Library-style Ontologies to Support Varying Model Views

Linda C. van der Gaag
Hermi J.M. Tabachneck-Schijf

Technical Report UU-CS-2009-032
December 2009

Department of Information and Computing Sciences
Utrecht University, Utrecht, The Netherlands
www.cs.uu.nl

ISSN: 0924-3275

Department of Information and Computing Sciences
Utrecht University
P.O. Box 80.089
3508 TB Utrecht
The Netherlands

Library-style Ontologies to Support Varying Model Views

Linda C. van der Gaag, Hermi J.M. Tabachneck-Schijf
Department of Information and Computing Sciences, Utrecht University,
P.O. Box 80.089, 3508 TB Utrecht, The Netherlands.
e-mail: {linda,hermi}@cs.uu.nl

Abstract

The next development in building Bayesian networks will most likely entail constructing multi-purpose models that can be employed for varying tasks and by different types of user. In this position paper, we argue that the development of a special type of ontology to organize the knowledge involved in such a multi-purpose model is crucial for the management of the model's content. This ontology should preserve all knowledge elicited for the construction of the model and be accessible to domain experts and knowledge engineers alike. Based on the different ways in which people learn and gain expertise, we further argue that knowledge elicitation will result in task-specific knowledge mostly, which is best stored in the format in which it is elicited. To support varying model views for different tasks and different types of user, we propose that the elicited knowledge be organized in a library-style ontology of separate modules.

1 Introduction

While in the early years of the field of Bayesian networks attention focused primarily on algorithmic issues, the last decade has seen an increasing interest in methods to support the construction of such networks. The field also has become more and more experienced in building decision-support systems that include a Bayesian network. Bayesian networks by now have evolved beyond laboratory settings and are being employed by non-academic users. In turn, users of these network-based decision-support systems are starting to see the possibilities that these systems offer, and begin to ask for more. For some of our biomedical diagnostic applications for example, we have been asked whether we could perhaps adapt the model for teaching purposes. It is therefore likely that the next development in the field of Bayesian networks will entail building multi-purpose models which can be employed for different tasks and, in all likelihood, by different types of user.

In this paper we argue that to support model views for varying tasks, a suite of Bayesian networks should be built rather than a single unified network. We further argue that in the first step of developing such a suite, knowledge elicitation will necessarily result in task-specific information mostly, although also some task-neutral knowledge may emerge. Organizing the elicited knowledge into a library-style ontology of multiple task-specific and task-neutral modules, is then best suited to empower reuse of knowledge segments and to facilitate ready composition of model views. We also reiterate our view that this ontology should capture all elicited knowledge and be accessible to both domain experts and engineers.

We begin by defining different types of model view in Section 2, and outline the task model view under discussion in the current paper. We argue that a unified multi-purpose model for a range of tasks would quickly become too large and unyieldy to afford the knowledge engineers and the domain experts an overview of its contents. For one of our moderately-sized single-purpose networks in the biomedical field, for example, we noticed already that an experienced knowledge engineer who was new to the application, was not quite able to gain an overview of the network's contents without considerable help from the engineer who had originally constructed the network and its associated documentation. Since having a clear overview of the knowledge involved is crucial for the construction of task views, we advocate building a suite of, in all likelihood, smaller models, rather than a single Bayesian network.

In Section 3 we outline our view of ontologies for suites of Bayesian networks. We rationalize why an ontology should be constructed of the elicited knowledge, before actually developing the suite. Our ontology provides an unambiguous and well-structured documentation of *all* elicited knowledge and thus includes not just the knowledge that is explicitly captured in the suite's models but also any background and meta-level information that remains implicit in the suite. The latter types of information support, for example, viewing the elicited knowledge from different perspectives as required for developing different task views, and are likely to be instrumental in lessening future elicitation efforts for new views which may not have been foreseen at the suite's initial construction. The ontology, moreover, is enhanced with information to support future modeling efforts for such views. Note that our view of ontologies for suites of Bayesian networks thus is more comprehensive than current views of ontologies for knowledge-based systems in general. Our rationalization in Section 3 is much in line with our earlier arguments for developing ontologies for single, task-specific networks [15]. Our experiences with developing ontologies for single-model applications in the biomedical field moreover emphasize the importance of rich documentation, not just for constructing the model but also for its use and maintenance over a longer period of time [18].

We are not the first to suggest the use of ontologies. Ontologies are being developed for a variety of purposes, ranging from providing a portal for access to the semantic web, to documenting elicited knowledge for the development of knowledge-based systems in general, see for example [5, 12, 14, 31]. For many of these purposes, a rigorously formal, logic-based or other mathematical ontology language is used to allow automated processing. For our purpose of supporting the development and maintenance of a suite of networks by well-structured documentation however, the ontology should primarily provide a medium for communication between the engineers and the experts involved in the suite's construction. Having experienced that a rigorously formal language is not easily accessed by non-mathematical experts, we advocate, in Section 3, the use of a less formal language for our ontologies. One of the commonly-acknowledged drawbacks of using informal languages for documentation is that they do not allow automated processing. For our single-purpose Bayesian networks, however, we currently are gaining experience with developing ontologies using semi-formal representations which do allow automated processing to at least some extent [16].

In Section 4, we address the knowledge content of our comprehensive ontologies for suites of Bayesian networks. In order to align the content of the ontologies with elicited knowledge, we consider the processes by which humans learn and structure their own knowledge. We argue that the professional knowledge of practicing experts is mostly both task- and domain-specific, although also some task-neutral information may emerge during knowledge elicitation. In Section 5, we argue that the elicited knowledge is best stored in the form in which it is obtained, if only to forestall, as much as possible, the introduction of biases by the knowledge engineers involved. We further propose that the elicited knowledge be organized into separate modules to support use of segments of the knowledge in the construction of the suite's models and of their associated task model views. Organizing the modules in a library-style ontology further encourages re-use of knowledge segments when the suite is being extended to cover new tasks.

Considering semi-automated processing and modularizing knowledge, we would like to note that in our earlier work we propagated the evolution of a meta-library of generic knowledge structures [16]. An example of such a generic structure for the biomedical field captures all domain-independent knowledge that plays a role in relating test results to their underlying true value [17]. To exploit such a structure, the knowledge engineer instantiates it with domain-specific knowledge and uses the instantiated structure for semi-automated design of a segment of the Bayesian network under construction. Note that while this meta-library of generic structures includes only knowledge which is domain-independent and preferably task-neutral, the library-style ontology proposed in the current paper is tailored to a particular domain and to multiple specific tasks within this domain. The availability of a meta-library of generic structures would nonetheless support the construction of an ontology for a suite of Bayesian networks as proposed here.

The paper ends with a discussion and some perspectives for further elaboration of the presented ideas to a practicable knowledge-engineering approach to developing multi-purpose suites of Bayesian networks.

2 Model views of Bayesian networks

We distinguish two types of model view for a suite of Bayesian networks, namely *task model views* and *interaction model views*. To explain the difference between these two types of view, we distinguish three products delivered in the development of a suite of networks and review the steps to render them. We illustrate these products and steps by describing the development of a suite of models in the example domain of oesophageal cancer. Future users of the projected suite are attending oncologists, whom the suite should assist in staging their patients' cancers [10], and students, whom the suite should support by simulating tumor growth and explaining the consequences involved for a patient; a third group of prospective users are the engineers who are responsible for debugging and maintaining the suite.

