

Personal Learning Environment

Mafawez T. Alharbi, Amelia Platt, Ali H. Al-Bayatti
*Software Technology Research Laboratory
De Montfort University, Leicester, UK*

Abstract

Virtual Learning Environments (VLE) have become popular in higher education in recent years due to their ability to provide additional and flexible solutions for students and researchers. However, the limitations of VLEs have led to the development of a new generation of VLE – the Personal Learning Environment (PLE). PLEs avoid these limitations and have new features that allow students to control and develop new applications, such as Web 2.0 applications and social networks. Whilst PLEs have resolved some of the drawbacks of VLEs, it is argued that PLEs also have greater potential to cover a wider range of aspects. This paper presents a proactive context-aware architecture for PLE supporting two major objectives: lifelong access and learner-centric study, covering both traditional formal (institution-based) and informal (private, non-institution-based) academic learning. Bayesian Networks are graphical modeling tools that have been used for modeling uncertain knowledge. Moreover, BN has been used in this research to implement the proposed architecture.

1. Introduction

E-learning is defined as the use of technology to enable people to study anywhere and at any time. Numerous examples of e-learning date back several decades. It may be argued that the advent of VLEs made the packaging of a diverse set of e-learning services to support teachers and learners both easy and affordable. Virtual Learning Environments (VLEs) are currently a core component of education in most teaching establishments from primary level through to higher education. Whilst VLEs provide many benefits [1], they also have a number of limitations [2]. In particular, VLEs are institution-centric; they are ‘owned’ by the institutions that manage them and are designed to support formal learning. Furthermore, teachers control the materials and services that are made available to learners. Typically, a learner enrolled on a module or course has access to the materials and services for that module or course for the duration of their enrolment; VLEs do not, therefore, support lifelong learning [3]. These limitations have led researchers to define

approaches to the development of Personal Learning Environments (PLEs).

2. Current Proposals and Implementations of PLE

2.1. Related work

PLEs share many of the characteristics of Web 2.0 tools, such as the ability to create, publish, share and distribute information easily and to establish web-based communities that facilitate collaboration using various social networking sites. PLEs based on Web 2.0 tools are thus increasingly popular [4].

Researchers have begun to define the requirements of PLEs [5, 6]. A proposal in [7] describes PLEs from a pedagogic perspective. The proposal describes the functionality, purposes and challenges of PLEs. As part of the research, a study of 33 students from vocational and polytechnic level schools in eastern Finland was conducted. The main objective of the study was to give these students the opportunity to design their own PLE technologies, such as blogs, websites and Web 2.0 services. The students were required to keep blogs about the development of their PLE, which were analysed at the end of one year of study. It was found that the students spent most of their time learning how to use the tools needed in order to build their PLEs and used their PLE in much the same way as a VLE.

In [8], a framework for PLEs that enable users to control their own learning is proposed. The study argues that a PLE is an intersection of VLE, Web 2.0 and portfolios. The study noted that the flow of information in VLE communication moves largely in one direction, as learners rely on instructors to direct their learning. In contrast, learners in PLEs rely on their peers to direct their learning, resulting in two-way communication. The advantage of this proposal is the support available for sharing between learners. The framework proposes a tag view of all the elements in the PLE and also uses OpenId. Its main limitation is that the framework gives little detail as to how it could be implemented.

Colloquia are another example of a PLE that allows learners to control their personal information,

maintain learning resources and set up activities [9]. Each participant in the system acts as a creator and receiver of resources, so both learners and teachers share the same system. It aims to move learners from a traditional classroom environment to a group conversation based on learning and to provide learners with mobile and personal tools that enable them to work offline.

2.2. PLE from the perspective of this study

While a mature set of user requirements for PLEs has not yet been defined, two major objectives have nevertheless emerged: PLEs must provide lifelong access and they must be learner-centric. The first tentative PLE platforms to emerge have largely been based on Web 2.0 applications and ideas, taking advantage of the two-way communication characteristics that these provide as well as the ability of a user to produce (share) and consume information resources easily. However, it is argued that these tend to consider PLEs as extensions of VLEs and to regard PLE users as learners typically associated with institutions. The definition of a PLE user in this study is anyone who is a producer or consumer of e-information. This covers the use of a PLE for many purposes in addition to traditional formal (institution-based) and informal (private, non-institution-based) academic learning. For instance, the PLE could be used for social or business purposes. A business person may use the PLE to ensure they are automatically informed of new contracts and tenders of interest to their business (consumer information) and respond by publishing a subcontract (producer information) to collaborate with other businesses as part of the response to the tender.

