VL-PATSy: Facilitating vicarious learning via intelligent resource provision

Richard Cox 1, Jianxiong Pang

Department of Informatics, University of Sussex, Brighton, UK

Abstract. This paper describes an adaptive system called VL-PATSy, an extension to an existing system (PATSy) that adds a mechanism for serving vicarious learning (VL) resources. Vicarious learning is the notion that people can and will learn through being given access to the learning experiences of others. The VL resources represent an innovative form of learner support for a complex cognitive task (clinical reasoning). The VL resources are delivered in response to system-detected reasoning impasse events or in response to student demand in timely and context-sensitive ways. A three tier XML-oriented, rule-based architecture was chosen as an efficient and lightweight implementation solution and a design pattern approach was utilised.

Keywords. vicarious learning, clinical reasoning, clinical education, event notification, design patterns, case-based teaching, intelligent tutoring, adaptive systems, metadata

1. Introduction

This paper describes a rule-governed, adaptive, educational support system and its associated core pedagogical methodologies in the clinical domain of speech and language therapy (SLT). The system has been developed as part of a project called VL-PATSy 1. The overall aim of VL-PATSy is to provide vicarious learning resources to trainee Speech and Language Therapists (SLT’s). Vicarious learning is the notion that people can and will learn through being given access to the learning experiences of others [1,4,13]. Simple traditional examples are instances such as master classes in music, and the process of clinical teachers going through cases with students. In VL, one student is the focus of tutorial attention, but others present also benefit as a result of observing the interaction. Other learners facing similar problems will pose issues within a similar “zone of proximal development”, fostering an empathy which allows the vicarious learner to key into

1 Correspondence to: Richard Cox, Department of Informatics, University of Sussex, Falmer, Brighton, BN1 9QH, UK. Tel.: +44 1273 67 8605; Fax: +44 1273 87 7873; E-mail: riche@sussex.ac.uk

1 www.tlrp.org/proj/phase111/cox.htm
the acquisition process. VL offers an opportunity to preserve conversational elements and dialogue in e-learning contexts i.e. important components of Laurillard’s ‘conversational framework’[10]. VL also has potential for facilitating professional enculturation. The idea is that dialogue with expert practitioners helps induct the learner into the forms of discourse characteristic of the profession. In our project, the VL resources take the form of a database of video clips stored, indexed and disseminated as a new resource added to an established e-learning system (PATSy2). The database, together with other architecture modules, form VL-PATSy. The VL resource repository consists of a large (approx. 180) set of digital video clips. Each clip is a recording of either 2 students or a student and a tutor engaged in dialogue. Each video clip records a discrete clinical reasoning event (e.g. how to decide what diagnostic test to administer next). The participants in the VL video clips are interacting with virtual patient cases on the PATSy system [3,11,12]. The idea is that if a student using PATSy demonstrates inappropriate diagnostic reasoning or becomes stuck (reaches a reasoning impasse3), the system detects it and offers a choice of relevant VL resources (Figure 1).

2. The PATSy e-learning system

PATSy is a web-based generic shell designed to accept data from any discipline that has cases. The disciplines represented on PATSy currently include developmental reading disorders, neuropsychology, neurology/medical rehabilitation and speech and language pathologies. Sixty-one data-rich and extensively described cases of adults and children with disorders are accessible under 4 domain headings - speech and language, dyslexia, medical rehabilitation and neuropsychology. The system is used by more than 30 university departments including 80% of UK Speech and Language science departments. PATSy is designed for use in conjunction with more traditional methods of clinical training, professional education and academic teaching about disorders. As a multimedia database (video, audio, images), it serves as an electronic archive for case-based research as it contains rich collections of test data. It offers health science students an opportunity to practice diagnostic reasoning on ‘virtual patients’. Students can access real patient data in the form of videos, assessments, and medical histories. PATSy’s virtual patients can have psychometric and cognitive assessment tests ‘administered’ to them by learners. Typically, PATSy is used in blended learning contexts. Students attend clinical placements and tutors integrate online PATSy sessions with lecture content. PATSy complements taught content and clinical placements in important ways. Tutors are provided with a way of exposing a class of students to the same cases (clinical placements don’t always expose students to a wide range of cases and student experiences on placements can vary widely). PATSy also contains published and rare cases that can bring the journal literature alive for students. Lecturers can ‘hold back’ PATSy cases. Students can then be assessed dynamically and authentically as they diagnose a previously-unseen case using PATSy in its exam mode.

