INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH & DEVELOPMENT

ISSN 2278 – 0211 (Online)

# Event Prediction with Dynamic Knowledge Base on Health Care Data

**V. Revathi**
Assistant Professor, Department of IT
M. P. Nachimuthu M. Jaganathan Enginering College, India
**K. V. Sreelekha**
Assistant Professor, Department of IT
M. P. Nachimuthu M. Jaganathan Enginering College, India

***Abstract:***
*The association rule mining techniques are used to detect activities from data sets. Event detection refers to an action taken to an activity. The gap between the actual event and event notification should be minimized. Event derivation should also scale for a large number of complex rules. Attacks and its severity are identified from event derivation systems. Transactional databases and external data sources are used. Event detection system identifies the new events in uncertainty environment. Relevance estimation is a more challenging task under uncertain event analysis. Selectability and sampling mechanism are used to improve the derivation accuracy. A Bayesian network representation is used to derive new events given the arrival of an uncertain event and to compute its probability. The event derivation system is enhanced to map dynamic rules on uncertain data environment. Rule probability estimation is carried out using the apriori algorithm. The rule derivation process is optimized for domain specific model.*

***Key words****: Complex Event Processing (CEP), Uncertain Event*

## 1. Introduction

In recent years, there has been an increasing need for event driven (or active) systems (systems that react automatically to events). The existing event-driven systems in the database realm impacted both industry (triggers) and academia (view materialization). New applications in areas such as Business Process Management (BPM) [6]; sensor networks; security applications (example, bio hazards and computer security); engineering applications (such as forecasting networked resources availability); and scientific applications (e.g., utilization of the grid resources) all require sophisticated mechanisms to manage and react to events. Some events can be generated externally and deliver data across distributed systems, while the other events and their related data need to be derived by the system itself. In many of the cases, such derivation is carried out based on a set of rules. Carrying out such event derivation is hampered by the gap between actual occurrences of events, to which the system must respond. This gap results in uncertainty and may be attributed to unreliable event sources (e.g., an inaccurate sensor reading or an unreliable Web service).

In this work, a mechanism to construct the probability space that captures semantics and defines the probabilities of possible worlds using an abstraction based on a Bayesian network has been presented. In order to improve derivation efficiency two mechanisms have been employed: First mechanism Selectability, limits the scope of impact of events to only those rules to which they are relevant, and enables a more efficient calculation of the exact probability space. Second mechanism is one of approximating the probability space by employing a sampling technique over a set of rules.

## 2. Related Work

Complex event processing is supported by systems from various domains. These include ODE, Snoop, and others for active databases and the Situation Manager Rule Language, a general purpose event language. Event management was also introduced in the area of business process management [6] and service-based systems.

The majority of existing models do not support event uncertainty. Therefore, solutions adopted in the active database literature, such as the Rete network algorithm fails to provide an adequate solution to the problem, since they cannot estimate the probabilities. This work was followed by a probabilistic event language for supporting RFID readings [3].

A common mechanism for handling uncertainty reasoning is a Bayesian network, a method for graphically representing a probability space, using the probabilistic independencies to enable a relatively sparse representation of the probability space.

## 3. Event Derivation Models

The challenges associated with this event derivation; note first that such events only suggest a high probability of a flu outbreak, which does not necessarily mean that such an event should be derived. In addition to this, there is uncertainty regarding the data itself, because the provided data are rounded to the nearest ten.

| EID | Date | Daily Sales (Rounded) |
|---|---|---|
| 113001 | Nov 30 | 600 |
| 120101 | Dec 1 | 700 |
| 120201 | Dec 2 | 800 |
| 120301 | Dec 3 | 900 |
| 120401 | Dec 4 | 950 |
| 120501 | Dec 5 | 930 |

*Table 1: Over-the-Counter Sales Relation*

For example, the increase from November 30 (with 600 units sold) to December 4 (with 950 units sold) is of 350 units, yet rounding may also suggest a smaller increase of 341 units. This can be considered as an uncertainty at the source, resulting from inaccurate information provided by the event source. The rule presented here is a simplified version of rules expected to exist in the real world. For example, instead of just specifying a single certainty level of 90 %, one can compute the probability of an outbreak as a function of amount of increase (such as computing the probability to be min(90 + (X - 350)/10,100) where X is the minimum daily sales over a four-day count and should be at least 350).

