Intelligent Process for handling unexpected situation using monte- carlo sampling technique

K.Jagadeesh[1], Dr. S. K. Muthusundar[2]

[1]Research scholar, Meenakshi Academy of Higher Education and Research, K.K.Nagar, Chennai, TamiNadu, INDIA.

[2]Professor, [2]Dept. of CSE, Sri Muthukumaran Institution of Tech., Chennai-69.

Email:skmuthusundar@gmail.com

Email: jaganbheema@gmail.com

A b s t r a c t

Keywords:

Technique-based reasoning with unexpected information, Distributed systems, Complex action processing.

Conventional Technique-based professional systems use human professional knowledge to solve real-world issues that normally would need human intelligence. Professional knowledge is often represented in the type of techniques or as information within the computer. Our growing world require for action driven or active systems that react automatically to actions. In here they offered an efficient mechanism for action derivation under unexpectedly. In additional method is Selectable, which performs in to an important role in action derivation in both the deterministic and the unexpected settings. A model for representing derived actions was introduced together with a Monte Carlo sampling algorithm that approximates the derived action probabilities. This enhancement implemented in to the prioritization algorithm. In this Prioritization algorithms identify such cases in which the order of situation detection is undetermined and defines a mechanism for the definition of a deterministic detection execution.

1. INTRODUCTION

There is a growing require for systems that react automatically to actions, some actions are externally and deliver information across distributed systems, others require to be derived by the method itself based on available information. The earliest action-driven systems in the database realm impacted both industry (triggers) and academia (view materialization). New applications in areas such as Business Method Management, sensor networks, security applications (e.g., bio hazards) engineering applications (e.g., forecasting networked resources availability); and scientific applications (e.g., utilization of grid resources) all need sophisticated mechanisms to manage and react to actions.

Way of managing the gap between actual actions and complex action [1] notifications is to explicitly handle unexpectedly. This might be completed by modeling actions unexpectedly as a probability associated with each action, whether such actions are generated externally or derived. However, a major challenge in such explicit management of actions unexpectedly is that technique-based systems require to method multiple technique with multiple action sources. Correctly calculate the action probabilities while taking in to account various types of unexpectedly is not trivial. Clearly, correct quantification of the probability of derived actions serves as an important device for decision making. Action generation under unexpectedly ought to therefore be accompanied with an appropriate mechanism for probability computation. Present a solution of both issues, introducing a novel generic and formal mechanism and framework for managing action derivation under unexpectedly [2].

2. EXPERIMENTAL PROCEDURES DESCRIPTION

The new technique has been processed in to the action driven (or active) systems, i.e., systems that react automatically to actions. Proposed the current complex action processing [4] literature on manage knowledge under unexpectedly to permit unexpected derivation of actions. Present an efficient mechanism for action derivation under unexpectedly. A model for representing derived actions is introduced together with a Monte Carlo sampling algorithm that approximates the derived action probabilities. Experiment with the sampling algorithm, showing it to be comparable to the performance of a deterministic action composition technique [3].

2.1. Action-Driven Systems Method

In this module performed the method of action driven technique. An action-driven technique is a technique of objects, which interact with each other using a message-passing mechanism. Some actions are generated externally and deliver knowledge across distributed systems [10] others require to be derived by the technique itself based on available information. Action driven systems is more response to unpredictable situations.

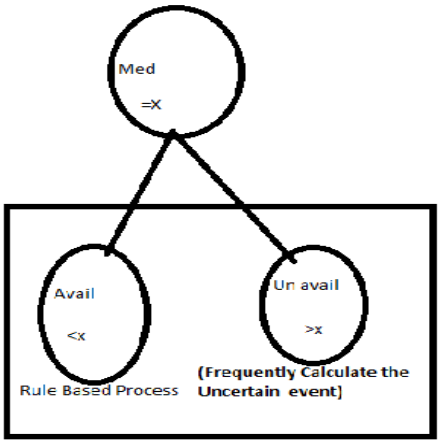
2.2 Set of Service, Metadata and Technique Based Method

This module they execute in to the method of set of techniques based method, which is also performs along Situation Manager Technique Language. In this efficient runtime detection execution [5] mechanism to reducing the complexity of active applications [6] also performed in to the ability of action-driven systems [7] to exactly generate actions. Technique-Based systems are used as a way to store and manipulate knowledge to interpret information in a useful way.

2.3 Probabilistic Action Model

In this module they performed in to the method of probabilistic action model level [8]. In here they execute in to automatically construct a Bayesian network from a set of actions and techniques, following the Knowledge Based Model Construction (KBMC). Also these techniques are defines how lots of new actions ought to be derived and helps to calculate their attributes and probabilities [9]. In addition support of the calculation of probabilities associated with derived actions.

2.4 Efficient and Correct Mechanism



Processed by Unexpected Actions in this module they perform in to the method of efficient and correct mechanism, which is improving derivation efficiency they employ Mechanism: Selectability, Sampling method over a set of techniques. Selectability is a significantly influence the performance of the classification technique.

3. DECISION TREE

Also the model representing derived actions was introduced together with a Monte Carlo sampling algorithm that approximates the derived action probabilities, and then the sampling algorithm provides a correct estimation of probabilities.

3.1 Prioritization Algorithms

Prioritization algorithms identify cases in which the order of situation detection is undetermined and defines a mechanism for the definition of a deterministic detection execution. These cases happen when actions occur simultaneously or an action has multiple roles in situation. In these Prioritization algorithms ought to identify such cases and recommend solution strategy. Also the Situation detection is performed with respect to the action occurrence time and not the time in which the action is detect by the technique [11].