PATSy has been designed to meet the needs of both students in initial clinical education (ICE) as well as by professional clinicians undergoing continuing professional

2www.patsy.ac.uk

3A clinical reasoning impasse is defined in [7] as ‘an inability to apply formal knowledge in a clinical setting’.
development (CPD). This has been achieved by a learner-centred approach to the design of PATSy’s modes of use. Clinical reasoning is difficult to teach via direct instruction and it is difficult to give students a set of rules that they may universally apply [9,16]. For this reason clinical science education (like law and other professions) is tending to adopt case-based teaching methods. Knowledge of the subject is represented in the form of many example cases [8,15]. PATSy is particularly well-suited to case-based teaching and for use in problem-based learning contexts. Case-based reasoning requires extensive experience and a large case-base for effective use. PATSy helps students acquire a ‘critical mass’ of cases and scaffolds them in their transition from predominantly hypothetico-deductive modes of reasoning to more heterogeneous, expert-like modes. In PATSy’s ICE modes, students are encouraged to reason in hypothetico-deductive mode to a greater extent than is the case for experienced clinicians (e.g. CPD users). Tutors may configure PATSy such that ICE users are discouraged from top-down test ‘score pattern spotting’ whereas CPD users are free to do so.

PATSy has built-in student-system interaction recording, making it an educational research instrument for the microgenetic study of students’ clinical reasoning skill acquisition. The log records in detail which tests are accessed and the order in which they were ‘administered’ to virtual patients. The system has been used to study students’ and experts’ diagnostic reasoning. A range of reasoning difficulties has been identified and a 6-level diagnostic accuracy scheme has been developed [2,6,7]. PATSy can also be configured to periodically prompt students with questions such as: ‘What conclusions can you draw from what you have just seen?’; ‘Where do you want to go next (further test data?, introductory video?)?’, ‘Why?’ Students’ typed responses to these prompt questions are interleaved with the system-generated log data. As mentioned above, the PATSy system has recently been extended by the addition of VL support resources.

**Figure 1.** VL-PATSy suggests a range of video clips to a student using PATSy following its detection of a clinical reasoning impasse or event. The PATSy user has chosen a video clip of two students discussing the clinical reasoning topic ‘heuristics for generating hypotheses’. 
Development of VL resources to support students’ clinical reasoning

Task-directed discussion (TDD) exercises were used as a methodology for generating pedagogically effective dialogues for use as vicarious learning resources. TDDs are language games of a type commonly used in English-as-a-second-language (ESL) teaching [14]. A TDD is a discrete language activity with a set goal [13]. They are useful as a means of generating discussion where spontaneous discussion is infrequent and where capturing naturally occurring dialogue is difficult and uncontrollable. Compared to open classroom or small-group tutorial contexts, the participation by pairs of participants in TDD exercises is designed to ensure a high degree of topic focus and increased amounts of discussion, together with deeper explorations of domain concepts. A typical domain-specific TDD topic might be one which elicits discussion about the definition of a concept. (e.g. ‘Please provide a definition of paraphasia’) or comparisons between concepts (‘Discuss when the Howard and Franklin picture reading test might be a better choice than the ADA reading-of-words test in diagnosing comprehension difficulties in an aphasic patient’). Other TDD examples focus on the use of professional register, e.g. ‘Read the following description of a client’s communication difficulties and then rephrase it in professional terminology’.

An extensive collection of TDD-derived dialogues (approx. 180) have been recorded, edited and incorporated into a structured database. VL-PATSy is a mixed initiative system. Students may browse VL-PATSy’s VL resources at any point as they work through a case. The video clips are tagged with metadata and the learner’s interaction with PATSy can fire rules which result in students being offered dialogues to view when they strike a reasoning impasse. The system monitors students’ interactions and intervenes where necessary. Event-condition-action (ECA) rules fire if certain events are detected. To illustrate, if a student administers an unusual and inappropriate sequence of language tests to a virtual patient, then a rule fires and VL video clips are proactively offered on relevant topics such as ‘hypothesis testing’. Clips with content tags such as ‘a student and tutor discuss how understanding what a test does helps with planning a test strategy’ and ‘a student and tutor discuss confirming and disconfirming a hypothesis’ might be offered (e.g. Figure 1). In research currently underway, VL-PATSy is being used as a research test-bed for comparing the relative efficacy of VL clips consisting of either student-tutor or student-student dialogue.

