

# Predictive and Contextual Feature Separation for Bayesian Metanetworks

Vagan Terziyan

Industrial Ontologies Group, Agora Center, University of Jyväskylä,  
P.O. Box 35 (Agora), FIN-40014 Jyväskylä, Finland  
vagan@it.jyu.fi

**Abstract.** Bayesian Networks are proven to be a comprehensive model to describe causal relationships among domain attributes with probabilistic measure of conditional dependency. However, depending on a context, many attributes of the model might not be relevant. If a Bayesian Network has been learned across multiple contexts then all uncovered conditional dependencies are averaged over all contexts and cannot guarantee high predictive accuracy when applied to a concrete case. We are considering a context as a set of contextual attributes, which are not directly effect probability distribution of the target attributes, but they effect on “relevance” of the predictive attributes towards target attributes. In this paper we use the Bayesian Metanetwork vision to model context-sensitive feature relevance. Separating contextual and predictive features is an important task. In this paper we also consider three strategies of extracting context from relevant features, which are based on: *part\_of* context, role-based context and interface-based context.

## 1 Introduction

A *Bayesian network* is a valuable tool for reasoning about probabilistic (causal) relationships [1]. A Bayesian network for a set of attributes  $X = \{X_1, \dots, X_n\}$  is a directed acyclic graph with a network structure  $S$  that encodes a set of conditional independence assertions about attributes in  $X$ , and a set  $P$  of local probability distributions associated with each attribute [2].

An important task in learning Bayesian networks from data is model selection [3]. The models-candidates are evaluated according to measured degree to which a network structure fits the prior knowledge and data. Then the best structure is selected or several good structures are processed in model averaging. Each attribute in ordinary Bayesian network has the same status, so they are just combined in possible models-candidates to encode possible conditional dependencies however many modifications of Bayesian networks require distinguishing between attributes, e.g. as follows:

- *Target attribute*, which probability is being estimated based on set of evidence.
- *Predictive attribute*, which values being observed and which influences the probability distribution of the target attribute(s).

- *Contextual attribute*, which has not direct visible effect to target attributes but influences relevance of attributes in the predictive model. A contextual attribute can be conditionally dependent on some other contextual attribute.

Causal independence in a Bayesian network refers to the situation where multiple causes provided by predictive attributes contribute independently to a common effect on a target attribute. Context specific independence refers to such dependencies that depend on particular values of contextual attributes.

In [4], Butz exploited contextual independencies based on assumption that while a conditional independence must hold over all contexts, a contextual independence need only hold for one particular context. He shows how contextual independencies can be modeled using multiple Bayesian networks. Boutilier et al. [5] presents two algorithms to exploit context specific independence in a Bayesian network. The first one is network transformation and clustering. The other one is a form of cutset conditioning. This is done using reasoning by cases, where each case is a possible assignment to the variables in the cutset. The results of inference for all cases are combined to give the final answer to the query. Zhang [6] presents a rule-based contextual variable elimination algorithm. Contextual variable elimination represents conditional probabilities in terms of generalized rules, which capture context specific independence in variables. Geiger and Heckerman [7] present another method to exploit context specific independence. With the notion of similarity networks, context specific independencies are made explicit in the graphical structure of a Bayesian network.

Bayesian Multi-nets were first introduced in ([8]) and then studied in ([9]) as a type of classifiers. A Bayesian multi-net is composed of the prior probability distribution of the class node and a *set* of local networks, each corresponding to a value that the class node can take. A recursive Bayesian multinet was introduced by Pena et al [10] as a decision tree with component Bayesian networks at the leaves and was applied to a geographical data-clustering problem. The key idea was to decompose the learning Bayesian network into learning component networks.

In our previous work [11, 12], is the multilevel probabilistic meta-model (Bayesian Metanetwork), has been presented, which is an extension of traditional BN and modification of recursive multinets. It assumes that interoperability between component networks can be modeled by another BN. Bayesian Metanetwork is a set of BN, which are put on each other in such a way that conditional or unconditional probability distributions associated with nodes of every previous probabilistic network depend on probability distributions associated with nodes of the next network. We assume parameters (probability distributions) of a BN as random variables and allow conditional dependencies between these probabilities. Algorithms for learning Bayesian Metanetworks were discussed in [13]. In [18] we presented another view to the Bayesian Metanetwork by presenting the concept of attribute “relevance” as additional (to an attribute value probability) computational parameter of a Bayesian Network. Based on computed relevance only a specific sub-network from the whole Bayesian Network will be extracted and used for reasoning.

