Models and Issues on Probabilistic Data Streams with Bayesian Networks

Hideyuki Kawashima
Graduate School of Systems
and Information Engineering
University of Tsukuba
Tennodai 1–1–1, Tsukuba,
Ibaraki, 305–8573 Japan
kawasima@cs.tsukuba.ac.jp

Ryo Sato
College of Information
Sciences, University of Tsukuba
Tennodai 1–1–1, Tsukuba,
Ibaraki, 305–8573 Japan
punisiro@kde.cs.tsukuba.ac.jp

Hiroyuki Kitagawa
Graduate School of Systems
and Information Engineering
University of Tsukuba
Tennodai 1–1–1, Tsukuba,
Ibaraki, 305–8573 Japan
kitagawa@cs.tsukuba.ac.jp

Abstract

This paper proposes the integration of probabilistic data streams and relational database by using Bayesian networks that is one of the most famous techniques for expressing uncertain contexts. A Baysian network is expressed by the graphical model while relational data are expressed by relation. To integrate them we make the relational model as the unified model for its simplicity. A Bayesian network is modeled as an abstract data type in an object relational database, and we define signatures to extract a probabilistic relation from a Bayesian network. We provide a scheme to integrate a probabilistic relation and normal relations. To allow continual queries over streams for a Bayesian network, we introduce a new concept, lifespan.

1 Introduction

The progress of sensor devices is dramatically rapid. It includes a variety of sensor devices such as network cameras, wireless sensor nodes and RFID readers. These devices generate massive sensor data streams, and many stream processing engines (SPE) have been developed to process them [1, 2, 7]. Operations on these SPEs are relational operators (selection, projection, join, aggregation, etc) and domain specific operators such as FFT[7] or Kleene Plus[6].

Simply stated, all the usual operators reduces the amount of data or translate data expression. They do not increase the amount of data. Our question is, How can we enrich stream data processing if the amount of data can be increased ? When a SPE receives an event, possible events can be reasoned from it, and the result of reasoning can be used for the estimation or anticipation of the physical world. We believe that by processing the result of reasoning by using conventional data processing techniques, a sophisticated reasoning system can be presented.

There are many kinds of reasoning techniques such as Bayesian network, hidden Markov model, or Kalman filter. As a first step, we focus on Bayesian network which is widely accepted[11].in this research. In the rest of this paper, A Bayesian Network is notated as BN. A BN has a graph structure and each node in the graph can express an event as a random variable. And causations between each event are expressed with arrows. When an event occurrence is arrived on a BN, the probability of the event is turned to 100%, and then on the all other nodes, probabilities are updated through a probability propagation procedure[1] A BN has many applications and its examples are context estimation[9] and spam filter[8].

When applying a BN for event streams generated from sensor data streams by using an event detector, the following two problems arise.

(P1) Lack of Data Processing: Usually the result of reasoning on a BN is integrated with other data, and then they are often selected, projected, joined, and aggregated to understand the contexts of the physical world. However, unfortunately, the processing are separately conducted on each application. One of the reasons is any integration infrastructures are not proposed yet.

(P2) Lack of Multiple Event Detection Concept on BN: A data stream has an infinite length of tuples and it continually arrives a system[4]. Thus in case a data stream is used for a BN, the BN should continually detect an event occurrence and conduct a probability propagation procedure. However, in the physical world, often multiple events occur at the same time because the physical world has a spatial spread and events occur in parallel. Thus, a mechanism to detect multiple events occurred in the same time, should be incorporated into a BN when a BN processes an event

[1] For probability propagation procedures, please see [10]

978-0-7695-3297-4/08 $25.00 © 2008 IEEE
DOI 10.1109/SAINT.2008.108

stream. Unfortunately, this mechanism is not developed to the best of our knowledge.

Then, this paper presents the following two contributions to cope with the above two problems:

(C1) Integration of a BN model and a Relational Model: As the first contribution, this paper proposes the object relational data model to take the BN into the relational data model. This integrates probabilistic data streams and relational data, and it enables to manipulate them uniformly and declaratively though SQL based language.

(C2) Lifespan: As the second contribution, this paper introduces the concept of occurring duration for each event in a BN. For each event, a user should set a lifespan which expresses the duration of occurrence of the event. After detecting the occurrence of an event, the event is regarded to be occurring until its lifespan passes. And it is turned not to be occurring after the lifespan. When a query for BN is evaluated, the occurrences of events in it are decided by the lifespans. Though this is intuitive and simple extension, nobody has paid attention on it. A temporal Bayesian network[3] is related to this concept, it is not the answer to this problem. The detail is described in Section 3.

The rest of this paper is organized as follows. Section 2 presents two contributions for the two problems. They are integration of BN with relational database, and incorporation of lifespan for a BN. Section 3 describes related work. Finally Section 4 concludes this paper and indicates future work issues.

2 Models and Issues

This section proposes approaches for two problems. Approaches for (P1) and (P2) are proposed in Section 2.1.2 and Section 2.2 respectively after reviewing BN.