The first product in the development of a suite of models consists of a stored pool of knowledge relevant to all tasks to be carried out with the suite. For the example domain, the pool should contain knowledge about relating the results of diagnostic tests to a stage of the cancer, as well as pathophysiological knowledge about how an oesophageal tumor invades, for example, blood vessels and thereby gives rise to secondary tumors distant from the oesophagus. The second product encompasses the actual suite of networks that allows computations to be carried out for the various tasks. For the domain of our example, the suite would include two models, one for the staging of oesophageal cancer based upon the results of diagnostic tests and one for simulating and explaining tumor growth. Note that these two models would differ, at least, in the amount of detail modeled: for example, while the network for the students would model the exact process by which cancer cells are conveyed through the blood vessels, the network for the oncologists would leave this process implicit, relating the presence of distant secondary tumors directly to a specific stage of the tumor's growth. The third product in the development of a suite of models comprises the interfaces, which are the concrete means that allow users to work with the suite. To allow an attending oncologist to consult the example suite of models, an interface closely resembling a patient's status would suffice, in which the test results obtained can be readily filled in; for the students, the addition of a more elaborate interface would be required, showing anatomical pictures, animations of tumor growth over time, and options for posing deeper 'why' and 'what if' questions.

In view of the three products outlined above, we briefly consider the steps that need to be taken to render them. The first step, leading to the first product, involves eliciting, structuring and organizing knowledge. The second step amounts to first selecting, from the pool of elicited knowledge, the knowledge that is going to determine the content and the structure of each of the models in the projected suite; the selected knowledge then is delimited in scope for example by making contextual assumptions, and subsequently represented in the mathematical formalism of Bayesian networks. This second step delivers the actual suite of models. The final step is characterized by designing interfaces to the suite, that is, the different ways the models can be presented to someone interacting with it, be this an engineer or an end-user focused on a specific domain task.

We consider a *task model view* to be one view of a suite of models. The task model view is the result of carrying out the elicitation and structuring of task-neutral and task-related domain knowledge and of making selections of the elicited knowledge to support a single or a few closely related tasks. For our example suite of models, one task model view might support the diagnostic reasoning task of an attending oncologist which includes the staging of his patient's cancer, while another task model view could support learning to understand the progression of cancer of the oesophagus. Note that these different task model views dictate different levels of modeling detail and, in fact, require different knowledge contents of the task-specific models in the suite. *Interaction model views,* on the other hand, comprise the interfaces of the suite that are tailored to task and user. For the diagnostics model view in our domain, for example, two interaction model views would be developed: one interaction model view could be optimized for data entry by an attending oncologist, and another might support maintenance of the model by the knowledge engineer.

To summarize, for different tasks to be carried out by different types of user, a suite of models can require several task model views, each of which can need several interaction model views. In an earlier paper, we laid out some methods to construct effective interaction model views for a single-purpose Bayesian network [28]. In the current paper, we concentrate on the elicitation, structuring and organization of domain knowledge to support the development of multiple task model views with a suite of networks.

3 Ontologies for Bayesian networks

A suite of Bayesian networks that supports several tasks in an application domain by different task model views, is very likely to be of a complexity necessitating development over multiple years, involving possibly different engineers and experts. Building and maintaining models of such complexity is a hard and time-consuming process. The expert knowledge elicited in the process constitutes a rich pool of information, segments of which can play different roles in the various tasks in the domain at hand. All this knowledge has to be carefully reviewed and structured, and ultimately captured in the formalism of Bayesian networks. In this process, a multitude of modeling decisions are taken as well as numerous decisions to demarcate the scope of the model. Such decisions tend to forestall an overview and thorough comprehension of the model by anyone who has not been intimately involved in its construction. We have experienced that construction and maintenance of large, unified networks are already seriously hampered if the elicited domain knowledge and the decisions taken are not made explicit by rich documentation [18]. This problem is bound to grow worse if a suite of networks rather than a single network is to be developed and maintained.

Having observed, in our earlier work, the advantages of developing an ontology for an application domain before actually building a network [15], we feel that the construction of a suite of models will especially benefit from an explicit, comprehensive ontology, which then serves not just as a documentation of all elicited knowledge but also, for example, as a means of ensuring consistency over the models within the suite and as a medium for communication between the experts and engineers involved. In this section, we discuss the varying roles of such an ontology in the construction and maintenance of the suite. We further address the language to be used for the ontology and argue that a properly developed ontology provides a scaffolding for semi-automated construction of network segments.

3.1 The roles of our ontologies

To support building and maintaining knowledge-based systems in general, sophisticated knowledge-engineering methodologies have been developed; among these is the well known and commonly employed CommonKADS methodology [25]. Most of these generally applicable knowledge-engineering approaches strongly recommend the development of a conceptual model for the domain of application before actually constructing a model in the knowledge-representation formalism to be used. Underlying this recommendation is the observation that capturing knowledge directly in the projected formalism may result in a representation in which the domain knowledge is not easily recognizable as a result of the modeling decisions taken. First developing a separate conceptual model may thus prevent unwarranted discrepancies between the elicited domain knowledge and its representation, which otherwise could adversely influence the system's contents and hamper its maintenance. In line with this generally accepted recommendation to first develop a conceptual model, we recently proposed to have this conceptual model be an ontology [15].

There exist many views of the concept of ontology in general; for a variety of views, see [5, 12, 14, 31]. In this paper, we use the term ontology to refer to an explicit specification of the elicited domain knowledge that is to be shared by the experts and the knowledge engineers involved in the construction and maintenance of a suite of Bayesian networks; the specification moreover is enriched with modeling information to support engineering of the suite. Ontologies for knowledge-sharing purposes have been studied extensively, both in theory [13] and in a more practical setting [14]. From this research, a number of criteria emerged for their construction and use. These criteria stress for example the importance of achieving clarity of the concepts and relations captured in the ontology to both knowledge engineers and domain experts, and of maintaining internal coherence and extendability. The criterion of minimal encoding bias in addition posits that the representation language to be used in the ontology should introduce as little bias as possible in the contents and structure of the elicited knowledge. Another often postulated criterion is the criterion of minimal ontological commitment, which states that the ontology should be developed independently of the projected use of the ontology and its contents, and hence be task-neutral. This criterion originates from the observation that any commitment for example to the problem-solving method that will be applied to the domain knowledge, will influence and thereby possibly bias the knowledge captured in the ontology [4]. In the remainder of this paper, we adopt most of these criteria; in Section 4 however, we will rationalize why we will not adopt the criterion of minimal ontological commitment just like that.

Ontologies developed for the purpose of knowledge sharing play multiple distinct roles in the construction of a suite of Bayesian networks. One of these roles is to make all elicited domain knowledge explicit. To this end, the ontology specifies not just the knowledge that is to be captured explicitly in at least one of the networks, but also the relevant background knowledge of the domain that will remain implicit in the suite and the meta-level knowledge of its regularities and organizational structure. While all intricacies of the domain should be captured in the ontology to achieve clarity, that is, to avoid multiple interpretations and lack of understanding, details should also be hidden by including meta-level information about the structure of the knowledge to preserve understandability and transparency on an overview level. A second prominent role of our ontology is to provide an explicit medium for communication between domain experts and engineers. The contents of the ontology should thus be unambiguously understood and agreed upon by all agents involved [31], allowing them to communicate about the domain knowledge without any misconceptions when building, evaluating and maintaining the suite of networks. We return to these two roles of our ontologies in Section 3.2. Another important role of ontologies designed as rich documentations of elicited knowledge, is to scaffold the construction of the projected suite of networks. Since the ontology includes all ingredients to be modeled in the networks, it essentially allows a process of semi-automated construction in which the knowledge engineer selects and pre-processes ontology segments which subsequently are translated into network segments for further processing; we return to this idea of semi-automated construction of networks from our ontologies in Section 3.3.

