View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by UCT Computer Science Research Document Archive

DEVELOPING A BASIS FOR KNOWLEDGE MANAGEMENT: A BAYESIAN NETWORK APPROACH

Technical Report Number CS03-14-00

Henry Brown
Dept. of Computer Science
University Avenue, Rondebosch
Cape Town 8000
+27 21 650 3799
hbrown@cs.uct.ac.za

Phumelelakahle Kunene
Dept. of Computer Science
University Avenue, Rondebosch
Cape Town 8000
+27 21 650 3799
pkunene@cs.uct.ac.za

Colin Rouse
Dept. of Computer Science
University Avenue, Rondebosch
Cape Town 8000
+27 21 650 3799
crouse@cs.uct.ac.za

Kurt April
Graduate School of Business
Rondebosch
Cape Town 8000
+27 21 406 1411
kurt@gsb.uct.ac.za

Sonia Berman
Dept. of Computer Science
University Avenue, Rondebosch
Cape Town 8000
+27 21 650 2663
sonia@cs.uct.ac.za

Anet Potgieter
Dept. of Computer Science
University Avenue, Rondebosch
Cape Town 8000
+27 21 650 2663
anet@cs.uct.ac.za

ABSTRACT

Knowledge Management (KM) is an evolving field that attempts to maximise and sustain the competitive advantage of a company through leveraging its knowledge resources. KM practises are often built on a foundation of knowledge transfer and knowledge sharing. Recently there has been an increase on the reliance of automated tools to perform these functions. Typical components of these tools include: querying large datasets, user profiling, user interfaces and recommender systems. Traditionally, these components have been implemented using different technologies. This paper describes an approach to building these components using a flexible architecture based on Bayesian Network technology. Finally the paper considers some of the advantages to adopting the latter approach.

Categories and Subject Descriptors
[Knowledge Management]: Distributed Knowledge, Information Architecture, Knowledge Retrieval

General Terms
Management, Theory

Keywords: Bayesian Networks, Knowledge Management, Knowledge transfer, Knowledge sharing

1 Introduction

Knowledge is a valuable resource in most modern day companies. A growing number of companies are realising that leveraging their organisational knowledge is a key component in achieving sustainable competitive advantage. However, the key to deriving sustainable competitive advantage from the organisational knowledge lies in the company's ability to leverage this knowledge. This challenging task requires that companies engage in a number of activities (such as knowledge transfer and sharing) which collectively fall under the banner of Knowledge Management (KM).

KM is a relatively young field which attempts to make sense of the work companies perform as they attempt to leverage their knowledge resources. One key area of research within the domain of KM centres on being able to decide what processes need to be in place if a company is to successfully pursue a KM project. One of the few factors that KM practitioners agree on, is that a company should at the very least support structures that allow for knowledge transfer and sharing. This paper attempts to further this discussion by asking "How does a company build such a basis to enable the transfer and sharing of knowledge?" In particular, this paper focuses on automated tools aimed at addressing these issues. The paper proposes the use of Bayesian Networks (BN) as a flexible tool which can be used as a basis in order to implement many of the automated features commonly associated with knowledge sharing and transfer capabilities.

In order to lay a common understanding on which to base this discussion it is necessary to firstly define what is meant (in the context of this paper) by many of the terms used in the remainder of this discussion. Furthermore, it is necessary to briefly look at the type of tools generally associated with knowledge sharing and transfer tools.

2 Common Framework

2.1 Definitions

Bayesian Network

A BN is a probabilistic graph model. It can be defined as a pair, (G, p), where G = (V; E) is a directed acyclic graph (DAG). Here, V is the node set which represents variables in the problem domain and E is the edge set which denotes probabilistic relationships among the variables. *p* represents the set of conditional probability distributions associated with the BN.

Knowledge Sharing and Knowledge Transfer

Knowledge sharing refers to the degree to which people in an organisation are able to share their knowledge gained as a result of their experience, expertise, culture etc., with peers. Knowledge transfer, on the other hand, refers to the process of extracting knowledge from the materials associated with a particular task in the environment. These materials are typically in the form of books, manuals, specifications etc., and do not involve any direct communication between peers.

