http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet

Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 9 (8): 1706-1712, 2010
ISSN 1812-5638

Converting Probabilistic Relational Data to Probabilistic XML Data Tree

[1,3]Jianwei Wang and [1,2]Zhongxiao Hao
[1]College of Computer Science and Technology,
Harbin University of Science and Technology, Harbin 150080, China
[2]College of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China
[3]Information and Computer Engineering College, Northeast Forestry University, Harbin 150040, China

Abstract: As a kind of special XML data, probabilistic XML has been also presented as the de facto standard for probability data exchange on the web. And there are a mass of probabilistic relational data. In order to realize the conversion of probabilistic relational data to probabilistic XML data, the representation methods of probability data in the probabilistic relational data and in the probabilistic XML data are firstly analyzed separately. And the probabilistic XML data tree is a classical probabilistic data model with some distribution nodes. The conversion process of probabilistic relational data to probabilistic XML data is composed of two stages what are schema conversion and data conversion. Algorithm PRS2PDTD converts the probabilistic relational schema to the probabilistic DTD, while preserving semantic constraints of the original probabilistic relational schema. And algorithm PRD2PT converts probabilistic relational data to the probabilistic XML data tree. The instance analysis results show that the algorithms are effective methods for conversion of the existent probabilistic relational data to the probabilistic XML data.

Key words: Probabilistic relational data, probabilistic DTD, probabilistic XML data tree, schema conversion, data conversion

INTRODUCTION

Because XML emerged as the de facto standard for data formats on the web (XML-TR, http://www.w3.org/TR/, 2007; Nath and Batanov, 2005; Haw and Lee, 2007), the probabilistic XML data has been also presented as a special style of XML data from 2001. It can represent the distribute distribution data in XML element in the form of probabilistic data. The researchers mainly focuses the fields what are the probabilistic XML data models (Nierman and Jagadish, 2002; Van Maurice *et al*., 2005; De Keijzer and van Keulen, 2007) the probabilistic XML algebra (Zhao *et al*., 2005; Magnani and Montesi, 2008; Dekhtyar *et al*., 2006) query methods (Benny and Sagiv, 2007; Kimelfeld *et al*., 2008) and prototype system design (Nierman and Jagadish, 2002; Hung and Subrahmanian, 2005; Li *et al*., 2006).

The probabilistic relational data is a kind of relational data model that represents the triple uncertain relation data in the form of the probability data (Zhongxiao, 1996; Yuan, 2005). Cavallo and Pittarelli (1987) and Barbar *et al*. (1992) proposed a kind of extended relation data model to represent probability data by adding the attribute of probability data and defined a function that can convert the relational data to the probabilistic relational data. Lee (1992) and Yuan (2005) applied the Dempster-Shafer theory and probability theory as the basis of theory to research the probabilistic relation data and defined the correlative algebra operation (Dey and Sarkar, 1996).

Since, the probabilistic XML data is the standard of probabilistic data format, the existent probabilistic relation data need be converted to probabilistic XML data. From 1998, the researchers have represented a series of methods in order to convert the relation data to the XML data. Jacinto and Marta (2002) studied the algorithm for bidirectional conversion between XML documents and relational databases. Fong *et al*. (2001) studied the algorithm for converting relational database into XML document. Teng *et al*. (2005) analyzed the conversion relational schema to XML Schema with constraints. Anthony *et al*. (2006) and Sengupta (2010) designed the tool of visual relational to XML conversion. The methods provided some ways for converting the probabilistic relation data to probabilistic XML data.

So, the algorithms for converting the probabilistic relation data to probabilistic XML data are proposed. At first the paper analyzes the representation of probability data in relational data schema and XML data schema. The

Corresponding Author: Jianwei Wang, College of Information and Computer Engineering, Northeast Forestry University, Harbin 150040, China

two triple form <attribute,probability> represents the attribute value and its probability in probabilistic relational data. Because the probabilistic XML data tree is a classical probabilistic XML data model, so the paper applies the tree model (Nierman and Jagadish, 2002; Li *et al*., 2006).

