Case-based Reasoning for Experience-based Collaborative Risk Management

Nielsen L. R. Machado, Luís A. L. Silva, Lisandra M. Fontoura
Programa de Pós-Graduação em Informática
Universidade Federal de Santa Maria – UFSM
Santa Maria, Brazil
{nielsenmachado, silva.luisalvaro, lisandramf}@gmail.com

Abstract - In a collaborative risk management scenario, project stakeholders often need natural forms of recording and reusing past risk management experiences so that they could better assess whether there are threats to the goals of new projects. The contribution of this paper is to propose an enhanced case-based reasoning (CBR) approach to support project participants to exploit such experiences which are here expressed as collaborative risk management discussion cases. The paper shows how these debates are structured through the exploitation of a dialogue game protocol for risk management. Then, it discusses how users can utilize queries based on facts and arguments so that past risk discussion cases could be retrieved from a case base. Attention is also given to case-based explanation templates, which are relevant for the understanding of key moves of argumentation in debate trees recorded in such enhanced cases retrieved. To demonstrate the practical utility of this approach, a case study involving the collaborative experience-based risk management of a software project is discussed.

Keywords-component: Case-Based Reasoning; Collaborative Risk Management Tool; Dialogue Game; Argumentation.

I. INTRODUCTION

Management of risks in new software projects is most effective if one can draw on concrete instances of dealing with risks in past projects. Once such experiences are collected and represented systematically in a reusable memory, project stakeholders involved in collaborative risk discussion tasks have the means of constructing a better assessment of the risks to their project’ goals, considering consequences in case these risks occur and, in general, taking actions to guarantee that the associated threats could be controlled more effectively. The overall goal is to fully exploit these past experiences in order to not only learn relevant lessons from past risk management episodes but also to avoid the repetition of past risk management mistakes in current projects.

Some degree of experience-based risk management is achieved through the exploitation of different Artificial Intelligence (AI) techniques (e.g. [1, 2]). These kinds of approach allow one to learn from data of past software projects so as to better understand the relevance of risks in new projects. Moreover, security risk management methodologies, such as Magerit [3] (in step 2: threads), for instance, discuss that past experiences should be exploited in the identification of threads to the assets of a project. In this context, Case-Based Reasoning (CBR) [4, 5] is an AI problem-solving paradigm which emphasizes the role of past experiences in the solution of future problems. By its nature, CBR is well adapted to the capture and reuse of factual and/or prescriptive information along with a solution for a risk management problem. Moreover, if stakeholders debate or argue about aspects of their project, CBR can easily accommodate any knowledge that these arguments express explicitly.

The initial claim of our approach for risk management is that collaborative debates can be expressed as an argumentation process [6]. Argumentation is understood as a process of dialogue in which different project participants are able to present different kinds of arguments when stating or justifying the assessment of risk management issues in a collaborative risk management discussion. In this dialectical context, we capture and organize such risk discussions in the body of enhanced cases for CBR according to a dialogue protocol, or “dialogue game” [7, 8]. This dialogue game is expressed as a set of moves of argumentation (e.g. locution acts such as propose, ask, inform, etc) that occur when multiple agents (human and/or computational agents) are engaged in a debate. As presented in our initial paper [9], a dialogue game for risk management was designed to mediate the interaction of project stakeholders when these agents develop collaborative risk management tasks. Our approach captures and makes use of a dialogue form (the dialogue, or set of locution acts) which people working on risk management etc. use naturally in their own discussions and debates among themselves, i.e. it respects the kind of knowledge representation that is informally closest to what they actually use in practice.

In this paper, we discuss a CBR extension which approaches not only factual project characteristics but also structural and type details of the arguments observed in past stakeholders’ discussions about how to construct solutions (e.g. risk management plans). We also discuss how project stakeholders can obtain personalized views of the results obtained when different kinds of queries are executed. In particular, the personalization resources proposed here rely on the exploitation of a set of case-based explanation templates, in a scenario in which CBR becomes a form of explanation-based reasoning [10]. In our project, these templates capture meaningful combinations of moves of dialogue, or locution acts, that are exploited in risk management discussions recorded in enhanced
cases. Such explanation templates are relevant for project stakeholders when inspecting and, consequently, forming an understanding of key risk management steps of argumentation advanced in risk debates. We also describe how these case-based query and explanation resources can be exploited by different project stakeholders when a web-based Risk Discussion System (RD System) is used. To illustrate our approach, we describe a case study in which a risk management project is analyzed by different project participants.