3.2 The language of our ontologies

The representation language used for capturing and organizing elicited domain knowledge influences the extent to which our ontologies can meet the various roles outlined above. The issue of selecting an appropriate language has been addressed by many researchers: some suggest that domain knowledge should be represented in natural language or in a language that is semi-informal [31], yet others argue that ontologies should be specified in a rigorously formal language and, in fact, should be machine readable [26].

An important argument for using a mathematically formal ontology language is that it allows a highly structured and unambiguous representation of the elicited knowledge. Such a formal representation in turn is likely to allow (semi-)automated derivation of segments of the Bayesian networks under construction. Rigorously formal languages often have limited expressiveness however, as is evidenced for example by the many attempts to extend first-order predicate logic to allow modeling of specific knowledge constructs. In our view, it is important that an ontology language should come with a rich semantics to introduce as little bias as possible in the represented contents. If the language would introduce biases, then the resulting ontology might not properly reflect the intricacies of the domain. Since the ontology is used for the construction of a network, the resulting model might then be biased as well, possibly in unforeseen ways. The development of an independent knowledge model recommended by most knowledge-engineering methodologies in fact, has its origin in this observation.

While the possibility of automated processing favors using a rigorously formal ontology language, the purpose of knowledge sharing provides a strong argument for using a less formal one. Since the contents of the ontology should be unambiguously understood by both the knowledge engineers and the domain experts involved in a suite's construction, it should be represented in a language that is understandable for all. The language of Bayesian networks is often considered suitable for this purpose: many argue that the graphical structures which capture the qualitative domain knowledge are intuitively understood by all readers, the domain experts included [20]. We experienced on many occasions, however, that experts who are not familiar with the Bayesian-network language nor with the concept of probabilistic independence, tend to misinterpret the graphical structure [11, 28]. For example, a medical diagnostic network will typically contain arcs from a disease to its various symptoms. A physician, however, may be inclined to reverse these arcs, since upon establishing a diagnosis for a patient reasoning typically goes from observed symptoms to disease. Yet, a network containing the reversed arcs may not correctly represent the independences that hold in the domain of application. In our opinion, many of the formal languages commonly used for ontologies are unsuitable for checking accumulated knowledge with non-mathematical experts. If the use of a formal language is uncommon in a domain of application, then a rigorously formal language is unsuited for the purpose of knowledge sharing between the knowledge engineers and the domain's experts and a less formal language had best be used.

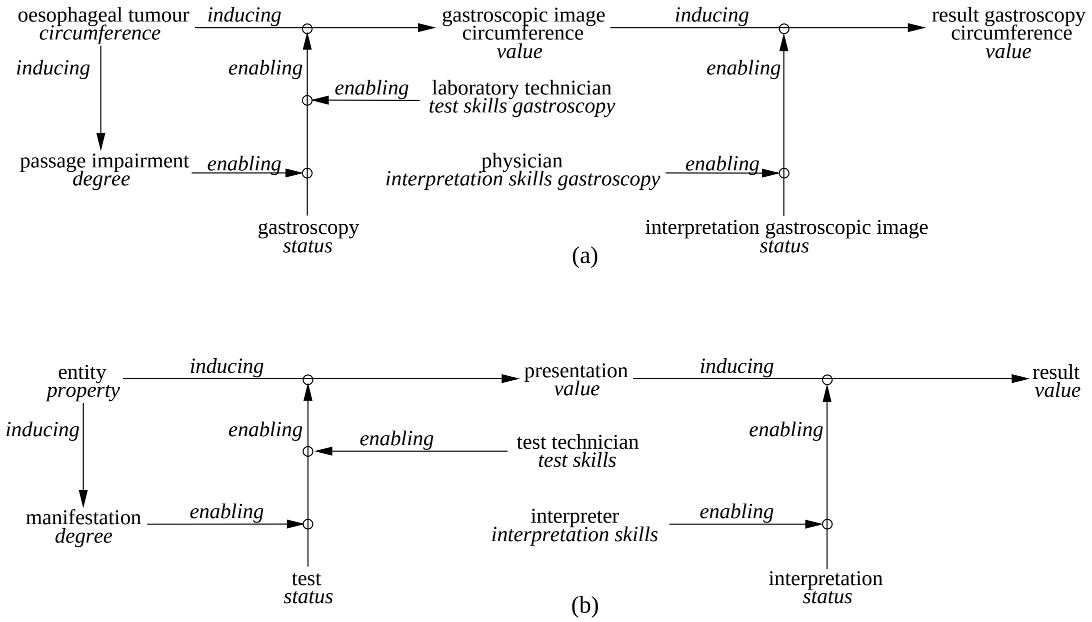

Figure 1: Relations between test results and the underlying true values, (a) for a gastroscopic examination of the circumference of an oesophageal tumor, and (b) for a diagnostic test in oncology in general.

For developing single-purpose Bayesian networks in the biomedical domain, we are currently gaining experience with a semi-formal ontology language composed of well-structured tables, depictions, graphs and hierarchy representations combined with text [15], which can be understood by both domain experts and knowledge engineers and allow automated processing to at least some extent. As an example, Figure 1(a) shows part of the ontology that we developed for the domain of oesophageal cancer. The depicted graphical structure captures knowledge about the relationship between the result of a gastroscopic examination of the circumference of a patient's tumor and the underlying true circumference. Each node in the structure depicts a particular state; this state reflects a situation, in time, in which a property of an object or process has adopted a value. The top leftmost node in the depicted structure, for example, models the true circumference of the oesophageal tumor of a particular patient; the bottom leftmost node captures the degree to which the swallowing capabilities of the patient are impaired. Labeled arcs are used to denote relations between the modeled states; associated with the arc is a table which details the relation in terms of the values of the related states. In our ontology language, different types of relation are defined. Figure 1(a) includes, for example, inducing relations and enabling relations. An inducing relation asserts that a particular state may induce another state; this relation involves time in the sense that the induced state cannot occur before the inducing state has occurred, yet leaves implicit the process by which the latter state induces the former one. The inducing relation between the two leftmost nodes in the depicted structure, for example, expresses that the circumference of a patient's oesophageal tumor may induce an impairment of the patient's swallowing capabilities; the table associated with the labeled arc (not shown in the figure) further details the relation by describing that the larger the circumference is, the higher the degree of impairment will be. An enabling relation differs from an inducing relation in that it links a state to a relation rather than to another state; it expresses that the enabling state must occur before the enabled relation can take effect. The enabling relations in the left part of Figure 1(a), for example, express that a gastroscopic examination of the patient's oesophagus can result in an image from which the tumor's circumference can be established, only if two conditions are met: the patient's swallowing capabilities should not be too seriously impaired and the laboratory technician should have sufficient skill in performing the gastroscopy.