The contributions of this study include:

- We argue that the distribution node form represents the element nodes and their probability data in the probabilistic XML data tree
- We analyze the path mapping rules from the probabilistic relational data to probabilistic XML data tree based on analyzing the path types in the probabilistic XML data tree
- Algorithm PRS2PDTD is proposed to convert probabilistic relational schema to probabilistic DTD
- Algorithm PRS2PT is proposed to convert probabilistic relational data to probabilistic XML data tree
- The instance analysis verifies the efficiency of the two algorithms

METHODS OF PROBABILITY DATA REPRESENTATION

Probabilistic relation schema PRS is a kind of non-classical relation model that represent the uncertainty of probabilistic relation triple in the form of the two triple <attribute,probability> (Yuan, 2005). Probabilistic relation PR is a finite set of triple in PRS and one triple in PR isnot equal with the other triple. The definition of probabilistic relation is proposed which is shown as follows:

Definition 1: Probabilistic relation PR is one of 1-NF probabilistic relation in the probabilistic relational schema PRS, the attribute set is $C = \{c_i,...,c_m\}$, there is one and only one attribute $c_i \leq a$, $ps > i = 2,...,m$, where, a is the attribute name of probabilistic relation, ps is the probability value with the restriction condition:

$$\sum_{u=1}^{|\{<a,ps>\}|} ps \leq 1$$

Probabilistic XML Data Tree (Nierman and Jagadish, 2002; Li *et al*., 2006) is a kind of classical data model of probabilistic XML data that the schema document is probabilistic DTD (Nierman and Jagadish, 2002). The probabilistic XML data tree and probabilistic DTD document are defined as follows.

Definition 2: Probabilistic XML data tree is six tuple PT = (N, E, R, F, V, P), where:

- N is the finite set of node $N = N_{ord},\cup N_{dist}\cup N_{max}\cup N_{poss}$

where, N_{ord} is the set of general node, N_{dist} is the set of distribution node, N_{max} is the set of the node that has the child node that is exclusive form one another, N_{poss} is the set of possibility node.

- $E\subseteq N\times N$ is the set of edges
- R is the root of PT
- F: $N_{ord}\rightarrow L$ is the label function of the general node, where, L is the name of element; $\forall n\in N_{poss}$, F(n) = poss
- v: $N_i\rightarrow D$ is the data of leaf node, where, N_i are the leaf nodes $N_l\subset N_{ord}\cup N_{poss}$, D is the data set
- P: $N_{ord}\cup N_{poss}\rightarrow p$ is the probability value of the given node, $0\leq p\leq 1$, the default value is 1 without assigning the probability value

Definition 3: Probabilistic DTD is a pair (f, N) that maps f to regular expressions over N, and $R\in N$ is the start symbol. PT = (N, E, R, F, V, P) satisfies probabilistic DTD, if its root is labeled by r, for each function F: $N_{ord}\rightarrow L$, and for every node N_{ord} with label L, the sequence L of labels of its children matches the regular expression:

$$f(N_{ord}), \forall n \in N_{poss}, F(n) = \text{'poss'}$$

There are different type paths in a probabilistic XML data tree. The path definitions are proposed which are as follows.

Definition 4: Absolute path in a probabilistic XML data tree PT is defined a series of nodes from the root R to the leaf node $V(N_i)$ $(V(N_l)\in D)$. $APATH_{PT}$ is the set of absolute path in a probabilistic XML data tree PT, $|APATH_{PT}|$ is the number of the element of absolute path set.

Definition 5: Extended path in a probabilistic XML data tree PT is absolute path set without leaf nodes. $EPATH_{PT}$ is the set of extended path in the probabilistic XML data tree PT, $|EPATH_{PT}|$ is the number of the element of extended path set.

Definition 6: Basic absolute path in a probabilistic XML data tree PT is absolute path set without leaf nodes and distribution nodes. $PATH_{PT}$ is the set of basic absolute path in a probabilistic XML data tree PT, $|PATH_{PT}|$ is the number of the element of basic absolute path set.