The rest of paper organized as follows. In Section 2 we first provide basic architecture of the Bayesian Metanetwork for managing Attribute Relevance and appropriate reasoning formalism behind the concept, summarizing [18]. Section 3

provides three major strategies of contextual features selections for Bayesian Metanetwork, which are based on: part\_of context, role-based context and interface-based context. We conclude in Section 4.

## 2 Bayesian Metanetwork for Managing Attributes' Relevance

*Relevance* is a property of an attribute as a whole, not a property of certain values of an attribute. This makes a difference between relevance and probability, because the last one has as many values as an attribute itself. Another words, when we say probability, we mean probability of the value of the attribute, when we say relevance, we mean relevance (*probability to be included to the model*) of the attribute as whole. Consider the general case of managing relevance (Fig. 1):

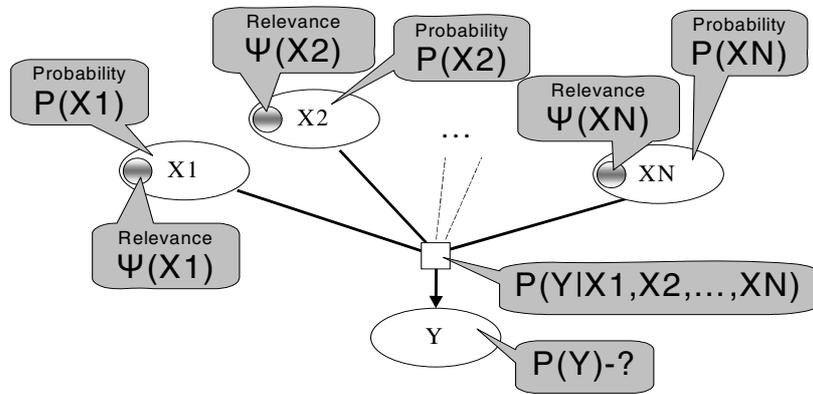


Fig. 1. General case of relevance management

In this case we have the following:

Predictive attributes: \$X\_1 \{x\_{11}, \dots, x\_{1n\_{x1}}\}; \dots; X\_N \{x\_{n1}, \dots, x\_{n\_{xn}}\}\$;

Target attribute: \$Y\$ with values \$\{y\_1, y\_2, \dots, y\_{ny}\}\$.

Probabilities: \$P(X\_1), P(X\_2), \dots, P(X\_N); P(Y|X\_1, X\_2, \dots, X\_N)\$.

Relevancies:

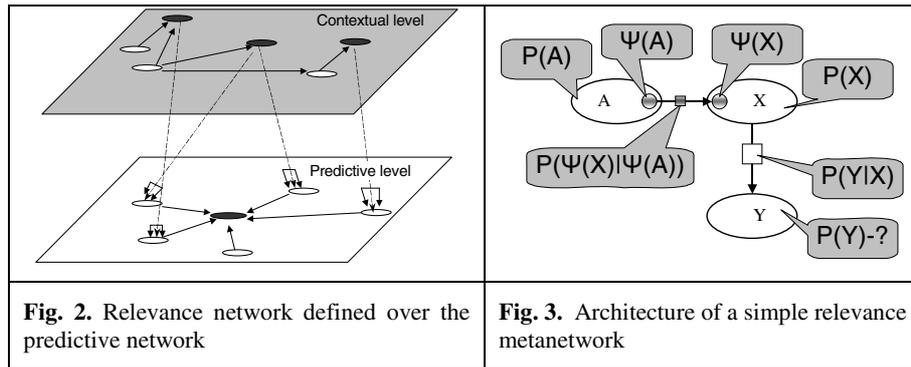
$$\psi_{x_1} = P(\psi(X_1) = \text{"yes"}); \psi_{x_2} = P(\psi(X_2) = \text{"yes"}); \dots; \psi_{x_n} = P(\psi(X_N) = \text{"yes"});$$