2.2 What constitutes a knowledge sharing/transfer environment

It is clear that there are a vast number of different systems implemented with the broad goal of helping the organisation to achieve knowledge sharing and transfer. However, for the purposes of this discussion the following elements are considered the key components of any such system. This does not imply that all these components are necessary to have a knowledge sharing/transfer environment but rather that it is possible to identify many of these components within knowledge sharing/transfer systems.

These sub-components include:

- User Profiling
  Many automated systems employ some type of user profiling system in order to try and better meet the knowledge needs of the particular user.
- Recommender Systems
  These systems use the experiences of other people (which are deemed to be similar in certain respects, such as job description) to recommend a particular course of action or knowledge item that may be of use to a user. This type of system tends to blur the distinction between knowledge transfer and knowledge sharing and can be seen as being an automated means of knowledge sharing.
- Interfaces
  More and more research has recently focussed on the interface between the automated knowledge system and the user. This can often have a dramatic impact on the amount of knowledge transfer (and thus subsequent knowledge sharing) that can occur.
- Querying large datasets
  A fundamental necessity in performing automated knowledge sharing/transfer is the ability to effectively query the typically large volume of data/information stored by the organisation.

The next section of this paper looks at ways in which many of these tasks have been addressed in the past.

The outline for the paper is as follows: section 3 provides a brief literature review of the various knowledge management components. Section 4 is a discussion on the possible roles Bayesian Networks may play in implementing these components. Section 5 continues by illustrating the advantages of using Bayesian Networks to develop components.

The final section of the paper provides the conclusions we reached in our investigation.

3 Literature Review

The literature review provides a brief explanation of the theory behind some of the major components of the tools used in knowledge management. In addition, this section explains these component's services.

3.1 User Profiling

User profiling is the process of gathering information and making inferences, based on this information, concerning user characteristics (Kobsa, 1995; Bohté, Langdon and Poutré, 2000). This information is embodied in the user profile (Kobsa, 1995), typically defined in some formal description technique (Kobsa, 1995; Mobasher, Srivastava and Cooley, 2000). The user profile representation is dependant on the agent who has interest in this information. For example, the user profile may be represented as some data-structure for a system component, or as a report for a human domain expert.

3.2 Recommender systems

Recommender systems aid a user in selecting an option from a selection of alternatives, using the recommendations of other users (collaborative filtering) or knowledge stored in a knowledge base (content filtering) (Olsson, 2003). Typically the three phases of the recommender process involve (Olsson, 2003):

1. Gathering all recommendations.
2. Applying these recommendations to a learning algorithm to build a prediction model.
3. Making predictions using this devised model (model-based prediction) using, for example, a clustering or nearest neighbour algorithm.

3.3 Interfaces

An incremental, or intelligent, interface will change according to some function. This function is dependent on time or the progress of the user.
The system will reveal different functionality as the function changes (Wærn, 1997). The interface operates by hiding functionality until the user actions indicate that the skill of the user is sufficient to enable previously hidden functionality. This allows the interface to change according to how it is used. It will become simpler for novice users, as complex functionality will be hidden, and more complex when expert user actions are received.

3.4 Querying Large datasets

The Boolean model of a document is specified using a set of index terms whose weights are binary. If the weight of a term is one (zero), the term is present (absent) in the document (Sethi, 2002; Choi, Kim and Raghavan, 2001; Salton, Fox and Voorhees, 1985). To retrieve documents from an IR system using Boolean modelling, the query is constructed of index terms linked by three connectives: "not", "and", "or". Only those documents that are deemed 'true' for the query (Choi, Kim and Raghavan, 2001) are retrieved. For example, consider four documents D1, D2, D3, and D4. The index term K1 is present in all four documents. K2 is true only for D1 and D2. K3 occurs in D1, D2, and D4. K4 is present only in D1. If the query Q = (K1 AND K2) OR (K3 AND (NOT K4)) is input to the system then the Boolean search will retrieve all documents indexed by K1 and K2, as well as all documents indexed by K3 which are not indexed by K4. Thus, the result is the set {D1, D2, D3} which satisfies the query and each document in it is 'true' for the query (Salton, Fox and Voorhees, 1985).