Definition 7: Candidate absolute path in a probabilistic XML data tree PT is extended path without any distributi- on node. $\mathrm{CPATH_{PT}}$ is the set of candidate absolute path in a probabilistic XML data tree PT, $|\mathrm{CPATH_{PT}}|$ is the number of the element of candidate absolute path set.

Definition 8: The extended absolute path epath is equal to the basic absolute $\mathrm{path_{path}}$ if the distribution nodes in the extended absolute path are deleted. The relation between epath and path is called path equivalence. $\forall \mathrm{epath} \in \mathrm{EPATH_{PT}}$, path = delete(epath.(dist,mux,poss) and path $\in \mathrm{PATH_{PT}}$, epath is equal with path which is denoted by the equation epath $\doteq$ path.

Let, $\mathrm{EPATH_{PT}}$ be extended path set and $\mathrm{PATH_{PT}}$ be basic absolute path set in the probabilistic XML tree PT. Then,

$$1 \leq | \mathrm{EPATH_{PT}} \cap \mathrm{PATH_{PT}} | \leq | \mathrm{PATH_{PT}} | - 1 \tag{1}$$

$$| \mathrm{PATH_{PT}} | \lessdot | \mathrm{EPATH_{PT}} | \tag{2}$$

Theorem 1: Let $\mathrm{DEPATH_{PT}} = \mathrm{EPATH_{PT}} - (\mathrm{EPATH_{PT}} \cap \mathrm{PATH_{PT}})$, $\mathrm{DEPATH_{PT}}\,|_{\max} = n - 1$, $|\,\mathrm{DEPATH_{PT}}\,|_{\min} = 1$. Then, $|\,\mathrm{EPATH_{PT}}\,| = n$. So, $\mathrm{DEPATH_{PT}}$ can be partitioned to some sets of the equivalent path.

Proof: From definition 5 and 6, the expression:

$$1 \leq | \mathrm{DEPATH_{PT}} | \leq n - 1 \tag{3}$$

comes into existence.

According to definition 8,

$$\forall \mathrm{path} \in \mathrm{PATH_{PT}}, \exists \mathrm{epath} \in \mathrm{EPATH_{PT}} \text{ and } \mathrm{epath} \square \mathrm{path}$$

then:

$$\mathrm{DEpath} = \{\, \mathrm{epath} \mid \mathrm{path} \in \mathrm{PATH_{PT}} \wedge \mathrm{epath} \doteq \mathrm{path} \wedge \mathrm{epath} \in \mathrm{EPATH_{PT}} \} \tag{4}$$

$$\mathrm{DEPATH_{PT}} = \{\, \mathrm{DEpath_{PT}} \mid \mathrm{DEpath_{PT}} \subseteq \mathrm{DPATH_{PT}} \wedge \cup \mathrm{DEpath_{PT}} = \mathrm{EPATH_{PT}} - (\mathrm{EPATH_{PT}} \cap \mathrm{PATH_{PT}}) \} \tag{5}$$

Definition 9: The absolute $\mathrm{path_{apath}}$ is equal to the extended absolute $\mathrm{path_{epath}}$ if the leaf nodes in the extended absolute path are deleted. The relation between apath and epath is called path consistent. $\forall \mathrm{epath} \in \mathrm{EPATH_{PT}}$ there is one and only one absolute path $\mathrm{apath} \in \mathrm{APATH_{PT}}$, when $\mathrm{epath} = \mathrm{delete(last(apath))}$, so epath is consist with apath which is denoted by the equation:

$$\mathrm{epath} \triangleq \mathrm{apath}$$

Theorem 2: Let, $\mathrm{DPATH_{PT}} = \mathrm{PATH_{PT}} - (\mathrm{EPATH_{PT}} \cap \mathrm{PATH_{PT}})$ $|\,\mathrm{DPATH_{PT}}\,|_{\max} = m - 1$, $|\,\mathrm{DPATH_{PT}}\,|_{\min} = 1$, then $|\mathrm{PATH_{PT}}|$, so, $\mathrm{DPATH_{PT}}$ can be partitioned to some sets of the consistent paths.