This paper is organized as follows: Section II presents some background information about risk management, argumentation and CBR. Section III discusses a) the formation of enhanced cases through facts and arguments, b) the alternative query forms utilized to find past cases so as to support users when deliberating solutions for new risk management problems and c) the exploitation of explanation templates for improving the users’ understanding of risk management discussion cases. Section IV presents a study case to illustrate the utility of our approach. Section V shortly reviews the paper proposals and presents some future steps in our project.

II. CBR, ARGUMENTATION AND EXPLANATION IN THE CONTEXT OF COLLABORATIVE RISK MANAGEMENT

Risk management requires the development of activities for the identification of possible problems and their causes, the analysis of risk probability and impact resulting in the prioritization of more critical risks, the construction of plans for the mitigation or even elimination of prioritized risks and the execution and monitoring of risk management plans [11]. In this context, the CBR paradigm for problem-solving supports the solution of new problems through the systematic collection and representation of cases in a case base, the retrieval of similar cases to a given problem situation, the reuse of past case solutions retrieved in the solution of new problems, the adjustment of these solutions to the context of these new problems when necessary and the retention of experiences of problem-solving in the case base so that the associated software system can improve its problem-solving capabilities [4, 5].

According to Kolodner [5], a case captures a contextualized piece of knowledge representing an experience that teaches a lesson fundamental to achieving the goals of the reasoner. Intuitively, it is possible to observe that “lessons” are likely to be offered by domain users in terms of different kinds of arguments, although such arguments still do not have places in traditional frameworks for CBR. Traditionally, only the key factual characteristics of a problem situation are utilized to index the information content of cases for typical CBR applications. As in other machine learning contexts, these case properties are usually encoded as a vector of attributes and values. Then, when users want to retrieve such cases from a case base, they make use of similarity-based forms of computing the distance between current and past cases [4, 5]. Although similarity could be computationally evaluated in different forms, the weighted Euclidian distance computation is the simplest forms of distance assessment in different CBR applications (see [12] for other forms of encoding case features and computing distances between case-like entities). In a CBR application, the overall idea is to retrieve the most similar cases to a given problem situation by stating a query which represents the current problem, or target case, to be solved. That is because CBR is built on the hypothesis that “similar problems have similar solutions”, which is a hypothesis that can be exploited in different application problems.

In addition to standard factual attributes, cases can also capture arguments and argumentation processes. These kinds of augmented cases are usually found in studies of the nature of argumentation in legal applications of CBR (e.g. [13]), although frameworks involving the use of cases in assisting the generation and evaluation of arguments have been reported in other application scenarios (e.g. [14]). Alternative argumentation models [8] have also been exploited in the development of intelligent systems for supporting the solution of problems in these argumentation applications. In this context, as described in [9], an argumentation process in which a debate is developed among different agents can be organized in the body of a case through the exploitation of a dialogue game model [7, 8]. These dialogue protocols are mainly defined by a set of locution acts, which are typical moves of speech used by these agents. In addition to such set of locution acts, these protocols are described by rules expressing how these locutions can be combined (e.g. which locutions can be used as responses to certain locutions). The interaction of two or more project participants involved in a debate can be mediated by such communication protocols.

According to [6], argumentation and explanation are interlinked activities when one is deliberating a solution for a problem. Among other reasons, they are often combined because certain users lack relevant knowledge to understand arguments being posed in a problem-solving situation, as well as because users are likely to pose arguments which express different considerations and explanations with respect to a decision. In effect, when pro and con arguments of a solution for a case-based problem are presented to users, they can review these arguments in order to form an understanding of the rationale behind certain decisions, as exploited in design problems [15], for instance. In practice, just the exploitation of cases that are similar to a current problem situation is the most effective form of explaining a solution in different application domains. It is also relevant to observe that explanations that are constructed on the grounds of past experiences are likely to be more convincing that standard rule-based explanation, as experimental results described in [16] show. In summary, the overall idea is to collect and record users’ arguments systematically in augmented cases for CBR so that one can exploit the content of these cases to provide additional layers of explanation on top of these argumentative structures.