Simplicity and search speed are the main advantages of the Boolean model (Choi, Kim and Raghavan, 2001). One of the primary disadvantages of this model is that documents are considered to be either relevant (true) or non-relevant (false) and there is no notion of a partial match or ranking. Even using the coordination level is an extremely primitive method of ranking documents. Furthermore, the Boolean method relies on the user to precisely and accurately formulate the query in order to get good results.

This vector model attempts to represent both documents and queries as vectors. The keywords used to describe the contents of documents or queries are assumed to correspond to the various elements of the vectors. Thus, if the indexing vocabulary consists of n distinct keywords, each document is an n-element vector in which the i^{th} element represents the importance of the i^{th} keyword to the document concerned (Wong & Raghaven, 1984). When a query is presented, the system formulates the query vector and matches against the documents based on a chosen method of determining similarity between vectors (Wong & Raghaven, 1984). For example, similarity between the query and a document may be defined as the scalar product of the corresponding vectors and the documents could be ranked in the decreasing order of this measure.

4 Discussion

The following discussion investigates the use of BNs to provide the type of services discussed in the literature review.

4.1 User Profiling

Bayesian networks are ideally suited for user profiling. Attributes of interest may be modeled as nodes in the Bayesian network. Probabilities of desired attributes may then be queried given the values of other attributes (some of which may not be set).

Consider a simple example of fraud detection. We would like to classify or profile certain users according to a set of attributes to ascertain to what probability they may or may not be fraudulent. Expectedly, this network comprises of two class nodes (fraudulent and not-fraudulent) and set of attribute nodes that relate to these classes.

Let us define 3 attribute nodes that will determine whether an individual is fraudulent or not:

1. Employed: this attribute node indicates whether an individual user is employed or not. It has the discrete values *true* or *false*.
2. Salary: this attribute node illustrates whether an individual earns a particular category of salary per month. It has the discrete values of *high*, *medium* and *low*.
3. Age: this attribute node illustrates what age-category an individual falls under. This has the discrete values of “>50”, “30><50”, “<30”.

Typically, we would like to make queries such as what is the probability that an individual is fraudulent given that they earn a medium salary, are unemployed and younger than 30 years of age. The Bayesian network logic is able to process such a query giving the likelihood or probability that that individual is fraudulent given those attributes.

Another key significant advantage of the Bayesian approach is that not all attribute values have to be defined when defining a query. The Bayesian logic sufficiently caters for unknown values in its computations – a significant advantage over other probabilistic models that require all attribute values to be set explicitly. Thus, the Bayesian approach provides more flexibility than other approaches since a larger “case-base” (consisting of cases that may contain null attribute values) may be processed.

4.2 Recommender systems

As aforementioned in the background section of this paper, recommender systems use various algorithms to predict what items, given user characteristics, may interest a particular user. We propose using a Bayesian Network approach as the prediction logic necessary for clustering and prediction.

Consider a typical recommender case where we would like to assess what elements or attributes hold a particular user's interests, so that elements of interest may be recommended. These interests may be set as nodes that may be queried in the BN. In addition, we may be interested in clustering users according to similar interests. In this case, we would define the clusters into which we would like to partition users. The Bayesian network would then partition the users into these predefined classes according to the similarities in attributes between them.

The developed model would be able to support queries such as finding the interests or preferences of users in a particular user-class. The recommender may use the result of this query to recommend similar elements of interest to a particular user (given the interest of users in that user's class).

4.3 Interfaces

Harrington (Harrington, 1996) was one of very few to incorporate Bayesian Networks into an adaptable user interface. This simple command-line system changes the structure of the functionality to reflect the best interface according to the received user actions[1].

The system consisted of three types of nodes: information, learning and uncertainty nodes. The information node would contain the usage of a particular type of data or command. Learning nodes used the information contained in the information nodes to make decisions for the user and the adaptations. The uncertainty node held the value of how sure the learning node would be based on past experience.

4.4 Querying large datasets

One approach to using BNs to allow the querying of large datasets employs the use of two BNs: the document network and the query network. The former network contains nodes corresponding to documents (abstract), texts (specific text content of a document), text concepts (extracting from the text by various techniques like manually assigned terms, automatic key word extraction, etc.). This network is built once for a given document collection. The relationships between text nodes and text concept nodes are set by various indexing schemes; therefore, it is possible to associate a weight with each of these links (Turtle and Croft, 1990; Croft and Turtle, 1989).