PLEs can therefore accommodate a diverse set of users and interests and must provide flexibility and satisfaction to the user. One consequence of this is that the user must be able to control the information he or she owns, produces, publishes, shares or gathers. This will require the user to make decisions such as who to share his or her information with, how others sharing it may use it and for how long and in which location it should be stored. Similarly, the user must be able to configure the set of tools which he or she wishes to use for their PLE. It is argued that a context-aware PLE can support these requirements.

2.3. Context-aware system

Ubiquitous computing (pervasive systems) was first proposed by Weiser (1991). Context-aware systems are a type of pervasive system and are viewed by computer scientists as a mature technology [10, 11]. A definition for context is given

by Day in [12]: “context is any information that can be used to characterise the situation of an entity, an entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and application themselves”. Context-aware systems are able to gather contextual information from a variety of sources without explicit user interaction and adapt their operation accordingly [13]. Context-aware systems have the ability to integrate easily with any service domain, such as healthcare, commerce, learning and transport.

A context-aware system must include three essential elements: sensors, processing and action. Three types of sensors are defined: physical, virtual and logical [14]. A physical sensor, such as a camera or thermometer, captures information about its local environment [15]. In contrast, virtual sensors extract information from virtual space, which is defined as the set of data, applications and tools created and deployed by the user. Logical sensors combine physical and virtual sensors to extract context information. For example, a company can infer that an employee is working from home using login information (a virtual sensor) and a camera (physical sensor) [10].

2.4. The advantages of a context-aware PLE

A context-aware PLE will automatically take the context of each user into consideration. The user is able to define his or her individual interests. The system will support a diverse set of users rather than formal learners only. The PLE architecture is institution-independent and supports users' lifelong individual requirements. The proposed PLE architecture will provide an interface with Web 2.0 applications.

3. Proposed architecture

A proactive context-aware PLE architecture is presented in this section. A high-level architecture is shown in Fig 1. It consists of two layers: the top layer is the PLE service – the PLE interface for the user – while the bottom layer shows other independent tools or service providers selected by the user. The two layers communicate with each other as follows:

- Top layer: PLE service

The PLE service consists of three main entities: Personal Manager, Context-Aware Engine and User Profile. These entities communicate with each other to provide the user with a service tailored to their individual needs. A user interfaces with the PLE using this layer.

- Bottom layer: Provider

This layer consists of various tools and independent service providers. It provides virtual sensors (context information) that drive the PLE service.

The providers represent the set of independent services (for example Facebook, YouTube, Personal Calendar and IEEE Xplore digital library) defined in the User Profile (see Section C: User Profile). These are accessed by the user through the Personal Manager on demand. The responsibility of the provider is to supply the Context-Aware Engine with appropriate context information that is consistent with the User Profile.