III. A ENHANCED CBR APPROACH FOR COLLABORATIVE RISK MANAGEMENT

The exploitation of a dialogue game for the development of risk management tasks permits project stakeholders to discuss collaboratively the risks of a software project. In such a debate, project participants can express their opinion about causes and effects of risks, in addition to designing plans so that risks identified can be minimized. Through the exploitation of this dialogue protocol, participants in risk debates can reach a consensus while deciding which actions shall be taken in each risk situation. In this context, the CBR resources of our
RD System permit that these users examine and reuse concrete problem-solving experiences of risk management in the determination of risks in current projects.

A. How enhanced cases are formed when collaborative risk management tasks are developed

Concrete cases of collaborative risk management are recorded in a reusable risk management memory, which takes the form of a case base in the RD System. As it is usual in CBR, we represent the cases of this case base through a list of factual properties. To identify which properties we needed to utilize, we started with the observation that recent software projects are characterized in the agile and planned scenarios. Depending of the context that a project is developed, each one of these methodologies has their strong and weak points, which allows one to realize that contextual factors have a crucial role in the characterization of a project. Authors as Boehm and Turner [17] describe an approach which is based on risks for balancing the utilization of agile methods and planned methods in a project. Among others, attributes such as size, criticality, dynamism, skills of the team and culture are utilized in the characterization of a project. Additional attributes can be utilized so as to distinguish risk management cases obtained when different kinds of companies are considered.

Risk assessment on the localization project

<table>
<thead>
<tr>
<th>Characteristics of the Project:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Team Size: Very Small</td>
</tr>
<tr>
<td>Criticality: Money Loss</td>
</tr>
<tr>
<td>Team Distribution: Collocated</td>
</tr>
<tr>
<td>Rate of Change: Low</td>
</tr>
</tbody>
</table>

Discussion:

Start discussion: Risk assessment of a localization algorithm involving sensors - Programmer 1

Propose risk: The final product will not correspond to client expectations - Programmer 1

Ask: Are the project requirements defined properly? - Programmer 1

Inform: The requirements are okay but the time to develop this project is not enough - Technical Manager

Propose consequence: An unhappy client may try to find other people to develop this project; our competitors for instance - Technical Manager

Argument pro: If the client of this project is not happy with us, we may lose the project funds and the client - Programmer 1

Propose probability: High - Programmer 1

Propose impact: High - Programmer 1

... Propose plan: We need to record the changes of requirements more formally - Programmer 2

Summarize: Requirements that are defined correctly, recording of requirement changes, new meetings in case the client is not happy and the frequent presentation of results to the client - Programmer 1

Propose risk: The timetable of the project is not real - Programmer 1

Propose probability: Medium - Technical Manager

... Propose plan: We should discuss again the project timetable and the project requirements - Programmer 2

