C) and output(s) (u). Thus, estimating a prior probability for (u) is not realistic. However, given these assumptions, it may be argued that there is an equal probability of an individual (u) being a true or alternatively a false match. Therefore, a prior probability is estimated to be 0.5 for each potential user (u) universally as set out in Bayes theorem. In considering the application of these estimated probabilities, a number of conclusions have been drawn.

As the probabilities are applied universally to every context property the overall effects on context processing for each context property, and therefore the resulting CM, will universally be nil. There are observed negative side effects on the context-matching process in that the application of the estimated probabilities will intuitively reduce the normalized result for the context match universally to a value lower than the value for the threshold (t). Addressing this issue will require additional steps in the proposed algorithm with respect to the value for (t). There will also be an unknown effect on the $\langle W \rangle$ parameter with potential consequences resulting in potential compromises in the ability to implement priorities and therefore enable effective CS.

The issues identified above in relation to probabilities in the application of Bayes theorem call into question the potential for Bayesian methods to address the demands of CP and the CM problem. An additional issue for Bayesian methods is the potential for significant computational overhead with the resulting tractability issues. Notwithstanding the factors discussed, given the observed similarities context processing using Bayesian methods remains an interesting direction for future research.

7.3.2 Inference and reasoning in context matching

The proposed approach, as currently constituted, does not utilize inference and reasoning, however, the *semantic context-modeling ontology* created using the OBCM approach using OWL DL along with the posited rule strategy has been designed to incorporate the ability to implement inference and reasoning in CP where the domain-specific design requirements call for such an approach. The proposed rule strategy uses logic functions such as AND, OR, and NOT in simple and complex rule structures where the ⟨*condition*⟩ may potentially incorporate multiple statements. Alternatively, multiple simple rules employing a Boolean structure similar to Horn style rules may be used. This capability and the portability of the proposed approach enables implementation in a range of rule engines using differing rule strategies. For example, the rule strategy adopted may be implemented using Java and Common Lisp in the Jess rule engine for the Java platform (Jess, 2010). These observations support the conclusion that inference and reasoning can be achieved based on the proposed approach where the domain requires it.

8 Conclusion and future work

While the research has addressed the requirements specification and the original design goals as discussed in Moore *et al.* (2010b), it has also identified a number of outstanding challenges and issues along with suboptimal solutions which remain open research questions as briefly discussed in the discussion. The approaches considered are thought to represent potentially useful directions for future research either singly or in hybrid solutions where personalization and PSP is required.

The difficulties in addressing the challenges and open research questions are not underestimated, however, resolving the challenges may provide an effective basis upon which intelligent context processing can be implemented with greater effectiveness and increased levels of computational intelligence.

References

Alter, S. 2008. Defining information systems as work systems: implications for the IS field. *European Journal of Information Systems* **17**(5), 448–469.

Ashford, R., Moore, P., Hu, B., Jackson, M. & Wan, J. 2010. Translational research and context in health monitoring systems. In *Proceedings of the Fourth International Conference on Complex, Intelligent and Software Intensive Systems (CISIS 2010)*, February 15–18, 81–86.

Berkan, R. C. & Trubatch, S. L. 1997. *Fuzzy Systems Design Principles: Building Fuzzy IF-THEN Rule Bases*, IEEE Press.

Berndtsson, M. & Lings, B. 1995. *Logical Events and ECA Rules*. Technical report, HS-IDA-TR-95-004, University of Skovde. `http://citeseerx.ist.psu.edu/legacymapper?did=77063`.

Boneh, D., Gentry, C., Lynn, B. & Shacham, H. 2003a. Aggregate and verifiably encrypted signatures from bilinear maps. In *Advances in Cryptology-EUROCRYPT 2003*, LNCS **2656** (47), 416–432.

Boneh, D., Mironov, I. & Shoup, V. 2003b. A secure signature scheme from bilinear maps. In *Topics in Cryptology-CT-RSA 2003*, LNCS **2612** (230), 98–110.

Checkland, P. & Holwell, F. 1997. *Information, Systems and Information Systems: Making Sense of the Field*, John Wiley and Sons.