2.1 Integration of Data Models

2.1.1 Review of BN

A BN is widely used to express uncertain events[11]. It has a directed acyclic graphical (DAG) structure and each node expresses an event by a random variable. Events are connected based on causal relationships. Each node has conditional probability table (CPT) and it is used update propagation of probabilities when event occurrences are detected.

When an event occurs, the event's probability is set to 1.0, and the change of probability is propagated to other nodes. To execute update propagation, two types of algorithms are widely known. They are strict reasoning algorithm and approximation reasoning algorithm that accelerates processing by sacrificing accuracy. This paper focuses on strict reasoning method referred to as "Pearl's Message Passing Algorithm"[10]. The algorithm spreads messages in all the nodes when an event occurs.

2.1.2 Abstract Data Type for BN

This section proposes an approach for **(P1) Lack of Data Processing on Bayesian Networks**.

Since BN has a graphical model which is different from relational data model, the data cannot be directly processed in relational data model. To integrate BN and relational database, this paper adopts an object relational data model of which object stores BN. The object is an abstract data type (ADT) and it is denoted as BN-ADT. Though BN-ADT manages BN in a relational database, the data managed in BN-ADT and data in relational database cannot be integrated because of different data models. To cope with it, this paper introduces signatures that extract data from BN-ADT and pack it into tuples. Since tuples are residents in relational database, then data from BN and from relational database are integrated.

Signatures Though we have many signatures, we introduce only two primitive signatures here because of space limitation.

getNode(Id or Ev or Pr, Sign, Value): The getNode signature searches nodes which match search conditions and returns them with context identifier which identifies the BN object searched by getNode. The getNode require three search conditions. **Id** denotes the identifier of a node. **Ev** denotes the event name of a node. **Pr** denotes probability of a node. First, as a search condition, either identifier or event name or probability should be chosen.

Second, a sign should be chosen from $\{<, \leq, =, !=, >, \geq\}$. Third, value should be given.

makeTuple(): The makeTuple signature extracts information from BN and pack it into tuple. That is, it translates data from BN to relation. Each generated tuple should have four attributes and two of them are from BN, and one is from a tuple which includes the BN. The three from BN are **Id**, **Ev** and **Pr**. The one from a tuple is: a primary key which denotes the identifier of the tuple which includes the BN.

Other Signature: Other signature includes **child**, **descendant**, **parent**, **ancestor**, **sibling**. The details are omitted because of space limitation.

2.2 Lifespan

This section proposes an approach for **(P2) Lack of Window Concept on BN**.

As described in Section 1, usual BN does not have the concept of continual event stream processing and thus it cannot deal with unbounded event streams. We propose lifespan concept which extends BN to deal with streams. The lifespan is the duration of event occurrence. Each event, which is expressed as a node in BN, should be given a lifespan by user. When a query is evaluated, the evaluation process firstly detects which events are occurred at the moment. The detection is conducted by checking the last occurred time and the lifespan for each event. If the detection time is overlapped with the lifespan of a event, the evaluation process detects the event as occurring. Finally, the probabilities for events detected as occurred are set to 100%, and then update propagation process is invoked in BN. For example, suppose two events A and B occur at time *t1* and *t2* with 2 length lifespans respectively. If a query is evaluated at *t0*, *t2*, *t4*, then the occurrence of, nothing, A and B, B are detected respectively for each time point.

2.3 Operators

BN-Rel inputs event streams and outputs probabilistic event streams. To deal with it, we should define model translation operators for four kinds of data models: stream(S), relation(R), probabilistic stream(PR) and probabilistic stream(PS). Following the Stanford STREAM team who defined model translation operators for relation and stream[4], we define operators for S, R, PR and PS here.

We adopt 7 kinds of model translation operators from 16 combinations. The description of operators are as follows.

1. S-to-R

   S-to-R cuts off a limited number of tuples from unbounded streams. This includes the usual window operator.

2. R-to-PR

   R-to-PR translates relation to probabilistic relation which forms possible worlds[5]. Since probability is the first class citizen (native and mandatory attribute) in probabilistic database, 100 % probability is attached to each tuple.

3. PR-to-R

   PR-to-R translates probabilistic relation to relation, and probability is changed to editable attribute by usual relational operators.

4. R-to-R

   R-to-R includes relational operators such as $\sigma, \pi, \bowtie$ $, \delta, \alpha$.

5. PR-to-PR

   PR-to-PR executes relational operators in possible worlds[5]. The execution is same as (4) except for that the probability of a tuple is computed by the framework of probabilistic data model.

6. PR-to-PS

   PR-to-PS outputs the result of query evaluations to the outside of BN-Rel as probabilistic streams.

```
MASTER  10sec
SELECT  bn
        .getNode(Ev=Fire)
        .makeTuple()
FROM    tableR
WHERE   tableR.Room=7
```

Figure 1. Query Example

7. R-to-S

   R-to-S outputs the result of query evaluations to the outside of BN-Rel as normal streams.

Readers may wonder why other combinations do not exist. It is because this system does not receive PS and does not generate PS internally.