Figure 1. An example of a risk management discussion case

In the work of Krutchen [18], a model called Octopus is proposed to characterize projects, where the key factors are: Team Size, Team Distribution, Criticality and Rate of Change. As these contextual factors are important in the area of software processes, they are also relevant for the indexing of enhanced cases. In practice, the risks of a project are dependent on such contextual factors, and the efficacy of actions to prevent them is also influenced by the project context. In addition to factual properties, we make it possible to enhance the body of each risk management case by exploiting argumentation-based characteristics. Integrating CBR and argumentation techniques, these novel case representation characteristics are grounded on the set of arguments that are collected and represented when our dialogue protocol for collaborative risk management [9] is exploited by project stakeholders. To present key locution acts of this dialogue protocol, we can examine a risk discussion case (see Fig. 1) in which different users deliberate risks of a software project. In this argumentation model for capturing arguments in cases, the general format of a single argument is: i) an identification of a locution act from our dialogue protocol, ii) a textual risk management statement (i.e. informal statements of the type that most users in this application domain are able to naturally offer) and iii) an identification of the discussion participant that is advancing such argument. These kinds of argument can be advanced by any discussion participant of a risk management debate which is started with “Propose risk” locutions. These argumentation moves are the root indexing concepts in the risk management discussion tree which is recorded in the body of an enhanced case. To gather information about such risks, a project participant can present question-like statements by using “Ask” locution acts, which can be answered via “Inform” locutions. These inform-like arguments can also be exploited in the presentation of any other contextual information for the development of the debate. Consequences related to the occurrence of risks in a project can also be examined explicitly when our dialogue protocol is used. In doing so, these consequence-like statements are advanced by discussion participants via “Propose consequence” locution acts. At any point of a debate, participants can also pose pro and con arguments in relation to different risk management issues. To promote the critical analysis of a risk management problem situation, our protocol permits participants to advance such arguments through “Argument pro” and “Argument con” locutions. Debate participants can also analyze the probability of a risk occurrence, as well as the impact that a risk is likely to have in a project. When this happens, “Propose probability” and “Propose impact” locution acts can be exploited by project participants. In practice, probability and impact estimates can be presented qualitatively or quantitatively, or both. Once initial steps of risks identification and analysis are completed, risk management plans should be constructed for the most relevant risks, or prioritized risks. In the protocol, risk mitigation, reduction, etc statements, for instance, can be advanced by discussion participants through “Propose plan” locutions. A participant can also summarize different aspects of a risk management discussion, through “Summarize” locutions. In summary, the set of locution acts available in our dialogue protocol for collaborative risk management expedites the debate of different risks in a project and the consequent recording of this discussion as a semi-formal argumentation model in the body of enhanced cases for use in CBR. As described here, this protocol is fully implemented in the RD System.

B. How risk discussion cases are retrieved from the case base so as to support the solution of new problems

The systematic recording of collaborative risk management experiences in a reusable memory is a fundamental issue for
an approach which promotes the exploitation of these experiences in the solution of new problems. However, it is also essential to make available to users alternative forms of consultation for this memory. We make it possible though similarity-based queries which can be formed and executed by discussion participants at any moment of a collaborative risk management debate. Such similarity estimates are determined when properties of the current risk management project, or query, are compared with properties recorded on past cases. Based on the enhanced case characteristics proposed, which are i) facts expressing the context of a risk management project and ii) argumentation moves from the discussion of typical risk management issues, our CBR framework allows users to retrieve the K most similar risk discussion cases to a given query. We utilize a standard “K nearest neighbor” algorithm [4, 5], and a weighted Euclidian distance function in the comparison of cases. So far in our project, an equal weighting scheme has been used despite the fact that the RD System permits the adjustment of such weights by users. For the comparison of argumentation information, our similarity algorithm matches textual statements expressed in a query with sentences recorded in past cases. In the PostgreSQL database [19], which is integrated with the RD System, this similarity assessment permits the identification of natural language arguments matching arguments recorded in past cases. In this context, two types of query methods are relevant for the retrieval of past risk management experiences.

The first query method is standard in CBR applications. In it, a user should input the factual properties of the current project (see Fig. 2 – A). An example of such query statement can involve a project which is very large (“Project Size = Very Large”), with many participants (“Team Size = Large”) that are scattered in different locations (“Team Distribution = Geographic Distribution”). In this project, a problem is that the project participants are uncertain about what risks should be considered (i.e. identified in a relevant list for future analysis) in the project and, consequently, what sort of things could fail in the project development. To solve this problem, discussion participants utilize the project properties mentioned as query parameters. When this query is executed, experiences of collaborative risk management involving projects with similar set of properties are likely to be retrieved from the case base. It means that the most similar collaborative risk management cases retrieved will be shaped by risk proposals (i.e. risk statements advanced by means of “Propose Risk”) that can be examined and, if useful, reused in different forms in the risk management discussion of a current project.