## 4. Probabilistic Event Model

### *4.1. Event Model*

An event is an actual occurrence or happening that is significant and atomic. Explicit events can be signaled by external event sources (such as OtCCMS events). Derived events are the events for which no direct signal exists, but rather need to be derived based on the other events, (e.g., Flu Outbreak and Anthrax Attack events). Data can be associated with an event occurrence. Some data types are common to all events (e.g., occurrence time of the event), while others are specific (e.g., sales in the OtCCMS event).

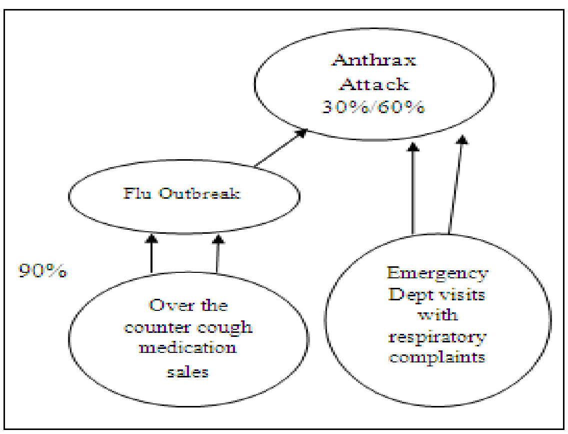


*Figure 1: Anthrax rule*

### *4.2. Derivation Model*

Derived events in the model are inferred using rules. Rather, we represent a rule by a quintuple, r = $\langle s_r, p_r, a_r, m_r, pr_r \rangle$ defining the necessary conditions for the derivation of new events. Such a quintuple can be implemented in a variety of ways, as a set of procedural statements.

**5. Event Detection Algorithms**

Selectability, as defined by function $s_r$ in a rule specification, plays an important role in the event derivation, in both deterministic and the uncertain settings. It is the role of the selection expression to filter out the events that are irrelevant for the derivation according to r.

*5.1. Selectability Algorithm*

As in the uncertain setting derivation is carried out on EIDs (Event Instance Data), algorithms are required to compute which EIDs, from a given system event history H, are selectable. Deciding whether an EID E is selectable by rule r may, by itself, incur significant computational effort. Therefore, if the system event history H contains n EIDs, and the size of the state space of the largest EID is m, then the number of possible event histories (and thus the complexity) is $O(m^n)$.

*5.2. Sampling Algorithm*

Given an existing Bayesian network, it is also efficiently possible to approximate the probability of an event occurrence using a sampling algorithm (several such algorithms are known), as follows: Given a Bayesian network with nodes $E_1, \ldots, E_n$, we calculate an approximation for the probability that $E_i$ = {occurred} by first generating m independent samples using a Bayesian network sampling algorithm.

*5.3. The A-Priori Algorithm*

This algorithm precedes level wise.

- Given support threshold s, in the first pass we find the items that appear in at least fraction s of the baskets.
- Pairs of items in $L_1$ become the candidate pairs $C_2$ for the second pass. We hope that the size of $C_2$ is not so large that there is not room for an integer count per candidate pair. The pairs in $C_2$ whose count reaches s are the frequent pairs, $L_2$.
- The candidate triples, $C_3$ are those sets {A, B, C} such that all of {A, B}, {A, C}, and {B, C} are in $L_2$. On the third pass, count the occurrences of triples in $C_3$ those with a count of at least s are the frequent triples, $L_3$.
- Proceed as far as you like (or the sets become empty). $L_i$ is the frequent sets of size i; $C_{i+1}$ is the set of sets of size i + 1 such that each subset of size i is in $L_i$.