2.4 Query Language

Our model is object relational data model and thus we design query language based on SQL. An example query of our language is shown in Figure 1. The query is processes as follows. Firstly, the query should choose a relation named “tableR” and then a tuple of which “Room = 7” is selected from the relation. Then an attribute “bn” is projected from the tuple. Then, a node of which “Ev = Fire” is chosen by the getNode signature, and finally the data is packed into a tuple by makeTuple. This process is conducted every 10 seconds as MASTER clause specifies.

3 Related Work

3.1 Stream Processing

From the viewpoint of stream processing, the technology is rapidly progressing. Though the first generation and second generation stream processing engines[1, 2] Ignored concrete applications, the third generation SPEs are motivated by specific application such as event stream processing[6], signal processing[7] and audio-visual sensors[12].

However, most of the work still ignores probabilistic reasoning. Even uncertainty which is inherent to the physical world, is not yet full attacked. Therefore we argue that this paper is novel on that it deals with probabilistic reasoning in the viewpoint of stream processing.

3.2 Temporal Bayesian Network

This paper presented the “lifespan” concept to detect multiple event occurrences in a Bayesian network. As an extension of Bayesian network for time domain, temporal Bayesian network of events (TBNE) is already proposed[3]. From [3], a TBNE is defined as follows. "A TBNE is a Bayesian network in which each node represents an event or state change of a variable, and an arc corresponds to a causal-temporal relation. A temporal node represents a possible state change of a variable and the time when it happens. Each value of a temporal node is defined by an ordered pair: the value of the variable to which it changes and the time interval of its occurrence. Time intervals represent relative times between the parent events and the corresponding state change."

The conceptual difference between the time interval in TBNE and the lifespan is the existence of causal-temporal relationship. The causal-temporal relationship expresses the relationship between nodes in a Bayesian network in a causal-temporal aspect. Plus, the purpose of TBNE is to conduct effective reasoning for causal-temporal events in a Bayesian network. While on the other hand, lifespan does not express the relationship between nodes in a Bayesian network. Each lifespan just expresses how long an event occurs for each event. Plus, the purpose of lifespan is to detect multiple events for a query. Thus, TBNE and lifespan are independent.

4 Conclusions and Future Work Issues

4.1 Conclusions

The purpose of this paper was the development of a technology that supports a system which continuously monitors the physical world event streams. To achieve the purpose, we formulated three problems. They were (1) integration of BN and relational data and (2) providing a mechanism to cope with unbounded event streams.

As for (1), we proposed a new object relational data model in which BN is stored and probabilistic values are packed into a relation. As for (2), we proposed a concept of lifespan, and clarified query evaluations to BN for unbounded event streams.

4.2 Future Work Issues

This paper described a model for the integration of relational database and reasoning. This field includes many research challenges. They include the following.

1. Full implementation of a real DBMS which integrates reasoning and relational data.

2. Considering efficient probability propagation algorithms on streaming environment.

3. Query optimization techniques over probabilistic reasoning and relational operators.

4. Establishment of multiple query optimization algorithms.

5. Considering other reasoning algorithms such as HMM and Kalman filter.

6. Establishment of parallel distributed algorithms for in network Query/reasoning processing.

Acknowledgement

This research has been supported in part by the Grant-in-Aid for Scientific Research from JSPS(#18200005, #18700096, #20700078).

References

[1] D. Abadi, Y. Ahmad, M. Balazinska, U. Cetintemel, M. Cherniack, J.-H. Hwang, W. Lindner, A. Maskey, A. Rasin, E. Ryvkina, N. Tatbul, Y. Xing, and S. Zdonik. The design of the borealis stream processing engine. In *Proc. of CIDR*, 2005.

[2] A. Arasu, S. Babu, and J. Widom. Cql: A language for continuous queries over streams and relations. In *Proc. of DBPL*, pages 1–19, 2003.

[3] G. Arroyo-Figueroa. Temporal bayesian network of events for diagnosis and prediction in dynamic domains. *Applied Intelligence*, 23:77–86, 2005.

[4] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom. Models and Issues in Data Stream Systems. In *Proc. of PODS*, 2002.

[5] N. Dalvi and D. Suciu. Management of probabilisitic data foundations and challenges. In *Proc. of PODS*, pages 1–12, 2007.

[6] Y. Diao, N. Immerman, and D. Gyllstrom. Sase+: An agile language for kleene closure over event streams. In *UMass Technical Report 07-03*, 2007.

[7] L. Girod, Y. Mei, R. Newton, S. Rost, A. Thiagarajan, H. Balakrishnan, and S. Madden. The case for a signal-oriented data stream management system. In *Proc. of CIDR*, pages 397–406, 2007.

[8] P. Graham. http://www.paulgraham.com/spam.html.

[9] M. Kadota, H. Aida, J. Nakazawa, and H. Tokuda. D-jenga: A parallel distributed bayesian inference mechanism on wireless sensor nodes. In *Proc. of INSS*, 2006.

[10] R. E. Neapolitan. *Learning Bayesian Networks*. Prentice Hall, 2004.

[11] S. Russel and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 2002.

[12] S. Yamada, Y. Watanabe, H. Kitagawa, and T. Amagasa. Location-based information delivery using stream processing engine streamspinner. In *Proc. of MDM*, page 57, 2006.