3. The architecture of VL-PATSy

The requirements of the VL-PATSy system are for an architecture exhibiting a high degree of independence and separation from the existing PATSy system. Hence a primary consideration of the design for VL-PATSy was for it to run separately from the existing PATSy system and for it to maintain its links to PATSy via message exchange. It was also desirable that the system should be resilient, scalable, extensible and maintainable. A three-tier architecture was employed. This consists of (1) a frontend tier i.e. an information, or content presentation layer, (2) a middle tier i.e. a monitor and notification mechanism, and (3) a backend tier in the form of multi-role server components and persistence facility (Figure 2). Three-tier architectures are common in many web-based applications and allowed us to choose from a wide range of existing tools and design patterns.
They are well-suited to a rule-based architecture - a rule-engine is by nature a server-class software component. Three-tier architectures have been in use in several different domains. For example, in the enterprise computing area, the 3-tiers consist of the client side (web browser or proprietary client), middleware services (messaging, persistence, security, caching and event notification) and a backend (i.e. server-side components).

However, VL-PATSy’s architecture differs from these examples in several key respects. The Server can extend its services by allowing domain experts (expert speech and language clinician educators in our case) to add more rules to the rule store. Domain experts can define functionality via rules. The front end and rule server components apply context information. The rule engine works with the HTTP service but is independent of the underlying web application. The architecture makes the underlying HTTP transparent to the domain experts. It enables domain experts to enter functionality and domain-specific knowledge into the tutoring system and provides communication channels between rule servers that can be used for interaction handling. The front end tier represents the chosen web application and HTTP communication facility. It is a combination of the presentation layers of both the PATSy and VL-PATSy systems. Three features of the front end tier are: (1) The rule servers are provided with frequent messages about the learner’s server requests as soon as they arrive at the PATSy server, (2) VL-PATSy’s responses (e.g. to present information to the learner) are not driven by user request directly, but indirectly - the information that is presented is driven by the learner’s activities, but not by the learner’s direct requests. Finally, (3) low-level learner behaviours (as recorded by PATSy’s user-system logs) are mapped onto more abstract ECA events. For example, a student may indicate premature commitment to a very specific hypothesis and begin his or her testing of the patient with a narrowly focussed test instead of starting out with a broad test. An abstract ECA rule might correspond to the statement ‘begin diagnosis using broad tests, and subsequently test specific hypotheses with more narrowly-focussed tests’. For each of the 28 virtual patient cases on PATSy, there would be many particular test sequences that might fire this abstract rule.
Note that a major requirement was for the ‘push’ of pedagogical advice from the VL-PATSy system to run concurrently and asynchronously, without blocking or interrupting the PATSy system’s normal mode of operation. The middle-tier is a monitoring and notification mechanism (i.e. a ‘spy’). One of the most important interactions between PATSy and VL-PATSy occurs via the processing of log data. The middle-tier must mediate between PATSy and VL-PATSy in order to relate the user-system interaction log data from PATSy to the delivery of vicarious learning content by VL-PATSy. Two crucial constraints are imposed on the middle-tier - (1) The rule servers must be informed of events that happen in each learning session and (2) PATSy must run normally without noticeable loss of performance. VL-PATSy responds to particular student-PATSy interaction events. If an event matches certain conditions then actions result. The actions cause an appropriate subset of VL resources (annotated video clips) to be offered to the student.

ECA rules have been used for defining the mapping relation between learner behaviours and teaching content delivery using XML standards. The interactions with VL-PATSy must be transparent to the PATSy users. This means that VL-PATSy must not interfere with event routing or execution in the PATSy system. PATSy’s user-system event logging system dynamically builds a publisher/subscriber design pattern in the rule engine. Rule objects are subscribers and they can subscribe and de-subscribe to PATSy events in run time. These design decisions have proved successful in a number of experimental runs of the system. VL-PATSy’s backend tier components include a database module, which is responsible for the management of metadata for video clips and ECA rules written in XML. Rule objects are subscribers and they can subscribe and de-subscribe to PATSy events in run time. These design decisions have proved successful in a number of experimental runs of the system. VL-PATSy’s backend tier components include a database module, which is responsible for the management of metadata for video clips and ECA rules written in XML. Rule objects are subscribers and they can subscribe and de-subscribe to PATSy events in run time. These design decisions have proved successful in a number of experimental runs of the system. VL-PATSy’s backend tier components include a database module, which is responsible for the management of metadata for video clips and ECA rules written in XML.

