# Application of Bayesian Networks to Shopping Assistance

Yang Xiang, Chenwen Ye, and Deborah Ann Stacey

University of Guelph, CANADA

**Abstract.** We develop an on-line shopping assistant that can help a e-shopper to select a product from many on-line shops based on the personal preference. The shopping assistant is developed based on value networks which extend Bayesian networks with user preference. On-line shopping of bicycles is used as the application domain.

## 1 Introduction

We investigate consumer shopping assistance by developing an intelligent shopping agent. To assist the consumer effectively, the agent must be able to evaluate a given product against the preferences of the consumer. We decompose the agent's task into two subtasks: to evaluate the product in terms of a set of consumer-independent attributes and to judge how well the product matches the consumer preferences. Both the evaluation of a product in some sense of quality and the judgment how well it suits a given consumer are intrinsically uncertain. We therefore treat the core computation as a matter of uncertain reasoning [Grass00]. We model the agent's uncertain knowledge on a particular type of product with a Bayesian network. We then extend the model into a value network by adding the consumer preferences. The problem domain used in this research is on-line bicycle shopping.

There are a few hundred on-line bicycle shops and manual shopping can be very time-consuming. A shopping assistant will be of practical usage. Different on-line shops use different terminologies to describe bicycles in their web pages. A set of standard features is needed to differentiate and compare different bicycles. Web pages of on-line bicycle shops are organized according to different layouts. This poses challenges for extracting key features of bicycles across different on-line shops. Retrieval of a web page anywhere in the Internet takes at least seconds and it is impractical to search several hundreds of on-line shops after a user query is submitted. These are some basic constraints to our development.

The architecture consists of several components: The *Information Source Base* contains website addresses of on-line bicycle shops and bicycle knowledge base. *Local Database* stores bicycle product information retrieved from Web. A *GUI* interacts with users and communicates with retrieval agents. The *Remote Retrieval Agent* is responsible for retrieving raw product information from a web site and returns its results. It uses page wrappers to retrieve bicycle information from shop web pages and converts into a structured format. The *Local Retrieval Agent* is

responsible for obtaining information requested by the decision module from the local database. It maps bicycle feature information into the format consistent with the knowledge base. The *Decision Module* computes a utility value for every product according to the information returned by *local retrieval agent*..

## 2 Cost Model

The bicycle domain is modeled using Bayesian networks [Pearl88] and value networks where user preference is represented by value distributions. The domain knowledge about a bicycle is represented using three models: the Cost Model, the Performance Model, and the Usage Model. Cost Model consists of a cost variable and a value variable $V_0$ reflecting the user's preference on the bicycle purchase cost. We set the chance node space as $\{c_0, c_1, \ldots, c_{79}\}$, where every state $c_i$ is associated with certain dollar range such as $c_0$ = [\$0, \$80), $c_1$ = [\$80, \$160), and so on.

The value of the cost variable is derived from combining three pieces of information as illustrated In Fig.1. P stands for the price of the bicycle, D stands for the bicycle delivery fee, and W stands for the possible extended warranty fee for the bicycle. The values of P, D, and W are extracted directly from the bicycle web page and are summed to yield the numerical cost as indicated by the symbol "+". The numerical cost C is then discretized to instantiate the cost variable in the value network.

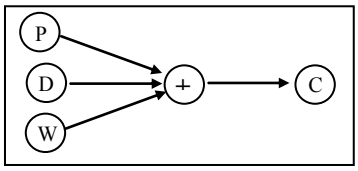


**Fig. 1.** The chance node "cost" in Cost Model

We set the value distribution at node $V_0$ by seeking the user input. As an example, suppose that the user wants to buy a bicycle in the cost range from \$200 to \$350. Each element in the space of cost variable is then compared with the cost range and a value in the range [0,1] is automatically determined. For instance, because $c_3$ = [\$240, \$320) is entirely contained in the range r = [200, 350], we set $V_0$ (cost = $c_3$) = 1.0. Because $c_2$ = [\$160, \$240) is partially overlapping with r, we set $V_0$ (cost = $c_2$) = (240-200)/80 = 0.5. Similarly, we set $V_0$ (cost = $c_4$) = (350-320)/80 = 0.375.

## 3 Performance Model

The *bicycle performance* model (Fig. 2) is concerned with both the *comfort level* and *efficiency* of the bicycle for a particular usage. *Comfort level* refers to the degree of comfort while riding the bicycle. *Efficiency* refers to the level of energy spent when riding the bicycle over some units of distance.

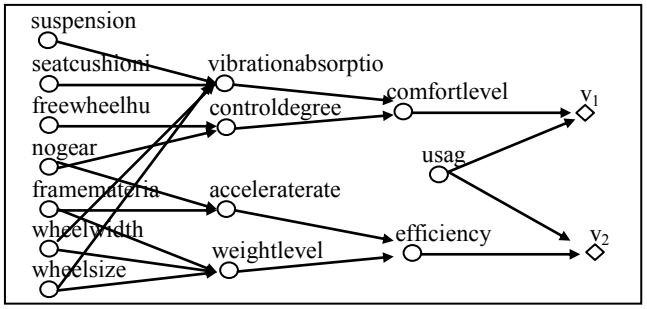


**Fig. 2.** Performance Model

The performance model contains value variables $V_1$ and $V_2$ to represent user preferences on comfort level and efficiency. $V_1$ depends on comfort level and intended usage of the bicycle. $V_2$ depends on the efficiency as well as the intended usage. The value distributions are defined by reflecting the common preference patterns when a bicycle is used. Every bicycle user prefers a more comfortable bicycle over a less comfortable one and prefers a more efficient bicycle over a less efficient one. We have thus set the values in the value distribution that correspond to extreme outcomes of comfort level and efficiency to 1 and 0. For example $V_2$(Efficiency=high, usage) = 1 for all usage outcomes and $V_2$(Efficiency=low, usage) = 0 for all usage outcomes. However, when the value of Efficiency is medium, the value distribution is determined using the following approach.