<u>5.3.1. Uses of A-Priori Algorithm</u>

Consider the following SQL on a Baskets (BID; item) relation with $10^8$ tuples involving $10^7$ baskets of 10 items each; assume 100,000 different items.

SELECT b1.item, b2.item, COUNT (*)
FROM Baskets b1, Baskets b2
WHERE b1.BID = b2.BID AND b1.item < b2.item GROUP BY b1.item, b2.item
HAVING COUNT (*) >= s;

Note: s is the support threshold and the second term of the WHERE clause is to prevent pairs of items that are really one item, and to prevent pairs from appearing twice.

<u>5.3.2. Enhancements of A-Priori Algorithm</u>

Two types:

- Cut down the size of the candidate sets $C_i$ for $i \geq 2$. This option is important, even for finding frequent pairs, since the number of candidates must be sufficiently small that a count for each can fit in main memory.
- Merge the attempts to find $L_1$, $L_2$, $L_3$,…… into one or two passes, rather than a pass per level.

**6. Event Detection with Rule base Management**

The event derivation system is enhanced to map dynamic rules on uncertain data environment. The rule base construction and maintenance operations are handled by the system. The system integrates the rule base update process.

*6.1. Patient Diagnosis*

The system uses TB patient diagnosis information. Patient diagnosis data can be imported from external databases. The user can also update new patient diagnosis details. Diagnosis list shows the patient symptom levels.

*6.2. Rule Base Management*

Rule base is used to manage the rules and event names. Rule base details are collected from domain experts. The rules are composed with attribute combination and values. Three types of rules are used in the system. Static, dynamic and hybrid rules are maintained under the rule base.

*6.3. Sampling Process*

The sampling process is performed to select event derivation models. The attribute values are sampled from uncertain data collections. The sampling algorithm is used to select sample data derivations. Sampled data values are passed to rule analysis process.

*6.4. Event Detection*

The rule base analysis is performed with user data collections. The Bayesian network is used in the event detection process. The attribute combinations are compared with rule base information. Event detection is carried out with rank and priority information.

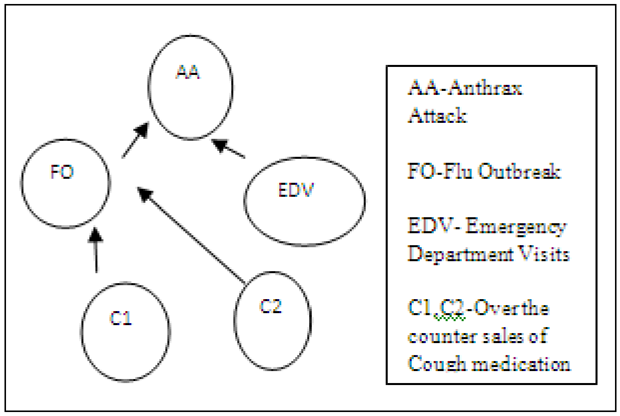


*Figure 2: Bayesian Network Graph Example*

*6.5. Selection Process*

The selection process is applied to filter irrelevant events. Item combinations are selected under the selection process. The Selectability algorithm is used in the selection process. Event derivations are used in the rule analysis process.

*6.6. Rule Base Update*

The dynamic rules are derived from user data and static rule base information. The dynamic rules are updated with event name and priority values. Rule ranking is performed with frequency information. Rules are updated with priority details.

*6.7. Dynamic Event Detection with Rule Base*

6.7.1. Update Algorithm

The rule base will be updated using the Following algorithm steps:

- Fetch data from remote sources
- Find item combinations
- Apply item selection process
- Verify rule base data values
- Find event name
- Check possible rules
- If new rules then update rule base
- Else check next item
- Update event name

**7. Data Set and Experimental Setup**

In our experiments, we varied control parameters that impact performance. These parameters include the number of events, out of all processed events, that are relevant to an event derivation, the number of possible worlds, the approximation precision, and rule probabilities.