The second query method allows project stakeholders to construct queries that are formed by both factual and argumentation characteristics. In doing so, these users can utilize the list of factual project properties, as described in our first query method, and a set of keywords/sentences along with the corresponding locution acts that were advanced in a discussion. A typical example of such query involves the discussion of a current project that aims to develop a new feature in an existing system. So, the “Business Model” attribute of this project is “System Component”. This system also has a “high rate of changes in its requirements”. Because of this problem, the company responsible for the system maintenance is often

losing money (i.e. the “Criticality” attribute of this project is “Money Loss”). In this project, a participant needs to analyze risks regarding “software requirements that are not clearly described”. To reuse past collaborative risk management experiences in the solution of this problem, such participant constructs a query involving both the “Propose risk” and “Propose plan” locution acts. Along with the locution “Propose risk”, this user inputs the sentence: “unclear specification of software requirements”, expressing the kind of risk that this participant wants to review in similar risk management experiences retrieved from the case base (see Fig. 2 – C). This user also inputs the following keyword in the query statement: “software requirements”. This keyword is described along with the “Propose plan” locution act, expressing that the participant wants to review past risk management plans involving this subject. When this query is executed, previous cases of collaborative risk management are retrieved from the case base. In particular, these cases may contain “Propose risk” and “Propose plan” moves of dialogue in which those keywords/sentences mentioned were advanced by discussion participants. As a result, information regarding risk proposals and risk management plans can be reused in different forms as, for instance, in the construction of new arguments to be advanced in the current debate.

C. How explanation templates can be utilized by users in the understanding of a case retrieved for a given query

In our project, project stakeholders are able to exploit a new form of case-based explanation with the purpose of locating and highlighting relevant argumentation information in past risk management cases retrieved for a given query. Among other reasons, these explanatory views of a query result were designed after the fact that argumentation trees representing collaborative risk management discussions might contain a large number of speech moves, which might hinder a user comparison between current and past cases when analyzing a query result. Therefore, we give users access to alternative template forms in order to supplement their retrieval results and, consequently, make easier the steps of reuse of past arguments in the discussion of new risk management problems. In doing so, such a template format provides a standard for the presentation of selected risk management locution moves available in an enhanced case. This explanation is possible since the argumentative structure of enhanced cases already provides an explicit history, or a sort of narrative explanation [10], for the risk management decision-making process. What a template does, in this case, is to offer users ways of highlighting relevant argumentation steps available in these narratives (see Fig. 2 – B).

An explanation template has a name, a textual description of what the template aims to emphasize in a discussion tree (e.g. what needs to be explained, what interests a user), and how it is done through the selection of a set of locution acts belonging to the dialogue protocol used. A template can be created, recorded and adjusted by a knowledge engineer user of the RD System. In a scenario of a template utilization, for instance, the most similar case for a given query is shown to a user. However, it is presented only through the template contents of certain locution acts, e.g. as selected for its relevance to the user in a suitably personalized consultation with the system.
The template “key risk management tasks”, for instance, locates and highlights the project participants’ arguments advanced for the solution of risk identification, risk analysis and risk response planning problems, while other arguments in a debate are temporarily hidden when a retrieved case is presented to a user. To achieve the template goal, this template structure should be described in terms of locution acts: “Propose risk”, “Propose impact”, “Propose probability” and “Propose plan”, which are the key risk management locutions available in the protocol. As this example shows, the template representation relies on a pre-defined mapping between the description of the template’s goal and the specific set of locution acts that are able to present a solution for such an explanation goal (see Fig. 2 – D). However, if users desire more information about any argument in a retrieved case, independent of whatever personalization has been performed, they can scan through the entire content of the selected case, seeing the complete tree of the discussion that is recorded there. To help with the search for relevant risk management information in this tree, the RD System permits users to filter or expand the nodes or tree branches of these debates.