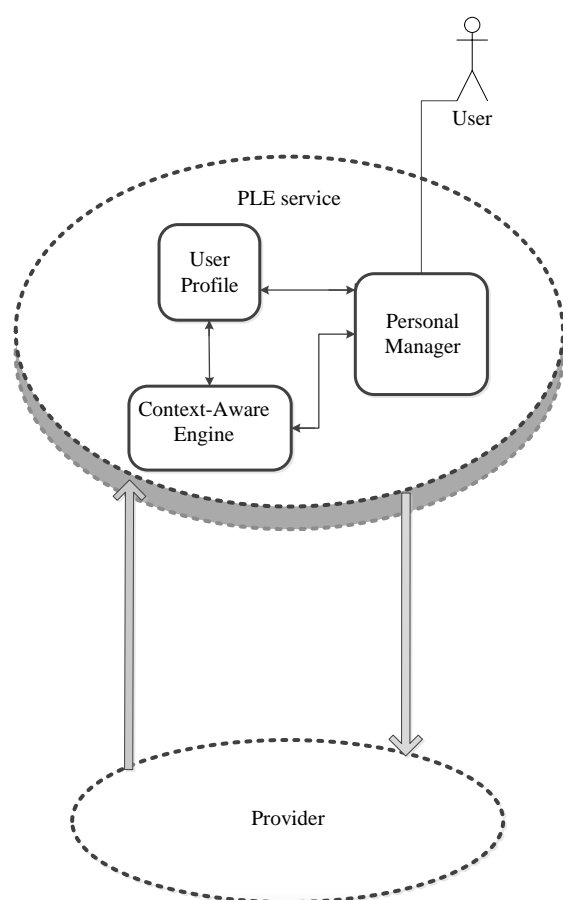


Figure 1: Overview of the proposed architecture

3.1. Personal Manager

The only way for the user to interact with the PLE service is through the Personal Manager. It is envisaged that the Personal Manager system will be deployed as an application on standard mobile devices. The main functions of the Personal Manager are:

- To allow the user to manage his or her profile.

- To present to the user any new information gathered by the Context Reasoner.
- To allow the user to access independent service providers.
- To enable other users to access gathered information from the user profile as a public portal.

In summary, the Personal Manager will coordinate all of the user's e-communications.

3.2. Context-Aware Engine

The Context-Aware Engine is responsible for filtering and interpreting the context information (for example virtual sensor information) produced by the tools or services at the bottom layer. This filtering is performed using the Context History together with the User Profile information provided by the user. This ensures the user receives only relevant information.

Fig 2 shows the components of Context-Aware Engine: Acquisition, Context History and Context Reasoner.

3.2.1. Acquisition. The main purpose of the Acquisition is to gather or isolate raw data (virtual sensors) originating from the independent service providers in the bottom layer of the architecture and send it to the Context Reasoner. The Acquisition must therefore have an API to interact with the independent service provider.

3.2.2. Context History. The Context History subsystem is responsible for the storage of high-level context information that has been delivered to the user previously. This storage is essential to prevent duplicate information being sent to the user at a later point in time. Furthermore, the Context History will help improve the Context Reasoner functionality by providing a more detailed history of the user's previous selections.

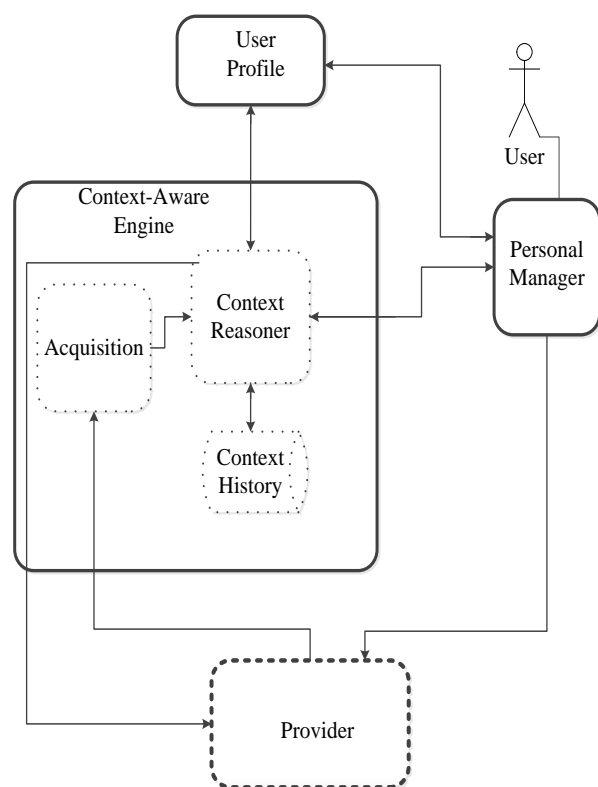


Figure 2: Context-Aware Engine

3.2.3. Context Reasoner. The Context Reasoner is responsible for using context information in an intelligent way and is the most challenging function for the Context-Aware Engine. Fig 3 shows the functionality of the Context Reasoner and its role within the Context-Aware Engine in providing the PLE service.

