Expert Systems with Applications 38 (2011) 10959–10965

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Online knowledge validation with prudence analysis in a document management application

Richard Dazeley [a,*], Sung Sik Park [b], Byeong H. Kang [b]

[a] Graduate School of Information Technology and Mathematical Sciences, University of Ballarat, Ballarat, Victoria 7353, Australia
[b] School of Computing and Information Systems, University of Tasmania, Hobart 7001, Tasmania

ARTICLE INFO

Keywords:
Knowledge acquisition
Knowledge maintenance
Prudence analysis
Validation
Knowledge based systems
Brittleness

ABSTRACT

Prudence analysis (PA) is a relatively new, practical and highly innovative approach to solving the problem of brittleness in knowledge based system (KBS) development. PA is essentially an online validation approach where as each situation or case is presented to the KBS for inferencing the result is simultaneously validated. Therefore, instead of the system simply providing a conclusion, it also provides a warning when the validation fails. Previous studies have shown that a modification to multiple classification ripple-down rules (MCRDR) referred to as rated MCRDR (RM) has been able to achieve strong and flexible results in simulated domains with artificial data sets. This paper presents a study into the effectiveness of RM in an eHealth document monitoring and classification domain using human expertise. Additionally, this paper also investigates what affect PA has when the KBS developer relied entirely on the warnings for maintenance. Results indicate that the system is surprisingly robust even when warning accuracy is allowed to drop quite low. This study of a previously little touched area provides a strong indication of the potential for future knowledge based system development.

1. Introduction

Knowledge based researchers have long battled with the issue of brittleness, which has either directly or indirectly led to the majority of methodological developments in symbolic based reasoning. Yet it is an issue that still remains in varying degrees in most systems today. In knowledge based systems (KBS) brittleness occurs when a system is asked to inference a situation that is beyond the knowledge captured in the knowledge base. This brittleness is particularly problematic in KBSs because the system does not display an error or crash when it occurs. To the user there is generally no discernable problem as the system still produces a response. Therefore, if the user is not sufficiently knowledgeable to be able to notice the response is flawed then they may treat it as correct. For instance, a nursing assistant may not challenge a response and thus not seek an expert's opinion, resulting in an incorrect treatment of a patient.

A brittle inference tends to occur when a request from outside the systems knowledge domain occurs or the domain of operation is missing knowledge. The cause of such inadequacies is often seen as being due to the concentration of specialised knowledge in the target domain for the particular system (1991). The majority of research into resolving brittleness could be grouped into one of three areas:

- Finding deeper levels of knowledge. For instance, methodologies, such as Knowledge Acquisition and Design Structuring (KADS) (Wielinga, Schreiber, & Breuker, 1992), were developed to help extract deeper forms of knowledge.
- Forming a layer of general knowledge to use when the specialised knowledge was inadequate, such as Cyc (Lenat, 1995; Matuszek, Cabral, Witbrock, & DeOliveira, 2006). This provides the opportunity to fall on levels of general knowledge when the domain specific knowledge falls short.
- Measuring the completeness of a knowledge base through methods of verification and validation (V&V) (Preece, 2001).

All three approaches have, however, provided significant challenges. For instance, methods of finding deep knowledge still do not tell us when we have it all. At some point the system must be made available to users and will always run the risk of missing knowledge. Additionally, if there is any shift in the domain's knowledge the system will require significant rework. Likewise, systems such as Cyc, fail to be able to adjust to the constant changes in one of the most contextually dynamic knowledge domains – general knowledge (Dazeley & Kang, 2008c). Thirdly, V&V methods tend to either perform with only known cases, therefore not checking unknown, or check all combinations of attributes which reveals many situations the system does not need to know about.

* Corresponding author.
E-mail addresses: r.dazeley@ballarat.edu.au (R. Dazeley), sspark@utas.edu.au (S.S. Park), bhkang@utas.edu.au (B.H. Kang).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.02.139

Fundamentally, validation is attempting to identify whether all the possible cases are covered by the KBS. Alternatively in the subfield of anomaly detection a system is analysed holistically to find structural anomalies, such as redundancies, conflicts or dead ends (Kusiak, 2000). These approaches are a static analysis of the system at a moment in time, usually during development, and are not generally applicable online while the system is in use. Prudence analysis (PA) is a form of dynamic online anomaly detection which uses actual cases as they are presented to determine if knowledge from outside the knowledge base is required. Therefore, PA validates real situations as they occur by detecting when an inferenced solution to a case may be wrong. The advantage of such an approach is that a system can provide a warning system for the user indicating that an inference goes beyond its knowledge domain. Essentially, PA uses a form of meta-knowledge to validate each inference. The expectation is that the user will then check the case being inferenced with a human expert that can validate the correctness of the inference by the system.