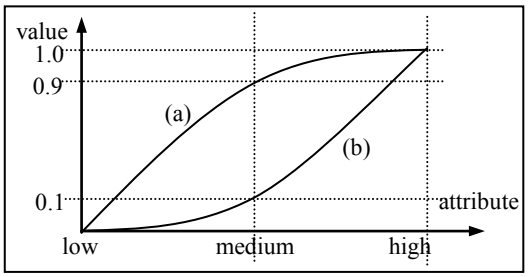


**Fig. 3.** The curve for setting value distributions in Performance Model

Fig. 3 illustrates two different preference patterns towards intermediate variable values. The user whose preference is represented by curve (a) values much highly on the intermediate variable values (medium) than the user whose preference is represented by curve (b). This is very similar to the risk seeking and risk aversion behavior according to decision theory.

## 4 Usage Model

Cost and performance models are insufficient to capture all necessary user preferences. For example, suppose that a user wants to use a bicycle for cruising. The typical bicycle type for this usage is the Comfort bicycle. However, evaluation based on the two models yields high values for mountain bicycles, because Comfort bicycles usually have lower comfort levels and poorer efficiency than Mountain bicycles. This prompted us to develop the usage model. Bicycles are classified into several types. Each type has its unique style corresponding to an intended usage. For instance, mountain bicycles are commonly used to pedal through mud, contend with roots, rocks and bumps, and ride off-road. When used for the primary purpose, the features of a bicycle can be fully exploited. We assume that the user of a bicycle has an intended primary usage of the bicycle. To avoid the evaluation problem described above, we use the usage model to address whether the type of a bicycle matches the user's intended primary usage of the bicycle.

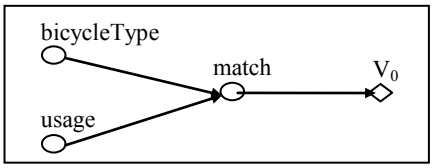


**Fig. 4.** Usage Model

The usage model is shown in Fig 4. This model has two observable chance nodes, the bicycle type and the primary intended usage of the bicycle by the user. The bicycle type can be observed from the bicycle web page. The user's primary intended usage is specified by the user.

The child variable *match* models whether a given type of bicycles matches the primary intended usage of a user. We set its conditional probability distribution subjectively. For example, we set $P(match = yes|bicycleType = mountain, usage = crusing) = 0.6$. It means that if a mountain bicycle is used for cruising, only 60% of its features can be utilized. The usage model resolved the limitation described above. Given two bicycles of the identical evaluation based on cost and performance models, the one with a better match between the primary usage and bicycle type will receive a higher overall evaluation.

## 5 Computing Overall Value

During evaluation, each value network is transformed into a Bayesian network by removing the value nodes. Probabilistic inference is then performed in the resultant Bayesian network. For the variables whose values can be obtained from the bicycle web page, these values are entered into the Bayesian network and the posterior probabilities for the parents of value nodes are obtained. We used an inference algorithm based on cluster trees [Jensen96] but other standard algorithms can be used

as well. Once the posteriors for the parents of value nodes are obtained, the value (UV) for each value node V can be computed using equation (1):

$$UV = \sum_{i=0}^{|D\alpha|} P(\alpha_{i \mid obs}) * V(\alpha = \alpha_i) \quad \textbf{(1)}$$

Where $\alpha$ represents the set of parent variables of V, each $\alpha_i$ is a configuration of $\alpha$ (with a value assigned to each variable in $\alpha$), and $D_\alpha$ is the space of $\alpha$ (the set of all such configurations). After the value for each value node is obtained, they are combined in a linear fashion to produce the overall value of the given bicycle:

$$U = \sum_i \beta_i * UV_i \quad \textbf{(2)}$$

where $UV_i$ is the value for the ith value variable and $\beta_i$ is the weight for the ith value. After each bicycle under consideration has been evaluated, a ranked list of bicycles is provided to the user for purchasing decision.

## 6 Remarks

We described the design and implementation of a bicycle shopping assistant using value networks which are decision-theoretic graphical models. Limited experiments comparing the performance of the shopping assistant and human users have also been conducted. Our experience and limited experimental results show that the approaches we take are feasible for similar types of e-shopping applications.

## References


[Grass00] Grass, J. and Zilberstein, S. A Value-Driven System for Autonomous Information Gathering. Journal of Intelligent Information Systems, Vol. 14, Kluwer Academic Publishers, pages 5-27, 2000.

[Jesen96] Jensen, F. V. An Introduction To Bayesian Networks. UCL Press, 1996.

[Pearl88] Pearl, J. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, 1988.