The overall responsibility of the Context Reasoner is to extract relevant high-level context information from the raw data using the information stored in the Context History and the User Profile.

In this approach, the Context Reasoner consists of the following components: Detector, Profile Detector, Generic and Generator.

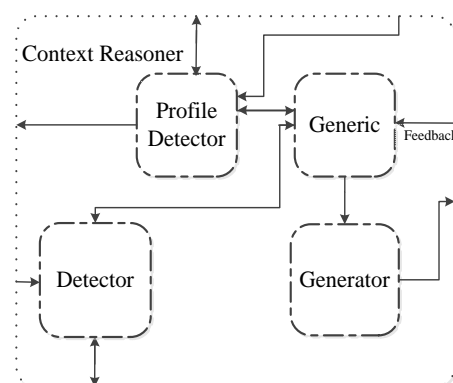


Figure 3: Context Reasoner

• Detector

The Detector continually receives context information from the Acquisition. It uses the Context History to determine whether this is new context information that has not been delivered to the user previously. If the context information is not new, the Detector is directed to stop processing; otherwise it will continue. Clearly, the Detector must contain an underlying mechanism that makes it possible to recognise the context.

• Profile Detector

The Profile Detector is connected with the user profile. It is responsible for providing both the Provider and the Generic function with information about the user.

• Generic

The Generic function is to predict or suggest new user interests based on existing context information stored in the Context History, together with information stored in the User Profile and user feedback. This function therefore has the ability to learn from the habits of the user.

Amazon recommendations are a good example. When a customer purchases a book from Amazon, the website recommends a similar book the next time he or she logs in. The other responsibility of the Generic function is to update the Context History with details of the action and check that similar context has not previously been rejected by the user.

• Generator

The Generator is the component that creates or produces an action based on input from the Generic function. The action should be delivered to the user

via the Personal Manager. For instance, the action may send information to the Personal Manager to notify them that a new photo of interest to the user has been posted on Flickr.

Now we will focus on Generic part. The Generic system consists of five main elements: Analysis Feedback, Relevant, Rejected, Prediction and Similarity. The components are described below:

1) Analysis Feedback

It is responsible for distributing the feedback depending on the rank level that has been received from the user of the system. The ranking starts from irrelevant and continues through to very relevant.

2) Relevant

This element contains all the high ranking feedback; it reflects the requirements from the outside world. The Relevant element should feed into the Prediction element in order to make a recommendation to the user.

3) Rejected

This element contains all the declined ranks. This means that the user's feedback constitutes rejection. Consequently, this element will help to prevent the receipt of synonyms of rejection in the future.

4) Prediction

The Prediction component has the ability to predict/produce contextual information. The context here is user preferences. The Prediction element has the ability to make predictions based on the contextual information, which it receives from three components: Profile Detector, Detector and Relevant. For example, the system is able to recommend or suggest some providers for the user.

5) Similarity

This component is responsible for preventing any similar context that has previously been rejected. Subsequently, the system is able to identify user intention. For example, if a user has rejected a paper called "Intelligent System", in the future, the system will prevent any paper related to that subject. Fig 4 shows the components of the Generic system.