The query network, on the other hand, contains nodes representing query concepts, queries as well as one node to represent the user's information need. Links are present between queries and the information need, as well as between query concept nodes and queries. Their approach also indicates how the networks are able to deal with uncertain evidence (Turtle and Croft, 1990; Croft and Turtle, 1989).

5 Advantages of BN in Knowledge Discovery (KD) and Information Retrieval (IR)

This section of the paper completes our investigation by describing the advantages of using BNs to implement KD and IR.

5.1 Decision theory

As Bayesian networks are models of the problem domain probability distribution, they can be used to compare the predictive distribution on the outcomes of possible actions. This means that it is possible to use decision theory for risk analysis, and choose in each situation the action, which maximizes the expected utility. It can be shown that in a very natural sense, this is the optimal procedure for making decisions (Myllymäki, Silander, Tirri and Uronen, 2001).

2) BN is a consistent, theoretically solid mechanism for processing uncertainty

5.2 Information

Probability theory provides a consistent calculus for uncertain inference. This means that the output of the system is always unambiguous. Given the input, all the alternative mechanisms for computing the output with the help of a BN model produce exactly the same answer (Myllymäki, Silander, Tirri and Uronen, 2001).

5.3 Scalability and maintainability advantages

BN models have been found to be very robust in the sense that small alterations in the model do not affect the performance of the system dramatically. This means that maintaining and updating existing models is easy since the functioning of the system changes smoothly as the model is being modified. For sales and marketing systems this is a crucial characteristic, as these systems need to be able to follow market changes rapidly without complex and time consuming re-modelling (Myllymäki, Silander, Tirri and Uronen, 2001).

5.4 Flexible applicability

BNs model the problem domain as a whole by constructing a joint probability distribution over different combinations of the domain variables. This

[1] This does not occur in real-time

means that the same BN model can be used for solving both discriminative tasks (classification) and regression problems (configuration problems and prediction). Besides predictive purposes, BNs can also be used for explorative data mining tasks by examining the conditional distributions, dependencies and correlations found by the modelling process (Myllymäki, Silander, Tirri and Uronen, 2001).

5.5 A theoretical framework for handling expert knowledge

In Bayesian modelling, expert domain knowledge can be coded as prior distributions, prior meaning that the probability distributions are defined before and independently of processing any possible sample data. This allows for combining expert knowledge with statistical data in a very practical way. Using suitable prior distributions, the priors can be given a semantically clear explanation in terms of the data (expert knowledge can be interpreted as an unseen data-set of the same form as the training data). This means that the experts will also be able to give an estimate of the weight or importance of their prior knowledge, compared to the training data available (Myllymäki, Silander, Tirri and Uronen, 2001).

5.6 A clear semantic interpretation of the model parameters

Unlike Neural Network models, which usually appear to the user as a black box, all the parameters in BNs have an understandable semantic interpretation. It is for this reason that BNs can be constructed directly by using domain expert knowledge, without a time-consuming learning process. On the other hand, if machine learning techniques are used (with or without expert knowledge) for constructing BN models from sample data, the resulting model can be analyzed and explained in terms that are understandable to domain experts (Myllymäki, Silander, Tirri and Uronen, 2001).

5.7 Different variable types

Probabilistic models can handle several different type variables at the same time, whereas many alternative model technologies have been designed for some single specific type of variables (continuous, discrete etc.). For these alternatives, working with several variable types requires some kind of transformation operations, which in some cases may be the cause for unexpected results. From the probabilistic point of view, all the basic entities are distributions, which mean that all the different variable types fall elegantly in the same unifying framework (Myllymäki, Silander, Tirri and Uronen, 2001).

