Distributed Software Interactive Behavior Analysis Based on Knowledge Fusion

Junfeng Man\textsuperscript{1,2}, Xiangbing Wen\textsuperscript{1}, Zhibing Wang\textsuperscript{1}, and Guangbin Liu\textsuperscript{1}

\textsuperscript{1}College of Computer and Communication, Hunan University of Technology, Zhuzhou, China
Email: mjfok@tom.com
\textsuperscript{2}College of Information Science and Technology, Central South University, Changsha, China

Abstract—A novel online analysis method for software Interactive Behavior (IBs) is presented in the paper. Scenario-sensitive method is adopted to model complicated IBs among third party software entities. By fusing real-time self-experience and pervious experience based on knowledge, the creditability of interactive entities is computed automatically. Multi-Entity Bayesian Network (MEBN) tool is adopted to construct reusable domain “knowledge fragments”. If current scenario is similar to pervious one, then pervious one is reused; if there is no similar scenario, evidences gained from monitoring and pervious experience are fused to construct behavior model for this scenario. The combination of large and small granularity knowledge reuse improves analysis efficiency of IBs.

Index Terms—multi-entity Bayesian network, interactive behavior analysis, scenario, knowledge fusion

I. INTRODUCTION

In open and dynamic network environment, distributed software is loosely aggregated with several heterogeneous entities. Entity elements may enter and leave dynamically, these elements may interconnect, intercommunicate, collaborate and unite each other in terms with variously cooperative work way. Currently, it is still confronted with many challenges for the analysis of IBs of distributed software.

(1) How is monitoring information fused with historical data to provide evidence for the analysis of IBs. In open and dynamic distributed software environment, the data monitored from multi-source may be represented with many forms. It is unpractical that information fusion is implemented by simple syntax matching. Advancing information processing from data to knowledge level, the realization is growing that sharing knowledge among distinct information systems requires first arriving at a common understanding of their respective semantics, and then formalizing that semantics in computable representations. Thus, computer can analyze and reason about IBs, proactively predict subsequent possible trend.

(2) In open and dynamic distributed software environment, how to solve the problem of uncertain knowledge between complicated interactive entities and their relationships. Uncertainty is ubiquitous to knowledge fusion. Almost any source of primary data carries some degree of uncertainty. Bayesian probability is a principled formalism for representing uncertainty and drawing inferences in the presence of uncertainty. Specifically, in a standard Bayesian Network (BN), all the hypotheses and relationships are fixed in advance, and only the evidence can vary from problem to problem. In open and dynamic distributed software environment, loosely coupled interactive entities may be strange for each other, or the entities which ever have interacted may have new interaction in new scenario, numbers of interacting entities cannot be known in advance. Standard BN cannot flexibly represent complexity and uncertainty of interactive entity behavior.

Because loosely-coupled entities in distributed software have their own profits, behavior strategies and rules, their running time behaviors have inherent laws, the collaboration of interactive entities makes them show some statistical characteristic in the mass at running time. The scenario and relationships between behavior and behavioral effects at running time are investigated, and Multi-Entity Bayesian Network (MEBN) is adopted to analyze running time behavior states and traces, behavior analyzing and predicting model is constructed, the intentions of interactive entities are inferred, and subsequent possible trend is proactively predicted.

The rest of this paper is organized as follows: Section 2 illustrates online analysis technology of IBs; In Section 3, above method and technology are used to “Trusted Purchasing Network” that we develop, deceitful or fraudulent behaviors in the process of trade are online analyzed, experiment results and corresponding analysis testify our theory.

II. INTERACTIVE BEHAVIOR ONLINE ANALYSIS

In software creditability computing, the creditability of interactive entities is computed automatically by fusing real-time self-experience and pervious experience based on knowledge, which can objectively and instantaneously testify our theory.

A. Multi-Entity Bayesian Network

The W3C responded to the limitation with the recently created Uncertainty Reasoning for the World Wide Web Incubator group (URW3-XG)[1]. The use of probabilistic
reasoning enables information systems to derive benefit from uncertain, incomplete information, instead of being restricted to complete knowledge alone. One of the most promising approaches to deal with uncertainty in the SW is BNs, a graphical, flexible means of parsimoniously expressing joint probability distributions over many interrelated hypotheses. However, BNs have some limitations on representational power that restricts their use for the SW. Amongst these limitations are the fact that the number of variables has to be known in advance and the technique’s lack of support for recursion. In order to address these shortcomings within the context of the SW, Costa proposed a Bayesian framework to probabilistic ontologies that provides a basis for representation and reasoning under uncertainty with the expressiveness required by SW applications [2]. This framework is based on the probabilistic ontology language PR-OWL, which uses MEBN [3] as its underlying logic. MEBN is a formalism that brings together the expressiveness of First-Order Logic (FOL) with BN’s ability to perform plausible reasoning.