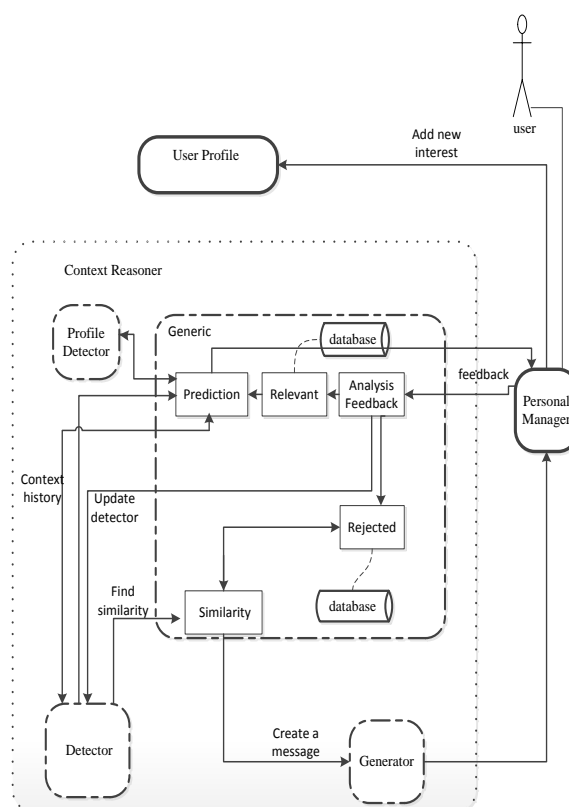


Figure 4: Generic

3.3. User Profile

The User Profile contains information about the user and consists of four elements: Personal or Static Information, Preferences, User Database and Profile Manager, as shown in Fig 5.

The main aim of the Profile Manager is to provide the Context-Aware Engine with the user's Personal or Static Information and Preferences. The user has the ability to access his or her profile to create, add and delete information through the Personal Manager.

Personal or Static Information includes standard information, such as name, address, gender, marriage status and e-mail address. In addition, the user can provide information relating to his or her various interests, noted here as preferences. It is expected that the user's preferences will change over time and that the user will be able to modify them directly. As an example, Jenny is a PLE user whose preferences state that she is only interested in being told about new family photographs posted on Facebook and Twitter. The PLE system will therefore not alert Jenny to any other activity from these independent service providers via the Context-Aware Engine.

User preferences have different lifetimes associated with them. For example, a user may be interested in a preference for a short time, which will then naturally time out and be removed; students can be given an essay, this is an example of a short-term preference. In contrast, the user may have a lifelong interest in a certain subject and, while the interest might evolve, it will remain in the profile until explicitly deleted by the user.

In addition, User Preferences accept suggestions about changing preferences from the Generic function within the Context-Aware Engine. User preferences are affected by the user's reaction to gathered information; for example when a user deletes information, the Generic part can alert or change the user preferences to create a dynamic phase that will provide better results in the future.

The user must also be able to define policy rules regarding the use of his or her profile by a third party. This gives the user the ability to decide when, with whom and what information they are willing to share.

The User Database comprises resources to which the user wishes to have long-term access. These are stored in the user database and are organised by the user.

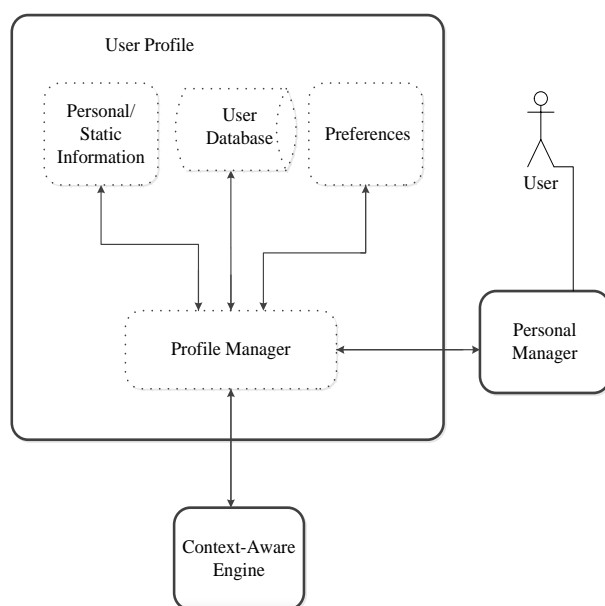


Figure 5: User Profile

4. Bayesian Network

Artificial intelligence (AI) refers to any system with the capacity to reason. AI has the ability to draw conclusions based on information gathered from

different sensors. There are various types of artificial intelligence techniques, including fuzzy logic, neural networks and Bayesian networks (BNs). The features of AI were summarised in [16], and include flexibility and adaptability and the ability to learn, reason and be dynamic.

A BN has been selected in this research because it is able to handle incomplete data and it can model the causal relationship between variables. The basic idea of BNs is to simulate reasoning in human beings; allowing person to make decisions based on information that is gathered from different sensors. Probability theory works when there are multiple sensors. BNs are suitable tools for presenting different sensors with uncertain information and connecting them in one system in addition to being able to present the level of uncertainty. BNs have been used in the artificial intelligence field, and are also known by other names, such as Bayesian belief networks, belief networks and causal networks. Korb and Nicholson (2004) provided a good introduction to BNs [17].