The performance of the proposed system has been evaluated using the following metrics:

- Event rate - event detection rates for transaction
- Event relevancy - relevant event identification ratio
- Possible worlds - scalability analysis
- Detection latency - event detection interval analysis
- Rule strength - strength analysis for static and dynamic rules.

*7.1. Performance as a result of number of relevant events*

The tested cases were the Standby/Noisy case, in which at most 1 percent of events were relevant to derivation, the Filtered case, in which 20 percent of the events were relevant to derivation, and the Complex case, in which 80 percent of the events were relevant to derivation. The graph shows the performance difference both in the deterministic case and in the two probabilistic uncertain settings.

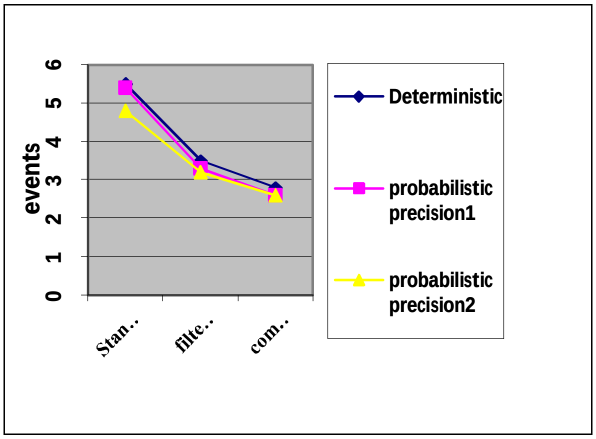


*Figure 3*

## 8. Conclusion

Rule bases are built to manage activities and their class information. Event derivation is carried out with rule bases. Dynamic rule base update model has been introduced to improve the event detection process. Selective and sampling algorithms are enhanced to derive events under dynamic and uncertain domain environments. The system integrates the expert's knowledge and local domain information for rule base construction. Rule base generation and maintenance operations are done using machine learning models. Event classes and its structures are generalized. Association rule mining methods are used to extract rule patterns.

## 9. References


1. Nodira Khoussainova, Magdalena Balazinska, Dan Suciu (2008) "Probabilistic Event Extraction from RFID Data" Computer Science & Engineering Department, University of Washington.
2. Segev Wasserkrug, Avigdor Gal, Opher Etzion and Yulia Turchin( 2012) "Efficient Processing of Uncertain Events in Rule-Based Systems" IEEE Transactions On Knowledge and Data Engineering, Vol. 24, No. 1.
3. Balazinska.M, Khoussainova.N, and Suciu.D (2008) "PEEX: Extracting Probabilistic Events from RFID Data," Proc. Int'l Conf. Data Eng.
4. Re, Dalvi.N, and Suciu.D ( 2006) "Query Evaluation on Probabilistic Databases," IEEE Data Eng. Bull., vol. 29, no. 1, pp. 25-31.
5. Wasserkrug.W,Gal, Etzion.O, and Turchin (2008) "Complex Event Processing over Uncertain Data," Proc. Second Int'l Conf. Distributed Event-Based Systems (DEBS '08), pp. 253-264.
6. Ammon.R, Emmersberger.E, Greiner.T, Paschke.A, Springer, and Wolff (2008) "Event-Driven Business Process Management," Proc. Second Int'l Conf. Distributed Event-Based Systems (DEBS '08).
7. Demers, Gehrke.J, Panda.B, Riedewald, Sharma.V, and White.W (2007) "Cayuga: a General Purpose Event Monitoring System," Proc. Third Biennial Conf. Innovative Data Systems Research, pp. 412-422.
8. Cowie.J, Ogielski, Premore, and Y. Yuanb, "Internet Worms and Global Routing Instabilities," Proc. SPIE, vol. 125, p. 4868.