The entire oesophageal cancer ontology includes not just the knowledge involved in interpreting the

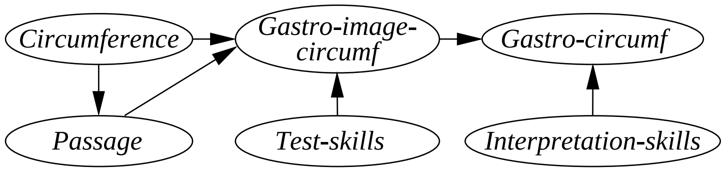

Figure 2: The initially derived probabilistic graphical structure.

result of a gastroscopic examination of the circumference of a patient's primary tumor, but also the knowledge involved in relating the results from various other diagnostic tests to their true underlying values. In addition to the knowledge pertaining to these tests separately, the ontology specifies the high-level regularities in the knowledge involved. The graphical structure capturing these regularities is depicted in Figure 1(b). The leftmost part of the structure, for example, expresses that the tested property may induce a manifestation; this manifestation may enable, possibly negatively, the performance of the test involved, which in turn may enable, again perhaps negatively, the test to yield a result. In our earlier research, we already argued that these high-level regularities can be exploited upon extending the network with the results of new tests, which might not have been foreseen at the network's inception: the high-level graph shows which knowledge may be involved and can thereby provide guidance for the elicitation of knowledge pertaining to the new tests. The high-level graph in addition is instrumental in maintaining internal coherence and consistency of the captured knowledge in that it serves to draw attention to any deviations of the newly added knowledge from the regularities that are present in the already included knowledge.

For further details of our ontology language and of the oesophageal cancer ontology more specifically, we refer the reader to [18].

3.3 Ontology-supported construction of networks

Upon building a suite of Bayesian networks for an application domain, it is a daunting prospect to have to capture all elicited knowledge in two ways, that is, first in a comprehensive ontology and then in actual networks. A carefully structured ontology, however, can be used to derive the networks' graphical structures in a semi-automated fashion. We briefly sketch such semi-automated construction; for further details, we refer the reader to [16].

Deriving probabilistic graphical structures from an ontology is done by a sequence of steps, some of which need be performed by a knowledge engineer and some of which can be supported by automated processing. First, the knowledge that is to be captured in a projected network needs to be selected. This first step involves a careful review of the elicited knowledge and must necessarily be performed and documented by the knowledge engineer. In the next step, the states and relations from the selected parts of the ontology are combined into a single graphical structure. Automated processing can to some extent support the knowledge engineer in performing this step. The resulting structure in essence describes all knowledge that needs to be captured in the projected network, but is not yet stated in the language of Bayesian networks: the structure may for example include enabling relations which link a node to a relation rather than to another node as would be required for a Bayesian network. We observe, however, that the structure states the selected knowledge in terms of properties of objects or processes. Since such properties typically play the role of variables in the application domain, the structure provides a convenient point of departure for deriving an initial network structure.

For deriving a probabilistic graphical structure, first the properties from the ontology's structure are translated into stochastic variables. From the example structure of Figure 1(a) for example, the degree to which a patient's swallowing capabilities are impaired, is translated directly into a stochastic variable for the Bayesian network under construction. Since translating properties into stochastic variables may involve re-defining some of the concepts involved, this step is performed by the knowledge engineer, possibly supported by software. The next step is to specify the arcs for the network's graphical structure. Since the relations in the ontology's structure capture (mostly) directional influences between the modeled properties, these relations can be translated more or less directly into arcs. Enabling relations, however, cannot be translated in this way, since the language of Bayesian networks does not allow arcs pointing to other arcs. We now observe that enabling relations essentially are indirect influences. In Figure 1(a), for example, the two enabling relations in the leftmost part of the structure link an impaired passage of food to whether or not a gastroscopic examination will yield an appropriate image, and can thus be translated in an arc from the stochastic variable modeling the degree of passage impairment to the variable capturing the gastroscopic image. The steps involved in the translation of relations from the ontology's structure to arcs in the probabilistic graphical structure can be performed in an automated way. The graphical structure resulting from this step is stated in the language of Bayesian networks, but is not guaranteed to properly reflect the independences holding in the application domain. In the final step of the derivation, therefore, the engineer has to meticulously verify that the structure correctly captures probabilistic independence. Also, the structure may need some optimization, for example for reducing the burden of probability elicitation. Figure 2 shows the probabilistic graphical structure that may thus be derived from the ontological structure of Figure 1(a).

4 Eliciting ontology knowledge

Given the prospective advantages of constructing a comprehensive ontology before actually building a suite of Bayesian networks, we now turn to the question of how to organize the elicited knowledge in the ontology so that it most usefully supports different task model views for the suite.

Many researchers recommend that ontologies be constructed independently of the projected use of the ontology and its contents, that is, many researchers advocate minimizing ontological commitment; see for example [4]. As already mentioned in Section 3, underlying this recommendation is the argument that any commitment to for example the problem-solving method to be applied may bias the contents of the ontology, and may thereby hamper its extendability and re-use. Constructing an ontology without any commitments to a particular task however, requires either eliciting task-neutral knowledge from domain experts, or stripping the task-specific aspects from the elicited knowledge. In this section, we address the feasibility of the first option; the second option is briefly addressed in Section 5.

We consider eliciting task-neutral information, that is, eliciting knowledge from experts without them having a particular task in mind. To provide task-neutral information, experts should be able to gather such information from their minds, which implies that the knowledge should be stored in their brains in such a way that task-neutral aspects are readily separated from task-specific aspects. We briefly lay out the different ways in which people learn information, and argue that these learning processes imply that the knowledge stored in the human brain is largely both domain- and task-specific. We then conclude that, given how knowledge is learned and stored, it would be extremely difficult to elicit task-neutral knowledge from an experienced professional.

4.1 Human knowledge acquisition processes

Humans in general, and hence professionals as well, gather knowledge over their entire lifetime, not just during periods of formal education. Many publications in the social sciences attest to the importance of learning outside of formal education: incidental, experiential, non-formal, informal and on-the-job learning are some of the terms used to describe this type of learning, see for example [1, 9, 22]. The definitions for this type of learning all differ, but they have in common that the learning occurs in a natural setting, be that farming in Senegal [23], college teaching in the Netherlands [7], or a childhood home. Learning in a natural setting can be intentional, in the sense that the learner is actively trying to learn something. Such learning occurs, for example, through an apprenticeship or by modeling, that is, purposely attending to how an expert performs a task with the intention to copy the skills. The learner can also acquire knowledge incidentally, having no intention to learn. This type of learning occurs very frequently: a child, for example, picks up seeing, hearing, feeling, tasting and smelling skills in order to make sense of ambient information, and learns to interact with its environment by movements, language and social skills, all without any formal teaching or any intention to learn. A physician similarly picks up medical knowledge and professional social skills while practicing medicine. Intentional and incidental non-formal learning share the fact that knowledge is acquired in the context of performing tasks in a natural setting, which usually is much richer than the setting provided in formal eduction. Early research has already shown that incidental learning does not necessarily have to result in less knowledge than intentional learning [19].

Emphasizing the omnipresence of non-formal learning and the importance of the setting in which the learning occurs, a unified view of learning was popularized in the late 1990s [32]. Although this view of learning may not be a mainstream learning theory, we feel that it provides a useful way to think about how and when what type of knowledge is acquired by humans in general. To support our rationalization of why it will be difficult to elicit task-neutral knowledge from experienced professionals, we therefore briefly describe this unified learning view.

In the unified view of learning, knowledge gathering is divided into four categories: transmission, acquisition, accretion, and emergence. The four learning processes are summarized in Figure 3; the left part of the figure describes which learning processes are at work at which stage of a lifetime, the right part shows the proportions of knowledge accounted for by the four processes. When people are asked to describe learning processes, they generally only mention the intentional processes of *transmission* and *acquisition*. We therefore describe these two processes first. Knowledge gathered during formal education from books and teachers is explicitly transmitted, that is, teachers tell their students what to read and do when, and provide explanations and scaffolding so that the students are encouraged to store the knowledge in their brains. Language and mathematical skills are examples of such transmitted knowledge. Note that these skills are not specific to a particular task. In fact, transmitted knowledge is often purposely task-neutral. In transmission learning, the knowledge is intentionally offered, but the amount that actually ends up in the brain depends highly on the student. The learning view estimates that over the course of a lifetime transmission accounts for only some 10% of the knowledge of an average person [32].