4.1. Advantages of Bayesian networks

BNs have a number of features that make them common and easy to use. These advantages are introduced in [17, 18], and are summarised below:

- The correctness of any inferences can be guaranteed.
- Evidence for and against a hypothesis can be combined with rules via affirming and denying weights.
- The probability of a hypothesis can be updated via more than one set of evidence.
- BNs offer some advantages for data analysis. For example, the models allow situations to be handled even when some data is missing.
- BNs are more likely to be used when reliable statistical data has been gathered.
- BNs are the best technique to deal with random variables or inaccurate knowledge.
- BNs work very well with forecasting when statistical data is available.
- BNs can explain how they arrive at a particular solution.
- The predictions for the future are based on past history.

4.2. Software tools

A number of software tools for BNs have recently appeared such as GeNIe and Simi. In this work, GeNIe software has been chosen to implement a BN model because it can provide a graphical user interface (GUI) and it can be run under Windows and Linux [19]. The model which is presented in this work has been created using GeNIe software.

GeNIe is one of the software programs for BNs which is free of charge for users; and has been used for research and industry. GeNIe was developed by the Decision System Laboratory at the University of Pittsburgh.

The users of GeNIe need to be familiar with the basic idea of BNs. [20] However, these users do not have to be experts in complex tasks, such as inference; and there is no need for users to understand how the inference technique works. The majority of GeNIe's users use it as a research tool.

4.3. Scenario

The following scenario is used to illustrate how the Generci part of PLE architecture operates and to show how BN can be used to predict the user preference. This scenario focuses on the use of PLE to support formal and informal learning in which is assumed that Jenny is a first-year PhD student at DMU. Jenny's research title is "Providing Authentication for Vehicle Ad-hoc Networks". Like any first-year researcher, Jenny wants to focus on basic and less advanced information (namely, general aspects) in the early stages of her research; the priorities in her user preferences requirements include:

- E-books, book titles from DMU library
- Blogs, Wikipedia, Google, Baidu
- Surveys in ACM, IEEE, ScienceDirect
- Courses in wireless networks

The following are excluded:

- iGoogle, YouTube, Slideshare and Haokanbu

Whenever a match of the requested resources is discovered, the Acquisition will gather the context information and forward it to the Context Reasoner. The Context Reasoner will process the context information with the context history and user preferences to filter suitable resources. The Generator will post details of these resources via web links (URLs) posted on the Personal Manager. This aggregation of resources will help Jenny to identify which of the suggested resources will be the most useful and also to discard irrelevant resources; the generic part of the Reasoning is thereby enabled to

refine preferences for these topics automatically in Jenny's profile, which will provide better results in the future. In addition, Jenny will be able to share any of her findings through her profile, enabling other users to benefit from them. This method should help researchers save time, effort and resources during the course of their study.

4.4. Modelling BN

In this research, BNs have been used to validate a prediction component which is able to predict users' preferences, such as providers and keywords. The objective is to work on predicting the provider. In order to create a structure using a BN, it is necessary to have a network which consists of nodes, and each node represents a variable. The variables must be identified and connected based on the causal relationship; and the variables have to be relevant to each other.

The key to achieving the best BN structure is to have good variable ordering; and it is important only to concentrate on the relevant variable. The target variable here is the Provider; it is not possible to measure the provider directly. However, it can be measured if the other related variables are known.

There are some factors which affect students to select their provider. These factors must be relevant information to the main target. These factors include: the supervisor's advice; type of material; membership and subject area. In addition, the membership variable is affected by two variables: free cost and high standard. Furthermore, the type of material is affected by three variables: the year of study, general information and details. Table 1 outlines all of the variables and their states.

Table 1. Variables with states

Variable	State
Provider	Ethos, library, IEEE, PubMed
Supervisor's advice	Ethos, library, IEEE, PubMed
Type of material	Thesis, book, conference
Year of study	First, second, third
General information	Yes, no
Details	Yes, no
Membership	Ethos, library, PubMed
High standard	Yes, no
Free cost	Yes, no
Subject area	Computer science, medical

A BN model has been proposed to represent and automatically detect providers for students. Fig 6 shows a BN model for the provider.