IV. A CASE STUDY OF COLLABORATIVE EXPERIENCE-BASED RISK MANAGEMENT

This illustrative case study involves a software project aiming to develop a web-based system for supporting a selected set of company managers to obtain the geographic location of certain employees of a company. The main problem of this project is that the software requirements are not described comprehensively since the project’s clients are not sure about the resources that this system would need to have for the privacy of the company employees not to be violated in particular situations. No experience with such a system was available there either among the project’s clients or the development team involved in the construction of this system. It means that the reuse of past risk management experiences amounts to a form of identifying risks that these participants would not be aware in such a new kind of project for this team. This is one of the reasons that we choose to discuss this case study. The contextual properties of this project are: “c1: Team Size = Very Small”, “c2: Team Distribution = In House”, “c3: Criticality = Comfort Loss” and “c4: Rate of Change = High”.

In our work, we describe clear and intuitive resources for participants of a collaborative risk discussion (e.g. a project manager, a technical leader) to exploit past experiences in the enhancement of risk management tasks of a new project. In this case study, we show how our extended CBR approach can support the development of identification, analysis and planning tasks of risk management. We show how both queries based on facts and queries based on facts and arguments can be utilized by project stakeholders. With cases retrieved from the case base as a result of such queries, explanation templates are utilized to facilitate the comparison between the target problem and the most similar cases retrieved and, consequently, to improve the reuse of past experiences.

A first task in the risk management of the target project is the identification and analysis of threats that are likely to occur. By using our approach, a project manager can analyze what kind of risks can occur in past projects that contain factual characteristics (c1 – c4) that are similar to the ones described in the target project. To assess the most similar cases retrieved, the “key risk management tasks” explanation template can be utilized due to the fact that this template highlights essential risk management arguments in a debate. Two substantially relevant past cases, with similarity measures 90.1% (‘p1’) and 79.5% (‘p2’), are retrieved. Considering only p1
because of its high rating, the risks thus identified are: “p1.r1: The final product will not correspond to client expectations”, “p1.r2: The algorithms that were defined may not be adequate to the functional requirements of the system”, “p1.r3: There is a difficulty which is the construction of realistic scenarios and collection of testing data” and “p1.r4: The client of the project and the members of the development team have different views about the same requirements”. Of the four risks, the project manager determined that only r2 did not apply to the new project under consideration. The effectiveness of plans for the resolution of risks is a central topic in risk management. In this case, it is important to observe that management plans that were successful in past projects can be reused in new projects. To construct new plans, project participants can make a new kind of CBR query in the RD System. In it, both contextual project characteristics represented as factual properties and keywords or sentences occurring in the argumentative analysis of prioritized risks can be utilized. In this case study, the technical leader of the project stated such query by using both i) the characteristics c1-c4 and ii) the locution act “Propose risk” along with the keywords “s1: requirements” and “s2: client expectations”, as well as the locution act “Propose plan” with “s3: testing and experience”. The explanation template that was selected by this user was the “Risk and plan proposals” since it emphasizes risks and treatment plans, which is the information that the technical leader is looking for. The retrieval that followed brings up a most similar (86.3%) case p3. Its associated management plan proposals are: “p3.mp1: We need to record the changes of requirements more formally”, “p3.mp2: The requirements should be documented properly”; “p3.mp3: To use small examples in order to guide the test of the system”; “p3.mp4: We can have some training before the project starts” and “p3.mp5: We should get support from the technical leader of this project”. The technical leader accepted the immediate relevance of items 1, 2 and 4. In addition, mp3 suggested a consideration specially adapted to the conditions of the new project under analysis: “small examples could be developed and linked to the use cases of the project, as well as the test cases of the system.”

V. CONCLUDING REMARKS

Collaboration and the reuse of experiences from past projects are crucial needs for users which aim to achieve effective risk management. Project stakeholders’ experience is naturally offered when these users advance different arguments in the collaborative debate of a project’s risks. Experience is also systematically captured when concrete risk management cases of problem-solving are shaped by not only factual information offered when these users advance different arguments in the systematic capture when concrete risk management cases of the project, as well as the test cases of the system.”

Future work involves giving the world access to (examples of) the RD System via the web, in addition to the development of evaluation experiments in collaborative risk management scenarios so that one could investigate forms of applying our approach in practice. We also plan to develop new explanation techniques for CBR and Argumentation in other to further improve the explanation capabilities of our CBR approach for experience-based collaborative risk management.

REFERENCES


267