Further intentional learning is *acquisition learning*, which is estimated to be good for some 20% of a person's knowledge. Acquired knowledge is gathered by the learner's own initiative: by exploring, experimenting, self-instruction, inquiry, intentional modeling and the like; this type of learning has also been termed active learning or inquiry-based learning [27, 30]. Because acquired knowledge is gathered by the learner's initiative, the chance that it will actually end up in the brain is higher than for transmitted knowledge, where the learner may or may not be interested enough to store the knowledge. Whether acquired knowledge is task-neutral or task- and domain-specific depends on the setting in which it is gathered. If the knowledge is acquired within the formal education process, it may still be partly task-neutral; if it is gathered in a natural setting, it will most certainly be task- and domain-specific, because it will be embedded in and driven by the demands of the setting.

When asked to distinguish between different learning processes, people may perhaps mention *emergence* in addition to transmission and acquisition learning; the fourth type of learning, called *accretion*, does not commonly come to mind however. Yet, while emergence is estimated to account for only some 1-2% of the knowledge of an average person, accretion is estimated to account for as much as 70% of what a person knows [32]. Although emergence is relatively rare, it is important for example in pushing science ahead: it is the result of self-constructing new ideas and meanings that did not exist before. While accounting for most of what is stored in a person's brain, accretion is often not thought of as learning: it is the gradual, often subconscious, process by which we learn for example social rules and behaviors [32]. Accreted skills are typically learned without the mediation of language and are picked up just by sensing and processing information and by acting in and reacting to sensed information in a natural setting. Accretion, however, is not just about the large amounts of information that are unconsciously being processed, it is also about information that may be consciously processed but is not intentionally learned. Just by processing encountered information, a trace of it is stored in the brain. As with all learned information, the setting in which the information was learned is stored with it. Accreted knowledge is used intensively: it is triggered, or recognized, every time a situation is experienced that contains enough aspects that are similar to the one the information was learned in. The fact that much accreted knowledge is not available on demand, meaning that we cannot readily activate it by conscious effort, doesn't make it any less powerful or important. The unconscious processing system in fact has a much higher processing capacity than the conscious system [6]. Current research even shows it to be better to only use unconscious thought when making complex decisions, as the limited capacity of conscious thought leads us to focus on subsets of information which quite often are not optimal [6, 24]. It follows from the above that because accreted knowledge is generally picked up and triggered within rich natural settings, it is largely both task- and

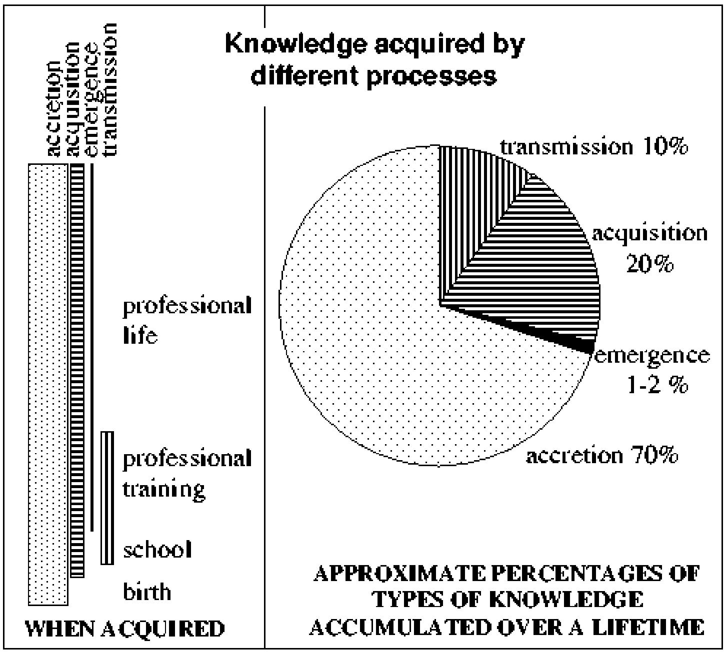

Figure 3: Knowledge gathered by the four different learning processes.

domain-specific.

4.2 Example: the acquisition of medical knowledge

While the four processes reviewed above relate to general learning practices, they are easily mapped onto what happens in the course of gathering professional knowledge. Although the exact percentages may vary somewhat for experienced professionals, the four processes will create roughly the same proportions of their knowledge. We illustrate the different learning processes with the example of gathering professional knowledge in medicine [3].

The basics for medical knowledge are taught by transmission in universities. The transmitted knowledge consists of task-neutral biomedical information, which is mostly causal and definitional in nature and describes the functioning and possible dysfunctioning of the human body [3]. It is this transmitted knowledge that upon elicitation would result in task-neutral knowledge segments. Next, in internships, the students are confronted with real patients in a real hospital setting. They now have to link the transmitted task-neutral information to actual clinical situations. In contrast to most biomedical knowledge, the knowledge required to attend to real clinical problems is task-specific in nature. It consists of knowledge of symptoms, of differential diagnoses and how to construct them, and of treatments for diseases and how to weigh the benefits and risks of these treatments, all embedded in concrete medical situations [3]. In internships, some transmitted information is still offered, but students are also acquiring knowledge by trying to figure out diagnoses and treatment plans themselves. Accretion is also at work, continually recording information from all perception faculties. Examples of accreted knowledge range from how to read symptoms from patients' look, smell, utterances and behavior, to how to communicate with colleagues, patients and their next of kin, to how to get around in the hospital and many other aspects of work. All that is learned is now embedded in the task at hand and in the medical culture and practices. In cognitive science terms, we say that the knowledge is *situated*.

It is taking the step from employing task-neutral knowledge in college to having to apply task-specific knowledge in a hospital setting that makes the transition from the university classrooms to practice so problematic for many medical students [3]. Students may have learned which disease causes which symptoms, and maybe even have seen pictures of such symptoms. However, recognizing the symptoms when exhibited by a patient is a very different matter. Each patient is unique, and may or may not exhibit all of the symptoms, which may or may not look like the description or picture in the book. Patients may further have more than one disease, which may result in an indistinct mixture of symptoms. And confusingly, many early symptoms of severe illnesses can look like relatively innocent diseases. Last but not least, the reasoning required now goes diagnostically from symptoms to disease, not causally from disease to symptoms. To allow its ready activation, the knowledge involved needs to be re-represented in the brain. The issue of having to re-represent knowledge is supported by research in various other contexts, from which it is also clear that switching information from one representation to another is very difficult. Switching representations, in fact, does not occur spontaneously and must be explicitly and extensively taught [2, 29].