Let's estimate \$P(Y)\$ according to [18]:

$$P(Y) = \frac{1}{\prod_{s=1}^N n_{xs}} \cdot \sum_{X_1} \sum_{X_2} \dots \sum_{X_N} [P(Y | X_1, X_2, \dots, X_N) \cdot \prod_{\forall r(\psi(X_r) = \text{"yes"})} n_{xr} \cdot \psi_{x_r} \cdot P(X_r) \cdot \prod_{\forall q(\psi(X_q) = \text{"no"})} (1 - \psi_{x_q})]$$

*Relevance Bayesian Metanetwork* can be defined on a given predictive probabilistic network as it shown in Fig. 2. It encodes the conditional dependencies over the relevancies. Relevance metanetwork contains prior relevancies and conditional relevancies. Considering such definition of relevance metanetwork over the predictive network it is clear that the strict correspondence between nodes of both

network exists but the arcs do not need to correspond strictly (as shown in Fig.2). It means that relevancies of two variables can be dependent, although their values are conditionally independent and vice versa (Fig. 3). So, the topologies of the networks are different in general case.



In a relevance network the relevancies are considered as random variables between which the conditional dependencies can be learned. For example in Fig. 4, the probability of target attribute Y can be computed as follows:

$$P(Y) = \frac{1}{nx} \cdot \sum_X \{P(Y | X) \cdot [nx \cdot P(X) \cdot \sum_{\psi_A} P(\psi_X | \psi_A) \cdot P(\psi_A) + (1 - \psi_X)]\}.$$

### 3 Multilevel Context Extraction for Bayesian Metanetworks

Distinguishing between relevant and irrelevant features of the domain objects is, of course, extremely important for the decision making within that domain. However another problem, to sort relevant features either to contextual or to predictive ones, is as much important too. As we can see from e.g. Bayesian Metanetworks above, contextual and predictive features have different roles in the model and present on different levels of its organization.

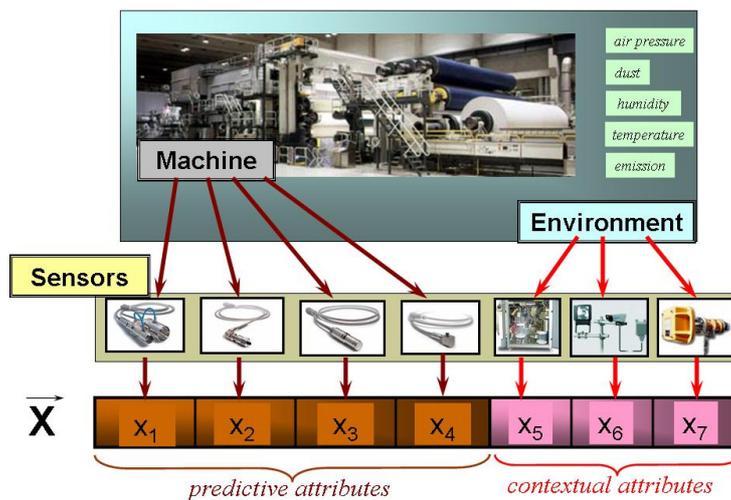
The theories of context according to [14] can be divided into two general types: the first, which sees context as a way of partitioning a global model of the world into smaller and simpler pieces; the second, which sees context as a local theory of the world in a network of relations with other local theories, can be considered as more general than the first one. On the other hand, contexts can be considered as local (i.e. *not shared*) models that encode a party's subjective view of a domain [15]. This makes contexts comparable and in some sense opposite to ontologies, which are considered as *shared* models of some domain that encode a view which is common to a set of different parties [16]. Contexts and ontologies have both strengths and weaknesses. It was argued in [17] that the strengths of ontologies are the weaknesses of contexts and vice versa. In [17] the attempt was made to contextualize the ontologies by acquiring certain useful properties that a pure shared approach cannot provide. The result is *Context OWL (C-OWL)*, a language whose syntax and

semantics have been obtained by extending the OWL to allow for the representation of contextual ontologies.

The above definitions are giving some hints on how to split the domain description (without complex mathematical processing) to predictive and contextual features, assuming that the goal is to enable reliable decision making based on Bayesian Metanetwork within that domain. In this chapter we consider three strategies of extracting context from relevant features, which are based on: *part\_of* context, role-based context and interface-based context.