Classification technique provide a Decision tree. The Decision tree is used to often capture the generated actions. In this relevant algorithms are calculating the Selectable of EIDs in the unexpected actions. Also it performs in to an important role in action derivation, in both the deterministic and the unexpected settings. Computing Selectability: The unexpected setting derivation is carried out on EIDs, algorithms are necessary to compute which EIDs, from a given technique action history H, whether an EID E are selectable by technique r, may by itself, incur significant computational work [13]. According to selectable depends on the feasible action histories in which the action corresponding to E participates. Therefore, a naive algorithm for locating all selectable EIDs involves scanning all action histories, and for each action history, finding all actions selectable by technique r using the function sr. However, this may be inefficient, as the number of feasible action histories may be exponential in the size of the state space of the actions. Therefore, if the technique action history H contains n EIDs, and largest EID size of the state space is m, then the number of feasible action histories (and thus the complexity) is O(mn).

3.2 Sampling Method over A Set Of Techniques

This Sampling method provide in to the high level statement of unexpected action derivation [14]. Also here they used in to the Monte Carlo sampling algorithm, it performs in to a sample for the explicit actions is generated using the mutual independence assumption and the derivation according to each technique is based on the probabilistic technique definition. The performance measure used in all experiments is the action processing rate per second, calculated as

Number of generated actions
----------Divided By----------
Total processing time

3.3 Action Generation under Unexpectedly

In this module performs in to the development of action generation under unexpectedly. In here they processed in to the forming value of action probability of various actions [12], this actions are performed in to externally or derived. Also they manage in to the explicit management of actions

unexpectedly moreover here the technique-based systems require to method multiple techniques with multiple action sources. In addition here the correct quantification of the probability of derived actions serves as an important device for decision making in this unexpectedly [15].

4. CONCLUSION

Provide empirical proof demonstrating the scalability & accuracy of approach. Enhancement execute in to the method of prioritization algorithm. In this Prioritization algorithms identify cases in which the order of situation detection is undetermined & defines a mechanism for the definition of a deterministic detection execution. These cases happen when actions occur simultaneously or an action has multiple roles in situation. In these Prioritization algorithms ought to identify such cases & recommend solution strategy. Also the Situation detection is performed with respect to the action occurrence time & not the time in which the action is detect by the method. A model for representing derived actions was introduced together with a Monte Carlo sampling algorithm.

REFERENCES

[1].Gal.A, Turchin.Y, Wasserkrug.S(2015), "Tuning Complex Action Processing Techniques Using the Prediction-Correction Paradigm," Proc. Third ACM Int'l Conf. Distributed Action-Based Systems (DEBS '09).

[2].Etzion.O, Gal.A, Wasserkrug.S, A Taxonomy(2013), "Representation of Sources of Unexpectedly in Active Systems," Proc. Sixth Int'l Conf. Next Generation Information Technologies and Systems (NGITS '06), pp. 174-185.

[3].Etzion.O, Gal.A, Wasserkrug.S(2005), "A Model for Reasoning with Unexpected Techniques in Action Composition Systems," Proc. 21st Conf. Unexpectedly in Artificial Intelligence (UAI '05), pp. 599-606.

[4].Etzion.O, Gal.A, Wasserkrug.S(2008), and Y. Turchin, "Complex Action Processing over Unexpected Data," Proc. Second Int'l Conf. Distributed Action-Based Systems (DEBS '08), pp. 253-264.

[5].Adi.A(2003), "A Language and an Execution Model for the Detection of Reactive Situations," PhD thesis, Technion—Israel Inst. Of Technology.

[6].Adi.A and Etzion.O(2004), "Amit—The Situation Manager," Int'l J. Very Large Data Bases, vol. 13, no. 5, pp. 177-203.

[7].Ammon.R, Emmersberger.C Greiner.T, Paschke.A, Springer.F Wolff.C(2008), "Action-Driven Business Process Management," Proc. Second Int'l Conf. Distributed Action-Based Systems (DEBS '08), July.

[8].Balazinska.M, Khoussainova.N, Suciu.D(2008), "PEEX: Extracting Probabilistic Actions from Rfid Data," Proc. Int'l Conf. Data Eng. (ICDE).

[9].Beeri`A, Eyal.A, Milo.T, Pilberg.A(2007),"Query-Based Monitoring of BPEL Business Processes," Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 1122-1124.

[10].Alencar.P, Blanco.R, Wang.J(2008), "A Metamodel for Distributed Action Based Systems," Proc. Second Int'l Conf. Distributed Action- Based Systems (DEBS '08), pp. 221-232.

[11].Bauer.M, Rothermel.K(2005), "An Architecture for Observing Physical World Actions," Proc. 11th Int'l Conf. Parallel and Distributed Systems (ICPADS '05), pp. 377-383, July.

[12].Breese. J.S, Goldman.R.P, WellmanM.P(1994), "Introduction to the Special Section on Knowledge-Based Construction of Probabilistic and Decision Models," IEEE Trans. Systems, Man and Cybernetics, vol. 24, no. 11, pp. 1577-1579, Nov.

[13].Aggarwal.C, Campbell.M, Li.C.S, Naphade.M, Wu.K.L, and Zhang.T(2004), "An Evaluation of over-the-Counter Medication Sales for Syndromic Surveillance," Proc. IEEE Int'l Conf. Data Mining—Life Sciences Data Mining Workshop.

[14].Chakravarthy.S, Mishra. D(1994), "Snoop: An Expressive Action Specification Language for Active Databases," Data and Knowledge Eng., vol. 14, no. 1, pp. 1-26.

[15].Chu.A, Deshpande.A, Hellerstein.J, and Hong.W(2006), "Approximate Data Collection in Sensor Networks Using Probabilistic Models," Proc. 22nd Int'l Conf. Data Eng. (ICDE), p.48.