Professional learning in medicine does not stop with the internship phase. It continues largely by a mixture of accretion and acquisition during the entire career. All knowledge picked up in this phase is in a task-specific format, because it is learned while carrying out specific tasks in the professional setting. In a physician, for example, interaction with patients is typically stored as cases, which are exemplars of sick people complete with diagnosis, treatment plan, and outcomes. The theory of situated learning describes this phenomenon and argues that learning as it normally occurs is a function of the activity, context and culture in which it occurs [21, 27]. In fact, the theory argues specifically that learning *never* occurs in a task-, context-, and culture-neutral manner.[1]

4.3 Eliciting task-specific knowledge

From the above observations, we conclude that the bulk of the professional knowledge of an expert is stored in the brain in a task- and domain-specific format. It is therefore reasonable to assume that most of the knowledge that comes to the fore upon elicitation is task- and domain-specific. Of course an engineer can explicitly ask a domain expert to provide task-neutral knowledge. If experience from practice is requested, however, the engineer is asking for extra information processing from the expert: the expert has to relate his or her knowledge in a different way than is stored in the brain. This, as argued in the example above of the medical students' transition from textbook knowledge to diagnostic and treatment knowledge, requires non-trivial effort, which, as it is to be done in real-time, will at least considerably slow down the elicitation. More potentially damaging even is that asking people to relay knowledge in a way that requires them to reason *about* their stored knowledge always increases the risk of introducing errors [8]. We conclude that, except for information that was transmitted in a task-neutral fashion, it will be difficult, time-consuming and error-prone to try to elicit task-neutral knowledge from domain experts.

Two recent examples from our own research will serve as illustrations. As a first example, when we asked experienced pig veterinarians to supply us with average disease symptoms for pigs that were sick, most of them provided us with symptoms belonging to one particular illness rather than a context-free average; some gave symptoms associated with a particular group of closely related diseases such as infections of the respiratory tract. A likely explanation of what happened is that the veterinarians called to mind a pig having a particular disease, of which they provided the symptoms: the veterinarians unwittingly related our question to their daily practice, thereby situating it in the setting in which they had learned the requested information, and consequently rendered their knowledge in the way it was stored. The veterinarians providing a few more symptoms ostensibly generalized but actually were doing exactly what their colleagues did: they provided the symptoms of diseases encountered within the same differential diagnosis.

As a second example, we recall a knowledge-elicitation session where we asked a group of veterinary experts to reason out loud about particular pig cases of which the clinical symptoms were described in terms of variables and values. When asked what would happen to their assessment of the case when a particular symptom was changed from present to absent, one of the participants asked, in earnest, how he could possibly change the symptoms of a pig. Very likely, the veterinary expert had called the case to mind as a concrete pig for which he had to come to a diagnosis. Thinking in this task-related setting, he could not imagine physically changing a pig's symptoms, which is in reality of course indeed impossible.

[1] According to this theory, the knowledge transmitted in medical school is also not task-neutral: the task is passing the exam. For our purpose, however, the issue is that the knowledge is independent of specific medical tasks.

5 Storing the elicited knowledge

Having established that it is rather unlikely that an engineer will elicit knowledge from a domain expert that is altogether task-neutral, we now address how the elicited knowledge is best stored in an ontology. More specifically, we compare constructing a single unified task-neutral ontology that is free of task biases, with constructing multiple task-specific ontology modules, and reject the former option in favor of the latter one. We then argue that a library-style organizational structure for the ontology best supports the development of a suite of Bayesian networks with multiple tasks views.

5.1 A single unified ontology or multiple ontology modules

We begin by comparing capturing all elicited knowledge in a single unified task-neutral ontology, with modeling it in multiple (mostly) task-specific ontology modules. For the construction of a single unified ontology, be it composed of task-neutral or task-specific knowledge, plead that no duplication of knowledge is needed and that it will be easier to ensure internal consistency upon maintenance and extension. In spite of these advantages, however, we reject building a single ontology. A unified ontology for a suite of Bayesian networks with multiple task model views is likely to become quite large in size. Even if it is well organized and highly structured, its mere size will cause the knowledge engineers and the domain experts to quickly lose track of its contents. Another argument against the construction of a single unified ontology is that it may be much more difficult to build multiple task model views from a single entity than from a collection of entities. Given the stronger arguments against a unified ontology, we propose to construct multiple task-focused ontology modules.

Addressing the format of the ontology's content, we note that there are quite strong arguments for storing knowledge in a task-neutral fashion. For example, if a particular segment of the domain knowledge plays a role in multiple different tasks, then storing it in a task-specific fashion would require capturing it multiple times, each time from a different perspective. Storing the knowledge in a task-neutral fashion, on the other hand, would not need such duplication. Also, when new task model views need to be developed, it is quite likely that these are already supported by the available task-neutral knowledge. If the knowledge had been stored in a task-specific fashion, developing a new task-specific ontology module would be required.

Although there are strong arguments for storing the elicited knowledge in a task-neutral fashion, it generally will be infeasible to do so. In Section 4, we argued that the bulk of the elicited knowledge will be available in a format that is both task- and domain-specific. Constructing a task-neutral ontology would thus require stripping the elicited knowledge from its task biases and integrating the resulting segments of neutral knowledge. The task of stripping the elicited knowledge from its task-specific context is non-trivial, however. In our opinion, it is even infeasible since not just the experts but also the engineers will have particular tasks in mind when surveying the various segments of knowledge. The engineers moreover are likely to be insufficiently knowledgeable in the domain of application to recognize the various included task biases. Although storing knowledge in a task-neutral fashion is preferred, therefore, its infeasibility supports storing the bulk of elicited information in a task-specific format.

We would like to note that although most of the elicited knowledge will be available in a task-specific fashion, some of it may be task-neutral, for example if originating from the transmission phase of learning professional knowledge. Also, some of the elicited information can be easily abstracted to create segments of task-neutral knowledge. An example of the latter observation comes from one of our veterinary applications. Upon reviewing the knowledge involved in measuring a pig's body temperature, our veterinary practitioners mentioned that catching a pig will cause stress to the animal and thereby raise its body temperature. The stress effects of catching a pig are not specific for the task for which it is being caught however, and hence are relevant for any task in which the animal is being handled. Some of the knowledge elicited in the contexts of the various tasks to be supported thus is explicitly reusable and can in fact be stored in the prefered task-neutral format.