### 3.1 *Part\_of* Context Extraction

It is known that it is more reliable to make decisions concerning any domain object if to take into account the environment within which this object is placed. For example in industrial applications related to condition monitoring, remote diagnostics, predictive maintenance, etc., it is really important to sense not only parameters of the machine (device) in question but also to measure the environmental conditions in which this machine is operating (See Fig. 4).



**Fig. 4.** To make diagnostics or to predict performance of some industrial machine it is reasonable to collect both: parameters measured directly from the machine and also parameters of the working environment of the machine

The attributes of the object and the attributes of its environment have different role in decision making process. If the first ones usually directly affect on the outcome (diagnosis, prediction, etc) and can be called “predictive” attributes, but alternatively the second ones most likely affect on the choice of right decision model for the diagnostics or prediction and can be called “contextual” attributes.

In general, the environment for any domain object is one or several other objects, which include this domain object as their part. For example, a department has some faculty as an environment, a wheel has some car as an environment, an arm has some

body as an environment, player Andriy Shevchenko has “Chelsea” football club as an environment, etc.

The idea of the *part\_of* context extraction is based on known hierarchy of the nested domain objects. If object A is part of object B (i.e. connected with *part\_of* relation on a semantic network), then all predictive attributes of object B will be contextual attributes for object A. This is illustrated in Fig. 5, where a sample of domain model represented by RDF<sup>1</sup>-based semantic network is shown. Also it is shown nested view to *part\_of* relation, which is also often used to visualize nested hierarchies. Using terminology of Semantic Web, in this example we have two resources: (a) Resource *k*, which is part of Resource *i*, has two datatype properties (property *q* with value *m* and property *p* with value *s*), and (b) Resource *i* itself with property *n* with value *r*. Actually we have four RDF statements: two about resource *k* based on two properties *part\_of* and *property\_n*, and two about resource *i* based on two properties *property\_q* and *property\_p*. In [19] the extension of RDF called CDF (Context Description Framework) is considered that allows making RDF statements in a context of some other RDF statements using for that *true\_in\_context* property for RDF statements (a kind of reification), and the value of this property is generally a contained of RDF statements. The CDF graph for the example above in also presented in Fig. 5. In the table from Fig.5 one can see a separation between predictive and

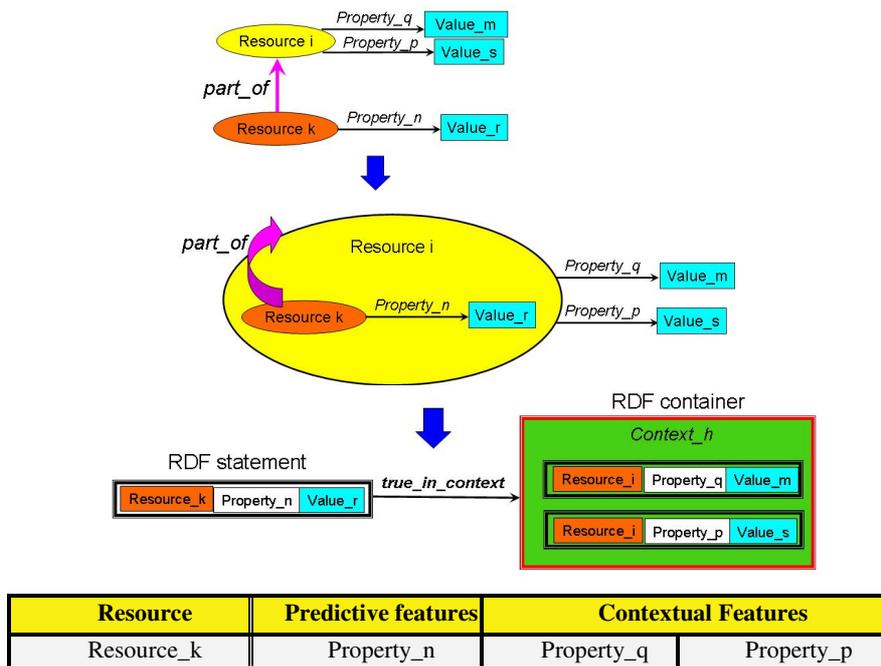


Fig. 5. The sample of *part\_of* context: (a) RDF graph view, (b) nested graph view, (c) CDF graph view, (d) table shows the separation of predictive and contextual features for Resource *k*