![Knowledge fusion based on semantic knowledge](image)

**Figure 1.** The analyzing process of IBs

MEBN is a first-order Bayesian logic that integrates classical first-order logic with probability theory. MEBN represents the world as comprised of entities that have attributes and are related to other entities. Knowledge about the attributes of entities and their relationships with each other is represented as a collection of MEBN fragments (MFrags) organized into MEBN Theories (MTheories). MFragment consists of both a set of CPTs and FOL logical constraints that establish their validating conditions. The number of random variables (RV) is not fixed in a MEBN model. Instead, RVs are instantiated dynamically. An MTheory is a set of MFrags that satisfy certain FOL consistency conditions that guarantee the existence of a unique joint probabilistic distribution (JPD) under its RVs. When all RVs are instantiated, all consistency conditions are satisfied, and all CPTs are generated, the MEBN yields a Scenario Specific Bayesian Network (SSBN). An SSBN is a normal BN. SSBN is stored into repository, which may be reused in similar scenario. Thus we implement two-stage knowledge reuse, MFrags are reused in constructing SSBN, SSBN is reused in analyzing similar scenario. This is a very important feature of the logic for modeling complex, intricate scenario.

“Trusted Purchasing Network” online business system is composed of 14 MFrags (figure 2). Each of these eleven MFrags represents the probability information about a group of their respective random variables. Collectively, the group implicitly expresses a JPD (Joint Distribution Probability) over truth-values of sets of FOL sentences. That is, probability distributions are specified locally over small groups of hypotheses and composed into globally consistent probability distributions over sets of hypotheses. MEBN theories extend ordinary Bayesian networks to provide an inner structure for random variables. Random variables in MEBN theories take arguments that refer to entities in the domain of application. This is because an MFragment is just a template, in other words, it does not represent individuals RVs, but a class of RVs. The values of its states appear only when the MFragment is instantiated.

![Figure 2. The MTheory of “Trusted Purchasing Network” online business system](image)
satisfied), and Absurd (a condition expression does not make sense). Input nodes are variables that influence the probabilistic distribution of its child resident nodes, but their distributions are defined within their own MFrags. In other words, in a complete MTheory, every input node must be a resident node in another Mfrag, where its probabilistic distribution will be defined. Resident nodes have the local probabilistic distributions defined in that Mfrag, including the probabilistic dependence on its parent values (that can be input or resident nodes). A node can have a list of arguments in parenthesis, which are replaced by unique identifiers of domain entities when the net is instantiated.

Another advantage of MEBN is to support recursion when constructing BNs. Its obvious difference with dynamic BNs is to execute recursion for part nodes, which decreases complexity of constructing BNs and improves inference efficiency. Figure 3 is the Mfrag AddtoCart operation, whose the second parameter is orderable, i.e., AddtoCarts(OR123, T1) is computed on condition that AddtoCarts(OR123, T0) is known.

B. Decision-making and inference

Multi-Entity Decision Graphs (MEDGs) extend MEBN logic to support decision making under uncertainty. MEDGs are related to MEBNs in the same way influence diagrams are related to BNs. A MEDG can be applied to any problem that involves optimal choice from a set of alternatives subject to given constraints. When a decision Mfrag (i.e. one that has decision and utility nodes) is added to a generative MTheory, the result is a MEDG.

The MTheory depicted in Figure 2 is a generative MTheory, which provides prior knowledge that can be updated upon receipt of evidence represented as finding MFrags. In a BN model, assessing the impact of new evidence involves conditioning on the values of evidence nodes and applying a belief propagation algorithm. When the algorithm finishes, beliefs of all nodes, including the node(s) of interest, reflect the impact of all evidence entered thus far. This process of entering evidence, updating beliefs, and inspecting the posterior beliefs of one or more nodes of interest is called belief propagation. Usually, the belief propagation process is carrying on answering probabilistic queries. Whereas BNs are static models that must be changed whenever the situation changes (e.g. number of buyers, time recursion, etc.), an MTheory implicitly represents an infinity of possible scenarios. Figure 4 illustrates the scenario that two users(users1 and user2) earn reputation for user3 by deceitful trade, thick arrows represent the process of decision-making.