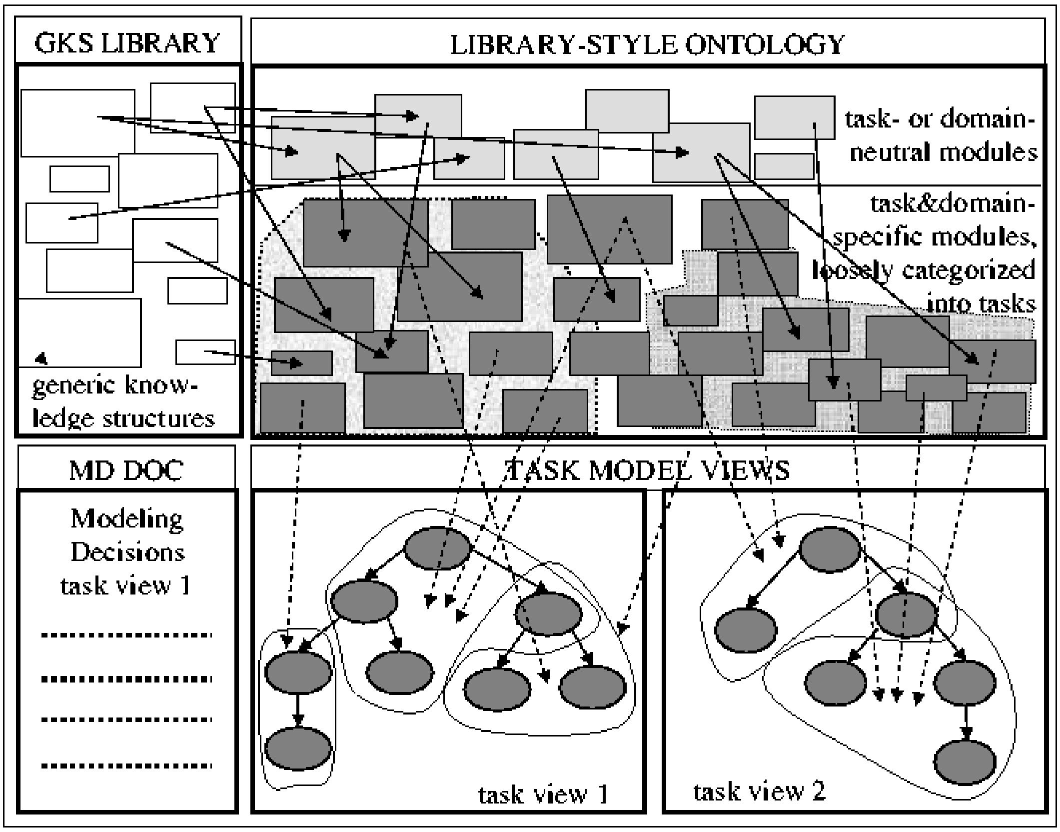

Figure 4: A library-style ontology for developing task model views: the library of ontology modules is supplemented with a library of generic knowledge structures and a document of modeling decisions; drawn arcs indicate instantiation of modules, dashed arcs indicate selection.

5.2 A library of ontology modules

Based upon the observations above, we envision capturing all elicited knowledge in an ontology composed of multiple modules, most of which are domain- and task-specific and some of which are task-neutral. As an example we consider again our application in oncology from earlier sections. Some of the modules of the ontology contain background knowledge that is common to all tasks in the domain yet independent of a specific task. An example of such knowledge is anatomical knowledge summarizing the structure and elements of the oesophagus and its neighboring organs in the human body. This knowledge is common to most tasks in the domain yet is not specific for any task in particular. It would be stored therefore in one or more task-neutral ontology modules. Other modules of the ontology contain knowledge that is common to one task but holds across domains. The graphical structure from Figure 1(b) pertaining to the interpretation of the results of diagnostic tests, showed a segment of such knowledge. The majority of the modules, however, capture knowledge that is both task- and domain-specific. Knowledge of which test results are the strongest indicators for particular stages of the patient's cancer for example, is strongly linked to the task of diagnosis and would be included in a task-specific ontology module for diagnostic tasks. Note that knowledge may have varying gradations of task specificity. For example, knowledge of when a patient for which cancer stages can be subjected to curative treatment and when he should receive palliative care, is not just linked to the choice of treatment but also to the task of prognostication. This segment of knowledge may thus be captured in more than one task- and domain-specific module, possibly described from varied perspectives.

From the above observations, we have that the domain knowledge elicited for a suite of Bayesian networks is captured in a collection of, mostly task- and domain-specific, ontology modules. To support the construction of multiple task model views for the suite, we now envision storing these ontology modules in a library-style organizational structure. For developing a particular task model view, the knowledge engineer retrieves from this library the required modules and combines them into a task-focused ontology to support the view. For example, to construct a concrete task-focused ontology for the model view of teaching diagnostics for the domain of oesophageal cancer, the task-neutral modules of anatomical knowledge would be selected as well as modules related to the tasks of diagnosis and prognostication; the modules of anatomy and prognostication would be included in the task-focused ontology to serve both simulation purposes and answering in-depth 'what-if' questions. Note that the other modules from the library need not be considered upon constructing the teaching-view ontology. For supporting a model view of diagnostics for an attending oncologist, on the other hand, the knowledge from the task-neutral modules of anatomy would most likely not be included explicitly in the task-focused ontology, as the model to be developed could leave this knowledge implicit. Now suppose that an ontology for the new task model view of predicting the effects of treatment is to be developed. Any task-neutral knowledge required for the new model view ideally is already present in the library and can be pulled in. Also the ontology module of prognostication already present in the library, captures some of the knowledge for the new task and can be used. In addition, however, new task-specific modules describing the physiological effects of treatment need to be developed and included. The knowledge for these new modules needs to be elicited from domain experts, where the elicitation can focus on just the task at hand.

Figure 4 sketches the basic idea of our library-style ontology for a suite of Bayesian networks. The top right part of the figure depicts the collection of task-neutral and task- and domain-specific ontology modules. This collection of modules is the pool of all elicited knowledge which constitutes the basis for developing different task model views for the suite. The top left part of the figure depicts the meta-library of generic knowledge structures mentioned in the introduction and conveys the idea that generic knowledge structures can be exploited to support capturing domain knowledge in ontology modules. The bottom right part of the figure shows the collection of task model views for the suite of networks. These task model views are constructed by selecting appropriate segments of knowledge from the library and combining these into a unified view. Upon constructing the task model views typically many modeling and demarcation decisions are taken. The bottom left part of the figure suggests that these decisions are also documented and maintained in the overall ontology.

6 Concluding observations

In this position paper, we argued that multiple task model views for a suite of Bayesian networks are best supported by a library-style ontology composed of mainly task- and domain-specific knowledge modules.

In posing our views, we addressed several issues. We began by reiterating the need of documenting all elicited knowledge. If this knowledge is not properly documented, construction and maintenance of large suites of networks inevitably become problematic. We recommended building an ontology to provide a well-structured explicit specification of the elicited knowledge and a medium for communication for the knowledge engineers and the experts involved in the networks' development. We argued that the ontology should not only store the knowledge needed for the different model views, but also any relevant background knowledge. Documentation of the information that cannot be read off the suite of networks directly is especially important when the development of the suite extends over several years of research and the suite ultimately is handed off to industry. We also attended to the language to be used for our ontologies. The necessity of including all types of relevant knowledge demands a language that has a rich semantics and permits semi-automated model building. We stressed that the language used should be accessible for non-mathematical domain experts. Earlier research had shown that rigorously formal representations, be they logic-based or stated in another mathematical language, cannot readily be understood by domain experts who are not trained in such representations. When stated in a semi-formal language that is accessible for the experts, a comprehensive ontology can provide a means of communication between the knowledge engineers and the experts, which serves to minimize the risk of omitting important information and of including erroneous information.

Next, we argued for aligning the content of the ontology with how practicing experts learn and store knowledge in their minds. Some knowledge, we argued, is stored in a task-neutral fashion, and should also be stored in this way in the ontology. However, we contended that most knowledge of domain experts is inherently related to specific tasks and is stored in that way in their brains. Constructing a task-neutral ontology would thus require stripping the task-specific professional knowledge from its task biases. This, however, is highly demanding, either on the part of the expert or on the part of the knowledge engineer, and error-prone. We therefore proposed storing task-specific knowledge in a task-specific fashion. Lastly, we proposed to develop a library-style ontology, composed of the aforementioned task-neutral and task-specific knowledge modules which subsequently are combined into task-specific ontologies to support concrete task model views for a suite of Bayesian networks. We illustrated the ease of development of multiple views and demonstrated that reuse of information is encouraged by organizing the domain knowledge in modules.

In the near future, we intend to further develop our concept of an ontology library by using it in the development of a suite of Bayesian networks in the field of veterinary science. By doing so, we hope to initiate the distribution of a publicly available collection of ontology modules and inspire the uncertainty community to contribute.

Acknowledgment. Much of the research presented in this paper is based upon our experiences with building single-purpose Bayesian networks with the help of domain experts in different areas of biomedicine; we would like to thank Berthe Aleman, Armin Elbers, Floris Groenendaal, Willie Loeffen and Babs Taal for the valuable discussions about their domains. We would further like to thank Eveline Helsper for sharing her extensive knowledge of ontologies with us and for her excellent work in constructing the oesophageal cancer ontology.