<sup>1</sup> <http://www.w3.org/RDF/>

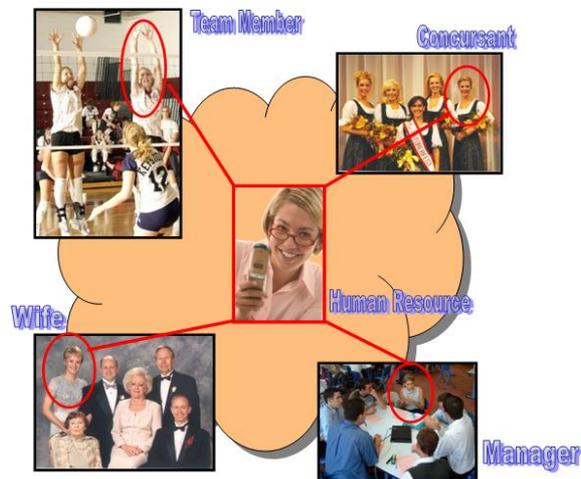
contextual features of the Resource  $k$ , which is based on *part\_of* relation. Thus possible Bayesian Metanetwork to model such sample will place predictive attributes to predictive level of the network and the contextual features to the contextual level.

The approach for feature separation described above is naturally recursive due to nested hierarchy of the domain provided by *part\_of* relation. If object A is part of object B and B is part of object C, then according to previous definitions it is true that: (a) predictive attributes of object B are in the same time contextual attributes of object A; (b) predictive attributes of object C are in the same time contextual attributes of object B. The above implies that the attributes of object C are in the same time *meta*-contextual attributes of object A.

A domain object generally can be part of several other objects. In this case its context should inherit all properties of its “parents”. For example if John is part of two objects “Golf Club” and “Symphonic Orchestra”. Thus the properties of John (e.g. age) should be considered in the context of all properties of the Golf Club and of the Symphonic Orchestra.

### 3.2 Role-Based Context Extraction

Another approach for context extraction is related to such domain objects, which are proactive components of some organizations or business processes. Most often this is applied to humans or intelligent agents. Such objects play certain roles in their organization or in their business process. The natural context for such objects descriptions can be the description of their current role (goals, duties, responsibilities, behavior, commitments, policies, etc.). In case if some object is in the same time member of several organizations (or processes) then all integrated duties should form the context of this object and possible contradictions should be resolved (see Fig. 6).



**Fig. 6.** The example of the proactive object (human resource), which is part of several organization and which is playing different roles in each of them. The context of this object should include the description of these roles (duties, commitments, responsibilities, etc).

As we can see some lady is the member (i.e. part of) several organizations (family, office, volleyball team, women's club). According to *part\_of* hierarchy the context for this lady description should include descriptions of all these organizations (similarly as with John in previous example). However the important part of the context will be also the description of the roles and appropriate duties the lady plays in these organizations (e.g. wife in the family, defender in the team, concursant in the club, manager in the office, etc.). The specific feature of the role-based context is that some commitments and duties related to someone's roles in different organizations can be contradictory and that is an important task of appropriate decision-making tools to resolve such contradictions.

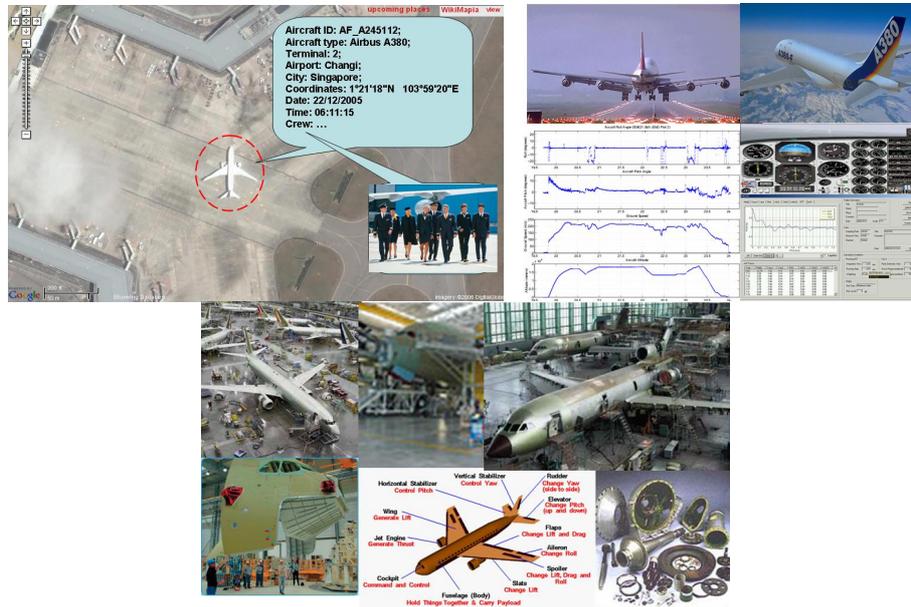
Consider two challenges related to *part\_of* hierarchies and appropriate contexts. The first one is the fact that the *part\_of* domain structuring (clustering), as well as any other domain ontology engineering, is essentially subjective. This means that the same object described according to two different domain ontologies will have two different sets of not only predictive features but also contextual ones. The second challenge is that *part\_of* hierarchies are generally dynamic and this result to the fact that the context is the function of time. For example certain object can proactively move from organization to organization, recreate commitments, change duties etc. This means that appropriate decision support system should take into account such temporal (and spatial also) dynamics of the contexts as well as its subjectivity.