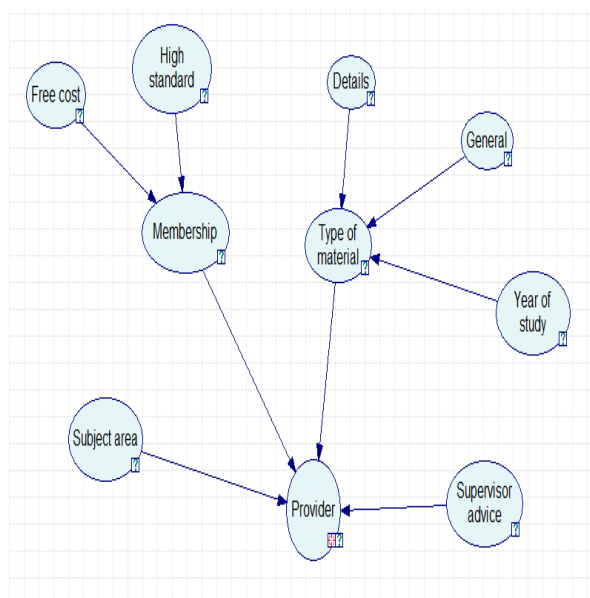


Figure 6: BN model

For each variable, a condition probability table (CPT) needs to be specified. In order to produce a CPT for each variable, statistical data is needed. This step is called “parameterising”. It is possible to elicit parameters from human experts. However there are no real parameters, and therefore those used in this research have been estimated. The strengths of the relationship or dependence between two variables are given by the CPT.

We have to add evidence to the BN. Evidence means that information has been provided about certain variables. The figure shows the BN model when new evidence is introduced.

An example would be if the current evidence for each variable from the user is comprised of a student’s subject area, which is computer science. The student has been advised by their supervisor to use IEEE; and he is in his second year of study. The student is no longer interested in general information, for instance he is looking for more in depth knowledge; and therefore he looks for a free cost and high-standard provider.

Once a BN is constructed, it can be used to make an inference about any variable. Inference refers to the

task of calculating the probabilities of each state in some variable, when the values of the other variables are known. Exact inference in polytree has been applied in this model since this is just one path between two variables, and the model is not a complex one. [17]

A result is produced after applying the inference. The probabilities of the providers’ states on demand are as follows: the library is 0.47, the Ethos is 0.3825, the IEEE is 0.1475 and PubMed is 0.

4.5. Dynamic Bayesian Networks (DBN)

BNs use one point at a time. This can be a limitation in static BNs. According to [18] the best way to solve this issue is to use a DBN that has some advantages. DBNs have the ability to deal with many inputs and outputs and it uses past history in order to predict the future. In most artificial intelligence systems, it is important to look for time-series. DBNs are useful models for any system which evolves over time, and it is able to predict future behaviour. DBNs comprise a single BN at different times, not only at one particular point in time. In DBNs, any system should depend on the previous point. This means that a system at a time (t) will only depend on its previous point (t-1)[20]. DBNs provide an explicit temporal dimension and are described as being a discrete-time mode.

With DBNs, previously received evidence is not directly used during DBN inference. However, they take the evidence available at the current time. The arc between the variables at the same time slice is called an “intra-slice arc”; whereas the arc between the variables in different time slices is called a “temporal arc”.

The advantage of this model is that it prevents users from getting many providers which they do not need. As a result, all the present providers are very useful for the users. The DBN model should support users with their favourite providers.

5. Conclusion

This paper presented a brief review of VLEs and their perceived limitations. PLEs are considered to be an evolved form of VLE, providing a student-centric approach to learning and supporting lifelong learning. It is argued that a PLE has a wider scope in that it can be used to provide the platform with a diverse range of user activities, including formal learning support. A PLE architecture driven by a Context-Aware Engine is described. The Context-Aware Engine has the functionality to acquire virtual sensor context information and to filter this using

preference defined in the user profile. The result is a set of suggested resources of current interest to the user, of which he or she is notified at PLE login through a set of structured links on the user's home page. It is argued that context-aware PLE can support the identification and distribution of relevant resources, relating to any activity defined by the user in their PLE profile automatically, and with little effort on the part of the user.