Proof: From definition 5 and 6,the inequation $1 \leq | \mathrm{DPATH_{PT}} | \leq m - 1$ comes into existence.

From definition 9 $\forall \mathrm{dpath} \in \mathrm{DPATH_{PT}}$,

$$\exists \mathrm{epath} \in \mathrm{EPATH_{PT}} \in \mathrm{DEpath_{PT}}, \mathrm{apath} \in \mathrm{APATH_{PT}} \tag{6}$$

if epath $\triangleq$ apath, then

$$\mathrm{Dpath_{PT}} = \{\, \mathrm{apath} \mid \mathrm{apath} \in \mathrm{APATH_{PT}} \wedge \mathrm{epath} \in \mathrm{DEpath_{PT}} \wedge \mathrm{epath} \triangleq \mathrm{apath} \} \tag{7}$$

$$\mathrm{DPATH_{PT}} = \{\, \mathrm{Dpath_{PT}} \mid \mathrm{Dpath_{PT}} \subseteq \mathrm{DPATH_{PT}} \wedge \cup \mathrm{Dpath_{PT}} = \mathrm{PATH_{PT}} - (\mathrm{EPATH_{PT}} \cap \mathrm{PATH_{PT}}) \} \tag{8}$$

ALGORITHM FOR CONVERSION PROBABILISTIC RELATIONAL SCHEMA TO PROBABILISTIC XML SCHEMA

The common method of converting probabilistic relation schema to probabilistic DTD is directly mapping. So the mapping rules are proposed which are listed as follows.

Rule 1: The basic absolute path set $\mathrm{path_{PR}}$ can be com- puted by the attribute set of probabilistic relation PR.

Let, C = $\{c_i,\ldots,c_m\}$ be the attribute set of probabilistic relation PR, then the absolute path set of probabilistic XML data tree PT is $\mathrm{path_{PR}} = \{\mathrm{PR}/c_1,\ldots,\mathrm{PR}/c_m\}$. And the result of union operation of absolute path set from PRS is:

$$\mathrm{PATH_{PRS}} = \bigcup_{i=1}^{n} \mathrm{path_{PR}} \tag{9}$$

Rule 2: The extended absolute path set $\mathrm{epath_{PR}}$ can be computed by the attribute set of probabilistic relation PR.

Let, the attribute set C = $\{c_i,\ldots,c_m\}$ of probabilistic relation PR, then the extended absolute path set of probabilistic XML data tree PT is shown as follows:

$$\mathrm{epath_{PR}} = \{\mathrm{PR}/c_1,\ldots,\mathrm{PR}/c_1/\mathrm{dist}/\mathrm{mux}/\mathrm{poss},\ldots,\mathrm{PR}/c_m\} \tag{10}$$

And the result of union operation of extended path set from PRSis shown as follows:

$$EPATH_{PRS} = \bigcup_{i=1}^{n} epath_{PR}$$

Rule 3: The candidate absolute path set $cpath_{PR}$ can be computed by the attribute set of probabilistic relation PR.

Let, basic absolute path set be $PATH_{PRS}$ and the can absolute path set be $EPATH_{PRS}$, then the candidate absolute path set $CPATH_{PT}$ is shown as follows:

$$CPATH_{PRS} = PATH_{PRS} \cap EPATH_{PRS} \tag{11}$$

According to the mapping rules, the algorithm process of conversion probabilistic relation schema to probabilis-tic XML schema is divided three steps. The root node r is firstly obtained from probabilistic relation schema PRS as the start tag of PDTD. The child node of r is secondly obtainned from probabilistic relation PR. At last the absolute path set is obtained from the attribute set of probabilistic relation PR. So, the algorithm is shown as follows.