### 3.3 Interface-Based Context Extraction

Another interpretation of a context and its influence to relevance of the domain objects' features is related to domain objects visualization through graphical user interfaces. We base on assumption that each interface is designed to certain category of users to provide them access to certain information needed to perform certain goal-driven activity. This means that the information about the same domain object being shown in different interface should be selected according to the goals assumed by each particular interface. Thus each interface can be considered as a kind of context, which affect on the set of relevant features of objects to be visualized through it.

In the example in Fig. 7 we are considering aircraft as domain object and we have three interfaces (i.e. three contexts) for presenting aircraft information to the users. The first one is for representing spatial information (Google Maps), the second one is pilots' control panel for representing aircraft operational parameters during the flight; and the third one is the aircraft design e-manual for aircraft manufacturers. Each interface is considered as a context, which affect on which parameters of the aircraft is reasonable to show through this interface. It is evident that not all possible parameters of the aircraft are relevant for the presentation of the aircraft in each of these particular interfaces.

One of specific features of such context-based visualization can be also "zooming relevance", which means that zooming of the interface screen (e.g. map view) may also lead to changes of parameters relevancy for the same domain objects on the screen.



**Fig. 7.** The example of the domain object (aircraft) is shown in different interfaces: (a) Google Maps; (b) pilots' control panel; (c) manufacturing design e-manual. Each interface is considered as a context, which affect on which parameters of the aircraft are to be shown.

## 4 Conclusions

Bayesian Networks are proven to be a comprehensive model to describe causal relationships among domain attributes with probabilistic measure of appropriate conditional dependency. However, depending on task and context, many attributes of the model might not be relevant. If a Bayesian Network has been learned across multiple contexts then all uncovered conditional dependencies are averaged over all contexts and cannot guarantee high predictive accuracy when applied to a concrete case. We are considering a context as a set of contextual attributes, which are not directly effect probability distribution of the target attributes, but they effect on a “relevance” of the predictive attributes towards target attributes. Distinguishing between relevant and irrelevant features of the domain objects is extremely important for the decision making, however another problem, to sort relevant features either to contextual or to predictive ones, is as much important too. In this paper we consider three strategies of extracting context from relevant features, which are based on: *part\_of* context, role-based context and interface-based context. The two challenges has been mention related to these strategies. The first one is the fact that domain models (providing the *part\_of* hierarchies), or organizational roles distribution, or interface modeling, etc., are essentially subjective. This means that the same object described according to two different domain ontologies will have two different sets of not only predictive features but also contextual ones. The second challenge is that

such contexts are generally dynamic. These challenges require from appropriate decision support system (e.g. based on Bayesian reasoning) to take into account such temporal (and spatial also) dynamics of the contexts as well as its subjectivity. The approaches to handle context described in this paper have been applied within SmartResource and UBIWARE projects<sup>2</sup>. In these projects we extended RDF to be applied to semantic annotation and visualization of dynamic and context sensitive resources [19] of different nature, as well as behavior and roles of these resources in various industrial business processes.