Currently, PA has only been studied by a minority of researchers, all of whom have centred their studies on a single family of KBSs, referred to as ripple-down rules (RDR) (Compton & Jansen, 1988). The primary reason for this is that RDR is an incremental KA and maintenance methodology. It is RDR's flexible and maintainable structure that makes it ideal for PA. Early work on PA such as WISE (Edwards, 1996; Kang, 1996), feature recognition prudence (FRP) (Edwards, 1996; Edwards, Compton, Kang, Preston, & Lazarus, 1995; Edwards, Kang, Preston, & Compton, 1995) and feature exception prudence (FEP) (Edwards, 1996; Edwards et al., 1995; Edwards, Kang et al., 1995) failed to deliver significant accuracy.

A new approach was taken by Compton, Preston, Edwards, and Kang (1996) of comparing cases with previously seen cases within context, and provided warnings if they differed in some unusual way. This simple method achieved a reasonably high level of accuracy on some datasets with significantly less false positives. This was subsequently followed by a Ph.D. thesis by Prayote (2007) which continued Compton et al.'s (1996) work by including a number of improvements, which allowed a reduction in the number of false positives. This study's results indicate a significant reduction in false positives and more accurate warnings. One of the main problems with both of these approaches was their reliance on an attribute's existence or absence in a case for the generation of warnings. This limits the methods ability to be applied primarily in domains with a controlled number of only relevant attributes. Domains with large amounts of irrelevant attributes such as free text classification will tend to produce a large amount of false positives.

The most recent study by Dazeley and Kang (2008b, 2008d) tried a new dynamic approach referred to as Rated MCRDR (RM) (Dazeley & Kang, 2003, 2009) by using a neural network approach to learning meta-knowledge. Dazeley and Kang (2008b, 2008d) concluded that the system was able to predict errors more accurately without increasing the false positives. More importantly though was that it contained two additional advantages over previous approaches. Firstly, it was versatile – previous approaches have a preset level of accuracy determined by the algorithms approach which cannot be altered, whereas RMs accuracy and number of false positives could be controlled. Secondly, if the system misses a case the system can still warn about similar cases in the future.

These early studies of RM however used simulated experts based purely on artificially generated knowledge, primarily using an inducted decision tree. This paper will provide details of a follow up study on this technique of PA by applying it in a domain using real data and human expertise. In this study we use an application referred to as MonClassifier which is part of a larger suite of applications called personalized web information management system (PWIMS) (Kim, Park, Deards, & Kang, 2004; Kim, Park, Kang, & Choi, 2004; Park, Kim, & Kang, 2003, 2004a, 2004b). This application has been used to collect and classify a number of free text knowledge domains. For each of these domains a knowledge base has been incrementally built using human experts, such as the collection of eHealth articles used in this study. However, this process is exceptionally time-consuming, requiring the expert to read and check every article manually. This study provides an opportunity to empirically study the viability of using PA to significantly reduce the experts load, potentially opening up numerous application domains for the use of expert knowledge.

The following section discusses the PWIMS and MonClassifier applications, highlighting the fundamental problem with expert system development in such an application. It is this problem that PA is able to solve, which will be discussed in Section 3. In order to study PAs effectiveness in such a domain two experiments were performed, which are described in Section 4 with results and discussion in Section 5.

2. Application domain

The personalized web information management system (PWIMS) provides support for dynamic and personal web portals in a simple suite of applications. The platform contains three main components:

- a Web monitoring agent
- a storage management (or knowledge management) component
- a knowledge sharing agent

The Web monitoring agent monitors a number of user-specified websites for newly uploaded pages. When a new page is uploaded the system retrieves the page and stores it in the database. Therefore, this system is ideal for monitoring sites such as news or research portals. When pages are gathered the storage management component is used by a domain expert to classify each article appropriately. The system can then redistribute the classified articles to a web page or push relevant pages to clients. Fig. 1 shows the PWIMS architecture.

In this study we were interested in investigating the ability of PA of reducing the knowledge acquisition effort in a domain using human expertise. Therefore, the Web monitoring and knowledge sharing agents are not relevant to this study and are not discussed further. The only component of PWIMS that is directly relevant to this study is the central storage management component referred to as MonClassifier. This sophisticated application takes each gathered web page (case) and presents it to the expert who classifies each in turn. Generally, this simply involves accepting the offered classification, but occasionally may require a reclassification. MonClassifier uses MCRDR and therefore is able to classify each web page into any number of possible classes. These classes are actually folders organised in a tree like hierarchy in a similar fashion to Windows Explorer Fig. 2. The expert actually only really needs to understand the folder organisation and does not require any understanding of the underlying knowledge base.

When the expert feels an article has been misclassified they can correct it via the KA interface provided. A new class can be created by the expert by simply creating a new folder. The expert can then create rules by simply selecting words from a difference list. Therefore, the MCRDR engine allows the expert to select the words or phrases that they believe adequately separate the cases within the current context. The MonClassifier application then stores the article in a MySQL database. Additionally, the database stores all relevant information required about the knowledge stored and the relevant cornerstone cases.