Coppola, P. & Della, M. A. 2004. The concept of relevance in mobile and ubiquitous access. In *Mobile and Ubiquitous Info 2003*, LNCS **2954**, 1–10.

Dey, A. K. & Abowd, G. D. 1999. *Towards a Better Understanding of Context and Context-Awareness*. GVU Technical report, GIT-GVU-99-22, Georgia College of Computing, Georgia Institute of Technology.

Gonzales, A. J. & Dankel, D. D. 1993. *The Engineering of Knowledge-Based Systems Theory and Practice*, Prentice-Hall.

Hayes-Roth, F. 1985. Rule-based systems. *Communications of the ACM* **28**(9), 921–932.

Hildreth, C. R. 1998. On-line library catalogues as IR systems: what can we learn from research? In *Future Trends in Information Science and Technology*, P.A. Yates Mercer (ed.), Taylor Graham, 9–25.

Hong, J. Y., Suh, E. H. & Kim, J. K. 2009. Context-aware systems: a literature review and classification. *Expert Systems with Applications* **36**(4), 8509–8522.

Horrocks, I., Patel-Schneider, P. F., Boley, H., Tabet, S., Grosof, B. & Dean, M. 2004. SWRL: a Semantic Web rule language combining OWL and RuleML. http://www.w3.org/Submission/SWRL/.

Horrocks, I. P., Patel-Schneider, P. F. & van-Harmelen, F. 2003. From SHIQ and RDF to OWL: the making of the Ontology Web Language. *Journal of Web Semantics* **1**(1), 7–26.

Jena 2012. Jena – a Semantic Web Framework for Java. http://jena.sourceforge.net/index.html.

Jess 2010. Jess: the rule engine for the JavaTM Platform. http://www.jessrules.com/.

Kermarrec, A M., Massoulie, L. & Ganesh, A. J. 2003. Probabilistic reliable dissemination in large-scale systems. *IEEE Transactions on Parallel and Distributed Systems* **14**(2), 248–258.

Klir, G. & Yuan, B. 1995. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*, Prentice-Hall.

Lan, Z., Zheng, Z. & Yawei, L. Y. 2010. Toward automated anomaly identification in large-scale systems. *IEEE Transactions on Parallel and Distributed Systems* **21**(2), 174–187.

Lera, I., Juiz, C. & Puigjaner, R. 2010. OWL-M extension for semantic representations of ontology alignments. In *Proceedings of the Fourth International Conference on Complex, Intelligent and Software Intensive Systems (CISIS 2010)*, February 15–18, 956–961.

Lopez, J., Bellas, F., Alberto Pan, A. & Montoto, P. 2011. Trends in practical applications of agents and multiagent systems. *Advances in Intelligent and Soft Computing* **90**, 137–144.

Mitchell, T. M. 1997. *Machine Learning*, McGraw-Hill.

Moore, P. 2009. The complexity of context in mobile information systems. In *1st International Workshop on Heterogeneous Environments and Technologies for Grid and P2P Systems (HETGP'2009)*, August 19–21, 91–96.

Moore, P. & Hu, B. 2007. A context framework with ontology for personalized and cooperative mobile learning. In *Computer Supported Cooperative Work in Design III*. LNCS **4402**, W. Shen, J. Luo, Z. Lin & J.-P. A. Barthes (eds), Springer-Verlag, 727–738.

Moore, P., Hu, B. & Jackson, M. 2010a. Fuzzy ECA rules for pervasive decision-centric personalized mobile learning. In *Computational Intelligence for Technology Enhanced Learning*. Studies in Computational Intelligence **273**, F. Xhafa, S. Caballe, A. Abraham, T. Dardoumis & A. Juan (eds), Springer-Verlag, 25–58.

Moore, P., Hu, B. & Jackson, M. 2011. Rule strategies for intelligent context-aware systems: the application of conditional relationships in decision-support. In *Proceedings of the Fifth International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS 2011)*. Korean Bible University, June 30–July 2, 9–16.