References

[1] H.P.A. Boshuizen, H.G. Schmidt, E.J.F.M Custers, M.W.J. van de Wiel (1995). Knowledge development and restructuring in the domain of medicine: the role of theory and practice. *Learning and Instruction*, vol. 5, pp. 269 – 289.

[2] H.P.A. Boshuizen, H.J.M. Tabachneck-Schijf (1998). Problem solving with multiple representations by multiple and single agents; an analysis of the issues involved. In: M.W. van Someren, P. Reimann, H.P.A. Boshuizen, T. de Jong (eds). *Learning with Multiple Representations*. Amsterdam: Elsevier Science, pp. 137 – 152.

[3] H.P.A. Boshuizen, M.W.J. van de Wiel (1998). One person, multiple re- presentations: An analysis of a simple, realistic multiple representation learning task. In: M.W. van Someren, P. Reimann, H.P.A. Boshuizen, T. de Jong (eds). *Learning with Multiple Representations*. Amsterdam: Elsevier Science, pp. 237 – 263.

[4] T. Bylander, B. Chandrasekaran (1988). Generic tasks for knowledge-based reasoning: the 'right' level of abstraction for knowledge acquisition. In: B.R. Gaines, J.H. Boose (eds). *Knowledge Acquisition for Knowledge-based Systems*, vol. 1. London: Academic Press, pp. 65 – 77.

[5] P.C.G. da Costa, K.B. Laskey, K.J. Laskey (2005). PR-OWL: A Bayesian ontology language for the Semantic Web. *International Semantic Web Conference, Workshop Uncertainty Reasoning for the Semantic Web,* Galway, pp. 23 – 33.

[6] A. Dijksterhuis, L.F. Nordgren (2006). A theory of unconscious thought. *Perspectives of Psychological Science*, vol. 1, pp. 95 – 109.

[7] I.M. van Eekelen, H.P.A. Boshuizen, J.D. Vermunt (2005). Self-regulation in higher education teacher learning. *Higher Education*, vol. 50, pp. 447 – 471.

[8] K.A. Ericsson, H.A. Simon (1993). *Protocol Analysis: Verbal Reports as Data*. Cambridge, MA: MIT Press.

[9] H. Eshach (2007). Bridging in-school and out-of-school learning: formal, non-formal, and informal education. *Journal of Science Education and Technology*, vol. 16, pp. 171 – 190.

[10] L.C. van der Gaag, S. Renooij, C.L.M. Witteman, B.M.P. Aleman, B.G. Taal (2002). Probabilities for a probabilistic network: a case study in oesophageal cancer. *Artificial Intelligence in Medicine*, vol. 25, pp. 123 – 148.

[11] L.C. van der Gaag, E.M. Helsper (2002). Experiences with modelling issues in building probabilistic networks. In: A. Gómez-Pérez, V.R. Benjamins (editors). *Knowledge Engineering and Knowledge Management: Ontologies and the Semantic Web. Proceedings of EKAW 2002*, Lecture Notes in Artificial Intelligence, vol. 2473. Berlin: Springer Verlag, pp. 21 – 26.

[12] Th.R. Gruber (1993). A translation approach to portable ontologies. *Knowledge Acquisition*, vol. 5, pp. 199 – 220.

[13] Th.R. Gruber (1995). Towards principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies*, vol. 43, pp. 907 – 928.

[14] G. van Heijst, A.Th. Schreiber, B.J. Wielinga (1997). Using explicit ontologies in KBS development. *International Journal of Human-Computer Studies* vol. 46, pp. 183 – 292.

[15] E.M. Helsper, L.C. van der Gaag (2002a). A case study in ontologies for probabilistic networks. In: M. Bramer, F. Coenen, A. Preece (eds). *Research and Development in Intelligent Systems XVIII*. London: Springer Verlag, pp. 229 – 242.

[16] E.M. Helsper, L.C. van der Gaag (2002b). Building Bayesian networks through ontologies. In: F. van Harmelen (editor). *Proceedings of the 15th European Conference on Artificial Intelligence*. Amsterdam: IOS Press, pp. 680 – 684.

[17] E.M. Helsper, L.C. van der Gaag (2005). Generic knowledge structures for probabilistic-network engineering. *Proceedings of the Third Bayesian Modeling Applications Workshop*, held in conjunction with the Twenty-first Conference on Uncertainty in Artificial Intelligence, Edinburgh.

[18] E.M. Helsper, L.C. van der Gaag (2007). Ontologies for probabilistic networks: a case study in the oesophageal-cancer domain. *The Knowledge Engineering Review*, vol. 22, pp. 67 – 86.

[19] T.S. Hyde, J.J. Jenkins (1973). Recall for words as a function of semantic, graphic, and syntactic orienting tasks. *Journal of Verbal Learning and Verbal Behavior*, vol. 12, pp. 471 – 480.

[20] F.V. Jensen (2001). *Bayesian Networks and Decision Graphs*. New York: Springer-Verlag.

[21] J. Lave (1988). *Cognition in Practice: Mind, Mathematics, and Culture in Everyday Life*. Cambridge, UK: Cambridge University Press.

[22] V.J. Marsick, K.E. Watkins (2001). Informal and incidental learning. *New Directions for Adult an Continuing Education*, vol. 89, pp. 25 – 35.

[23] A. Ndoye (2003). Experiential learning, self-beliefs and adult performance in Senegal. *International Journal of Lifelong Education*, vol. 22, pp. 353 – 366.

[24] J.W. Payne, A. Samper, J.R. Bettman, M.F. Luce (2008). Boundary conditions on unconscious thought in complex decision making. *Psychological Science*, vol. 19, pp. 1118 – 1123.

[25] G. Schreiber, H. Akkermans, A. Anjewierden, R. de Hoog, N. Shadbolt, W. Van de Velde, B. Wielinga (2000). *Knowledge Engineering and Management: The CommonKADS Methodology*. Cambridge, MA: MIT Press

[26] R. Studer, V.R. Benjamins, D. Fensel (1998). Knowledge engineering: principles and methods. *Data & Knowledge Engineering*, vol. 25, pp. 161–197.

[27] L. Suchman (1988). *Plans and Situated Actions: The Problem of Human/Machine Communication*. Cambridge, UK: Cambridge University Press.

[28] H.J.M. Tabachneck-Schijf, P.L. Geenen (2006). Preventing knowledge transfer errors: probabilistic decision support systems through the users' eyes. In: L.C. van der Gaag, R. Almond (eds). *Proceedings of the 4th Bayesian Modeling Workshop: Bayesian Models Meet Cognition*, Boston, pp. 74 – 82.

[29] H.J.M. Tabachneck-Schijf and H.A. Simon (1996). Alternative representations of instructional material. In: D. Peterson (editor). *Alternative Representations: an Interdisciplinary Theme in Cognitive Science*. London, England: Intellect Books, pp. 28 – 46.

[30] S. Papert, I. Harel (1991). *Constructionism: research reports and essays 1985 - 1990 by the Epistemology and Learning Research Group, the Media Lab, Massachusetts Institute of Technology*. Norwood, NJ: Ablex Publishing Corporation.

[31] M. Uschold, M. Gruninger (1996). Ontologies: principles, methods and applications. *The Knowledge Engineering Review*, vol. 11, pp. 93 – 136.

[32] L.O. Wilson (1997). *New View of Learning: Types of Learning*. Retrieved Jan 25, 2007, from `www.uwsp.edu/education/lwilson/learning`