3.1. VL-PATSy rules

The rule specification language is designed to have adequate expressibility for SLT clinical reasoning using XML mark-up language. An example of a pedagogical rule for detecting and responding to a student’s test knowledge impasse might be ‘when a student moves from test X to test Y, but X→Y is an inappropriate testing sequence, then offer relevant VL video clips’ (Figure 3).

![Figure 3. The ‘TESTS_BAD_FOCUS’ rule in VL-PATSy](image-url)
Each rule is composed of a trigger block and an action block. A basic format of a rule is:

\[
\text{when <some situation occurs> do <something>}
\]

Of course, VL-PATSy’s pedagogical rules may conflict and conflict resolution may be required. All rules are divided into two groups, group one is called ‘top priority’ rules - rules in this group must be fired if trigger conditions hold; group two is called ‘normal priority’ rules, if several rules fire simultaneously, then one is chosen randomly and the others are discarded. Event patterns as triggers are an important data type that represent learner’s behaviours. In the case of Figure 3 the TESTS_BAD_FOCUS rule is assigned ‘normal priority’. When a learner selects two tests X and Y, the TESTS_BAD_FOCUS rule is fired if and only if Y is not a (clinically) reasonable subsequent test to X. The ‘action’ block in Figure 3 contains an instruction to deliver a list of vicarious learning video clips. For example, TESTS_BAD_FOCUS rule specifies that those video clips should be tagged as DS (domain specific), and HT (hypothesis testing) and TS (deliver after a series of tests has been administered to the patient).

4. Related work

In intelligent-tutoring (ITS) terms, VL-PATSy is similar to a model-tracing tutor - e.g. Ms. Lindquist [5], an algebra tutor. It has a student model and a tutorial reasoning model that produces a human-like interactive dialogue to help students learn how to write algebraic expressions for word problems. The remediation feature is similar to the failure advisor rules of VL-PATSy. Ms. Lindquist utilises a continuous running ‘conversation’ and its data structures have been designed around a need for dialogue. In VL-PATSy, this kind of dialogue is less important, and VL-PATSy focusses more on monitoring the student’s activities and responding with suitable VL offerings in the form of links to relevant VL video clips. Interestingly, Ms Lindquist has a separate tutorial model that encodes pedagogical content knowledge in the form of different tutorial strategies and this module was partially developed by observing an experienced human tutor. This feature is in many senses similar to VL-PATSy’s design idea of pedagogical rules in that each cause-result relation has been specified by domain experts on the basis of their clinical expertise. However, unlike VL-PATSy, and given the non-trivial task of setting up a constraint/rule base, Ms. Lindquist relies heavily on authoring tools. To use Ms Lindquist users need to learn how to use its authoring tool whereas VL-PATSy provides a lighter-weight rule authoring system for its domain experts to use.

5. Conclusions

An adaptive system called VL-PATSy has been described. It represents an extension to an existing system (PATSy) which adds a mechanism for intelligently serving vicarious learning (VL) resources. The VL resources are an innovative form of learner support

---

4A major source of rule-firing conflict tends to happen when students type reflective notes into PATSy’s log. For example, two rules might fire due to a) the phrasing of the entry failing to demonstrate the use of professional register and b) the entry consisting of fewer than 200 characters.
during complex cognitive tasks (clinical reasoning). They are delivered a) in response to system-detected learning events or b) on student demand in timely and context-sensitive ways. A three tier XML-oriented, rule-based architecture was chosen as an efficient and lightweight solution.

Acknowledgements

The support of the Economic and Social Research Council is acknowledged (ESRC TLRP grant RES-139-25-012). Thanks to colleagues Professor Rosemary Varley, Dr John Lee, Dr Julie Morris, Kirsten Hoben, Dr Barbara Howarth & Susen Rabold.

References