Moore, P., Hu, B. & Wan, J. 2010b. 'Intelligent context' for personalized mobile learning. In *Architectures for Distributed and Complex M-Learning Systems: Applying Intelligent Technologies*, S. Caballe, F. Xhafa, T. Daradoumis & A. A. Juan (eds), IGI Global, 232–272.

Moore, P., Hu, B. & Wan, J. 2010c. Smart-context: a context ontology for pervasive mobile computing. *The Computer Journal* **53**(2), 191–207. http://comjnl.oxfordjournals.org/cgi/reprint/53/2/191.pdf.

Moore, P., Hu, B., Zhu, X., Campbell, W. & Ratcliffe, M. 2007. A survey of context modeling for pervasive cooperative learning. In *Proceedings of the International Symposium on Information Technologies and Applications in Education (ISITAE '07)*. IEEE, November 23–25, K5-1–K5-6.

Moore, P., Jackson, M. & Hu, B. 2010d. Constraint satisfaction in intelligent context-aware systems. In *Proceedings of the Fourth International Conference on Complex, Intelligent and Software Intensive Systems (CISIS 2010)*, February 15–18, 75–80.

Moore, P. & Pham, H. V. 2012. Intelligent context with decision support under uncertainty. In *Proceedings of the 6th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS 2012)*, July 4–6, 977–982.

Nurmi, D., Wolsk, R., Grzegorczyk, C., Obertelli, G., Soman, S., Youseff, L. & Zagorodnov, D. 2009. The eucalyptus open-source cloud-computing system. In *Proceedings of the 9th IEEE/ACM International Symposium on Cluster Computing and the Grid*, May18–21, 124–131.

Parsia, B., Sirin, E., Grau, B. C., Ruckhaus, E. & Hewlett, D. 2004. *Cautiously Approaching SWRL*. Technical report, University of Maryland, December. http://www.cs.uwaterloo.ca/gweddell/cs848/SWRL Parsi et al.pdf.

Pham, H. V., Thang, C., Nakaoka, I., Cooper, E. & Kamei, K. 2011. A proposal of hybrid kansei-som model for stock market investment. *International Journal of Innovative Computing, Information and Control* **7**(5(B)), 2863–2880.

Protege 2012. Welcome to Protege. http://protege.stanford.edu/.

Rimal, B. P., C., Eunmi & Lumb, I. 2009. A taxonomy and survey of cloud computing systems. In *Proceedings of the Fifth International Joint Conference on INC, IMS and IDC (NCM'09)*. Kookmin University, August 25–27, 44–51.

Saha, D. & Mukherjee, A. 2003. Pervasive computing: a paradigm for the 21st century. *IEEE Computer* **36**(3), 25–31.

Salem, B., Nakatsu, R. & Rauterberg, M. 2009. Kansei experience: aesthetic, emotions and inner balance. *International Journal of Cognitive Informatics and Natural Intelligence* **3**, 54–64.

Shackle, G. L. S. 1961. *Decision, Order and Time in Human Affairs*, Cambridge University Press.

Sheth, A. & Ramakrishnan, C. 2003. Semantic (Web) technology in action: ontology driven information systems for search, integration and analysis. *IEEE Data Engineering Bulletin*. Special Issue on Making the Semantic Web.

Sheth, A., Ramakrishnan, C. & Thomas, C. 2005. Semantics for the Semantic Web: the implicit, the formal and the powerful. *International Journal on Semantic Web & Information Systems* **1**(1), 1–18.

SPARQL 2007. SPARQL query language for RDF. `http://www.w3.org/TR/rdf-sparql-query/`.

SPARQL Update 2012. SPARQL 1.1 Update. `http://www.w3.org/TR/sparql11-update/`.

Thomas, A. M., Shah, H., Moore, P., Rayson, P., Wilcox, A. J., Osman, K., Evans, C., Chapman, C., Athwal, C., While, D., Pham, H. V. & Mount, S. 2012. E-education 3.0 challenges and opportunities for the future of iCampuses. In *Proceedings of the 6th International Conference on Complex, Intelligent, and Software Intensive Systems (CISIS 2012)*, July 4–6, 953–958.

W3C 2012. W3C Semantic Web activity. `http://www.w3.org/2001/sw/`.