5.8 8) A theoretical framework for handling missing data

In the BN framework, missing data is marginalized out by integrating over all the possibilities of the missing values. Although the advantages of probabilistic modelling have been largely recognized and accepted, the probabilistic approach has often been neglected in the past as the theoretically correct, but computationally infeasible methodology. Perhaps the most common argument against using probabilistic models has been that the number of parameters needed for defining the models is too high. Nevertheless, the theoretical framework of BN modelling suggests that it is possible to construct quite successful probabilistic models using only a moderate number of parameters. In addition, BNs appear to be rather insensitive to the accuracy of the parameters, so determining good parameter values is in many application areas quite feasible. For these reasons, there has during the last few years been a rapid growth in the number of BN models being developed. BN models are currently being applied in, for example, building intelligent agents and adaptive user interfaces (Microsoft, NASA), process control (NASA, General Electric, Lockheed), fault diagnosis (Hewlett Packard, Intel, American Airlines), pattern recognition and data mining (NASA), and medical diagnosis (BiopSys, Microsoft) (Myllymäki, Silander, Tirri and Uronen, 2001).

6 Conclusions

The automated tools used in addressing issues such as knowledge transfer and sharing, within KM, currently make use of a variety of technologies. The BN technology proposed in this paper has been shown to be a viable alternative to these technologies. The numerous advantages mentioned above (section 5) indicate why this approach can be beneficial.

This conclusion is also supported by our project which demonstrates the utility of BNs in implementing such tools as a user profiler and the ability to query large datasets.

7 References

Bohté S. M., Langdon, William B., Poutré, Han La. “On Current Technology for Information Filtering and User Profiling in Agent-Based Systems, Part I: A Perspective” (CWI, Centre for Mathematics and Computer Science P.O. Box 94079, 1090 GB Amsterdam, The Netherlands, January 2000)

Chen, M., Han, J., & Yu, P. (1996) Data Mining: An overview from Database perspective.

Choi, Minkoo Kim, Vijay V. Raghavan. (2001) Lecture Notes in Computer Science.

Croft, B.W and Turtle, H. A retrieval model for incorporating hypertext links. In Hypertext '89 Proceedings, pp 213 – 224, ACM, New York, 1989.

Goebel, M. Gruenwald, L. (1999) "A Survey of Data Mining and Knowledge Discovery Software Tools." *SIGKDD Explorations ACM SIGKDD*, June 1999, Vol. 1, Issue 1, pg. 21)

Han, J, and Cercone, N. (2000) RuleViz: A Model for Visualizing Knowledge Discovery Process.

Harrington, R. (1996), *Utilizing Bayesian Techniques for User Interface Intelligence*. Department of Defence, U.S Government. PhD Thesis (unpublished).
Miller, C.A. (2000), *Getting Intelligence into an Intelligent User Interface: Intent-Based Policy for Automated Resource Allocation*, Minneapolis: Honeywell Technologies.

Neri, F and Saitta, L. Machine learning for information extraction. Information Extraction: A Multidisciplinary Approach to an Emergine Information Technology, Vol. 1299:171-191, June 1997

Olsson, T. "Bootstrapping and Decentralizing Recommender Systems" (Computing Science Division, Department of Information Technology, Uppsala University, Uppsala Sweden, June 2003)

Salton, G., Fox, E. A., and Voorhees. Advanced Feedback Methods in Information Retrieval. J. of the American Society for Information Science, 36(3): pp. 200-210, 1985.

Sethi, I.K (Instructor). (2002) Information Retrieval Course Notes.
URL:
http://www.cse.secs.oakland.edu/isethi/IR/Coursenotes.html)

Quinlan, J.R. Induction of decision trees. Machine Learning, 1: 81-106, 1986.

Quinlan, J.R. C4.5: Programs for Machine Learning. Morgan Kaufmann, 1993.

Turtle, H. and Croft, B.W. Inference networks for document retrieval. In Proceedings SIGIR'90, pp 1-24 (September 1990). To appear in ACM Transactions on Information Systems, 1991.

Wærn, A. (1997), *What is an Intelligent Interface?*, Intelligent Interfaces, URL: http://www.sics.se/ (accessed on 25 April 2003).

Wong & Raghaven, (1984) The vector space model. Department of Computer Science, University of Regina, Canada.

Heinonen, O and Mannila, H. (1996) Attribute-oriented induction and conceptual clustering.

Kobsa, A. (1995) Supporting User Interfaces for All Through User Modeling, (Proceedings HCI International '95, Yokohama, Japan, 1995, pp. 155-157)

Mobasher, B. Cooley, R. Srivastava, J. "Automatic Personalisation Based on Web Mining". Communications of the ACM, August 2000, Vol. 43, No. 8