MEBN inference begins when a query is posed to assess the degree of belief in a target random variable given a set of evidence random variables. It is started with a generative MTheory, add a set of finding MFrags representing problem-specific information, and specify the target nodes for the query. The first step in MEBN inference is to construct the SSBN, which can be seen as an ordinary BN constructed by creating and combining instances of the MFrags in the generative MTheory. Next, a standard BN inference algorithm is applied. Finally, the answer to the query is obtained by inspecting the posterior probabilities of the target nodes.

III. DECEITFUL OR FRAUDULENT BEHAVIORS ANALYSIS

A. Deceitful analyzing of AddtoCarts behavior

AddtoCarts IB is composed of several AddtoCarts events. For ‘PriceReasonable’ of every added goods, ‘ConsumReasonable’ of this user, the creditability of AddtoCarts IB is computed. Formula (1) is used to compute ‘PriceReasonable’, formula (2) is used to compute the creditability of AddtoCarts IB after an AddtoCarts event. The process of computing posterior probability of the creditability of IBs with SSBN is the process of fusing real-time self-experience and pervious experience based on knowledge.

Figure 3. AddtoCarts SSBN with recursion

Figure 4. The MEDGs of earning reputation by deceitful trade

\[
\text{PriceReasonable} = 1 - \min\left(\frac{\text{CurrentPrice} - \text{GeneralPrice}}{\text{GeneralPrice}}, 1\right) \\
\text{AddtoCarts}(t) = \text{AddtoCarts}(t-1) \times W_1 + \text{PriceReasonable} \times W_2 + \text{ConsumReasonable} \times W_3 \\
W_1 + W_2 + W_3 = 1
\]

We analyze four types of creditability of AddtoCarts IBs. User1 has good previous reputation, he has an honest trade this time; User2 has bad previous reputation, he has a deceitful trade this time; User3 has good previous reputation, he has a deceitful trade this time; User4 has bad previous reputation, he has an honest trade this time. Trusted threshold is set to 0.5. From figure 5, we can see that this online analyzing method of IBs can accurately and instantaneously identify deceitful or fraudulent behaviors.

\[\text{AddtoBlackList} = \begin{cases} 
\text{ForbidTrade} & \text{if CollusionAnalysis} \leq 0.3 \\
\text{AddtoBlackList} & \text{if CollusionAnalysis} > 0.3 \\
\text{ContinueMonitor} & \text{if CollusionAnalysis} \geq 0.7
\end{cases} \]

Figure 6 shows the collusion analysis for four kinds of trade process. At first, honest values of Type1 and Type3 are greater than threshold (0.5), there is no collusion to earn reputation, honest values of Type2 and Type4 are less than threshold, there are possibly deceitful behaviors of collusion. The system will send early alarm. With the increase of trade number, posterior probability of collusion analysis is also constantly changing, if their values are less than threshold, the system will send early alarm. The system makes decision according to collusion analysis results. There three kinds of strategies: ContinueMonitor, AddtoBlackList and ForbidTrade. This method can execute real-time analysis for collusion in the process of trade, analyzing results guide system to take appropriate measure, which guarantees trade to be secure and reliable by the greatest extent.

VI. CONCLUSION

Open and dynamic distributed software system loosely aggregates several heterogeneous entities. Entity elements may enter and leave dynamically, their IBs are complicated and changeful. How to monitor and analyze IBs of distributed software is a very important scientific problem that has academic meaning and application value. Using MEBN tool, IBs analysis efficiency is improved by reusing large and small granularity knowledge. In subsequent research, we will improve the efficiency of SSBN construction and query; continue consummating formalization representation of behavior; pay more attention to solve the problem of inconsistent knowledge in the process of behavior analysis and inference; constantly enrich rules and repository to implement unsupervised analysis for IBs.

ACKNOWLEDGMENT

The financial supports from the national natural science fund of China under the grant No. 60773110, post-doctoral science fund of China under the grant No. 20080440216, natural science fund of Hunan province under the grant No. 09JJ6087 is gratefully acknowledged.

REFERENCES