In addition, this paper also proposes a BN model; and the presented model aims at helping students to get a suitable provider. This model could be used to determine the priority provider to the students. The steps for using BN are: firstly, that the variables have been identified. Secondly, that the strength of the relationship between the variables has been shown by CPT. Finally, that the evidence has been specified and the result is shown.

6. Acknowledgment

My thanks go to all staff and students in Software Technology Research Laboratory at De Montfort University, UK. The research has been funded by the Majmaah University in Saudi Arabia.

7. References

- [1] Y. Y. C. Charles, "Virtual Learning Environment (VLE): A Web-Based Collaborative Learning System," 1998, pp. 480-480.
- [2] D. Gillet, E.L. Law, and A. Chatterjee, "Personal learning environments in a global higher engineering education Web 2.0 realm," in *Education Engineering (EDUCON)*, 2010 IEEE, 2010, pp. 897-906.
- [3] G. Attwell, "Personal Learning Environments-the future of eLearning?," *eLearning papers*, vol. 2, pp. 1-7, 2007.
- [4] F. Wang, X. Li, C. Zhao, and C. Xu, "Construct Personal Learning Environment Based on Web2.0," in *Management and Service Science*, 2009. MASS '09. International Conference on, 2009, pp. 1-4.
- [5] M. van Harmelen, "Personal Learning Environments," in *Advanced Learning Technologies*, 2006. Sixth International Conference on, 2006, pp. 815-816.
- [6] S. Wilson, O. Liber, M. Johnson, P. Beauvoir, and P. Sharples, C. Milligan, "Personal Learning Environments: Challenging the dominant design of educational systems," *Interactive Learning Environments*, vol. 16, pp. 1-2, 2008.
- [7] T. Valtonen, S. Hacklin, P. Dillon, m. Vesisenaho, J. Kukkonen, and A. Hietanen, "Perspectives on personal learning environments held by vocational students," *Computers & Education*, vol. 58, pp. 732-739, 2012.
- [8] H. Zhao, L. Yang, and Y. Wang, "The personal learning environment (PLE) based on web2.0," in *Web Society (SWS)*, 2010 IEEE 2nd Symposium on, 2010, pp. 22-25.
- [9] O. Liber, "Colloquia—a conversation manager," *Campus-Wide Information Systems*, vol. 17, pp. 56-62, 2000.
- [10] M. Baldauf, S. Dustdar, and F. Rosenberg, "A survey on context-aware systems," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 2, pp. 263-277, 2007.
- [11] J. Hong, E. Suh, and S. Kim, "Context-aware systems: A literature review and classification," *Expert Systems with Applications*, vol. 36, pp. 8509-8522, 2009.
- [12] A. K. Dey, "Understanding and using context," *Personal and ubiquitous computing*, vol. 5, pp. 4-7, 2001.
- [13] B. N. Schilit, N. Adams, and R. Want, "Context-aware computing applications," 1994, pp. 85-90.
- [14] J. Indulska and P. Sutton, "Location management in pervasive systems," 2003, pp. 143-151.
- [15] A. Schmidt and K. Van Laerhoven, "How to build smart appliances?," *Personal Communications, IEEE*, vol. 8, pp. 66-71, 2001.
- [16] Krishnakumar, K. (2003). *Intelligent Systems for Aerospace Engineering-An Overview*, DTIC Document.
- [17] Korb, K. B. and A. E. Nicholson (2003). *Bayesian artificial intelligence*, Chapman & Hall/CRC.
- [18] Si, H., Y. Kawahara, et al. (2005). "A stochastic approach for creating context-aware services based on context histories in smart home." *COGNITIVE SCIENCE RESEARCH PAPER-UNIVERSITY OF SUSSEX CSRP* 577: 37.
- [19] Pourret, O., P. Naïm, et al. (2008). *Bayesian networks: a practical guide to applications*, Wiley.
- [20] Mihajlovic, V. and M. Petkovic (2001). "Dynamic bayesian networks: A state of the art."