## References

1. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, San Francisco (1988)
2. Henrion, M.: Some Practical Issues in Constructing Belief Networks. In: Henrion, M. (ed.) Proceedings of the 3-rd Annual Conf. on Uncertainty in Artificial Intelligence, pp. 161–174. Elsevier, Amsterdam (1989)
3. Heckerman, D.: A Tutorial on Learning with Bayesian Networks, Technical Report MSR-TR-95-06 Microsoft Research, (March 1995)
4. Butz, C.J.: Exploiting Contextual Independencies in Web Search and User Profiling. In: Proceedings of the World Congress on Computational Intelligence, Hawaii, USA, pp. 1051–1056 (2002)
5. Boutilier, C., Friedman, N., Goldszmidt, M., Koller, D.: Context-Specific Independence in Bayesian Networks. In: Proceedings of the 12-th Conference on Uncertainty in Artificial Intelligence, Portland, USA, pp. 115–123 (1996)
6. Zhang, N.L.: Inference in Bayesian networks: The Role of Context-Specific Independence. International Journal of Information Technology and Decision Making 1(1), 91–119 (2002)
7. Geiger, D., Heckerman, D.: Knowledge Representation and Inference in Similarity Networks and Bayesian Multinets. Artificial Intelligence 82, 45–74 (1996)
8. Friedman, N., Geiger, D., Goldszmidt, M.: Bayesian Network Classifiers. Machine Learning 29(2-3), 131–161 (1997)
9. Cheng, J., Greiner, R.: Learning Bayesian Belief Network Classifiers: Algorithms and System. In: Stroulia, E., Matwin, S. (eds.) Advances in Artificial Intelligence. LNCS (LNAI), vol. 2056, pp. 141–151. Springer, Heidelberg (2001)
10. Pena, J., Lozano, J.A., Larranaga, P.: Learning Bayesian Networks for Clustering by Means of Constructive Induction. Machine Learning 47(1), 63–90 (2002)
11. Terziyan, V., Bayesian, A.: Metanetwork. International Journal on Artificial Intelligence Tools 14(3), 371–384 (2005)
12. Terziyan, V., Vitko, O.: Bayesian Metanetwork for Modelling User Preferences in Mobile Environment. In: Günter, A., Kruse, R., Neumann, B. (eds.) KI 2003. LNCS (LNAI), vol. 2821, pp. 370–384. Springer, Heidelberg (2003)
13. Terziyan, V., Vitko, O.: Learning Bayesian Metanetworks from Data with Multilevel Uncertainty. In: Bramer, M., Devedzic, V. (eds.) Proceedings of the First IFIP International Conference on Artificial Intelligence and Innovations (AIAI-2004), Toulouse, France, pp. 187–196. Kluwer, Dordrecht (2004)

---

<sup>2</sup> Projects of Industrial Ontologies Group, <http://www.cs.jyu.fi/ai/OntoGroup/projects.htm>

14. Bouquet, P., Ghidini, C., Giunchiglia, F., Blanzieri, E.: Theories and Uses of Context in Knowledge Representation and Reasoning. In: Akman, V., Bazzanella, C. (eds.) Special Issue on Context, *Journal of Pragmatics*, vol. 35(3), pp. 455–484. Elsevier, Amsterdam (2003)
15. Ghidini, C., Giunchiglia, F.: Local Models Semantics, or Contextual Reasoning = Locality + Compatibility, *Artificial Intelligence*, Vol. *Artificial Intelligence* 127(2), 221–259 (2001)
16. Patel-Schneider, P.F., Hayes, P., Horrocks, I.: Web Ontology Language (OWL) Abstract Syntax and Semantics, Tech. report, W3C Febr (2003), <http://www.w3.org/TR/owl-semantics/>
17. Bouquet, P., Giunchiglia, F., Van Harmelen, F., Serafini, L., Stuckenschmidt, H.: Contextualizing Ontologies. *Journal of Web Semantics* 26, 1–19 (2004)
18. Terziyan, V.: Bayesian Metanetwork for Context-Sensitive Feature Relevance. In: Antoniou, G., Potamias, G., Spyropoulos, C., Plexousakis, D. (eds.) SETN 2006. LNCS (LNAI), vol. 3955, pp. 356–366. Springer, Heidelberg (2006)
19. Khriyenko, O., Terziyan, V.: A Framework for Context-Sensitive Metadata Description. *Intern. Journal of Metadata, Semantics and Ontologies* 1(2), 154–164 (2006)