Algorithm 1: PRS2PDTD (PRS)

```
input : probabilistic relation schema PRS
output:probabilistic XML schema PDTD
   begin
      PDTD=ø;
      r=name(PRS);PDTD= PDTD∪{r};
      element=ø;
      for i=1 to | PRS |{
r_PRS_i=name(PRS[i]);
element =element∪{r_PRS_i};
element(i)=ø;
for j=1 to |column(PRS[i])|{
sub_element(i)=element(i)∪{a(j)};
if ps∈a(j) then
         {{a(j) (dist)}∈PDTD; {dist(mux)}∈PDTD;
            {mux(poss)}∈PDTD;}}
      {element(element(i))}∈PDTD; }
   {r(element)}∈PDTD;
  return(PDTD)
end
```

Theorem 3: Algorithm PRS2PDTD () for converting the probabilistic relation schema to probabilistic XML data is correct and complete.

Proof: Definition 1, 3, 5, 6, 7, 8 and the mapping rules offer methods to compute PRS2PDTD (PRS) and prove themselves when PRS is a set of probabilistic relational schema in the form of 1-NF. The algorithm fulfills the methods offered by these definition and these theorems. Therefore, the algorithm can correctly compute PRS2PDTD (PRS) for any PR in PRS. Thus the algorithm is correct and complete.

ALGORITHM OF CONVERSION PROBABILISTIC RELATION DATA TO PROBABILISTIC XML DATA TREE

The method of conversion probabilistic relation data to probabilistic XML data is to convert the triples of probabilistic relation data to the absolute path. So the mapping rules are proposed which are listed as follows.

Rule: The absolute path set $apath_{PR}$ can be computed by the attribute set of probabilistic relation PR.

Let, the sattribute set $C = \{c_1,...,c_m\}$ of probabilistic relation PR, then the absolute path set of probabilistic XML data tree is shown as follows:

$$\begin{aligned} apath_{PR} = \{&PR/c_1/v_1(c_1),...,\\ &PR/c_1/dist/muv/poss/\{<v,p>\},\\ &...,PR/c_m/v_m(c_m)\} \end{aligned} \tag{12}$$

And the result of union operation of absolute path set of probabilistic XML data tree $apath_{PRS}$ is:

$$APATH_{PRS} = \bigcup_{i=1}^{n} apath_{PR} \tag{13}$$

According to the mapping rule, the algorithm process of conversion probabilistic relation data to probabilistic XML data is proposed which is shown as follows.

The root node r is firstly obtained from probabilistic relation data PRS as the root of PT. The absolute path set$_{apath_{PR}}$ is secondly obtained from probabilistic relation PR. At last the absolute path set is created from the attribute set of probabilistic relation PR.

So, the algorithm is shown as follows.

Algorithm 2: PRD2PT(PRS)

```
input : probabilistic relation PRS
ouput : absolute path set APATH of probabilistic XML tree PT
begin
      r=name(PRS);APATH=ø;
      for i=1 to | PRS |{
      element=name(PRS(i));
         A=ø;
      for j=1 to| row(PRS[i])|{
      for k=1 to |column(PRS[i])|{
         if ps∈column(PRS[i]) then{
         A= A∪{r/element / c(j)/v(c(j))};
         else
      A= A
           ∪{r/element /c(j)/dist/muv/poss[ps]/ v(c(j))}}}};
      APATH= APATH∪A;}
      return(APATH);
end
```

Theorem 4: Algorithm PRD2PT () for converting the probabilistic relational data to probabilistic XML data tree is correct and complete.

Proof: Definition 1, 2, 3 and the mapping rule offer methods to compute PRD2PT (PRS) and prove themselves when PRS is a set of probabilistic relational data in the form of 1-NF. The algorithm fulfills the methods offered by these definitions and the rule. Therefore, the algorithm can correctly compute PRD2PT (PRS) for any PR in PRS. Thus, the algorithm is correct and complete.

INSTANCE ANALYSIS

The purpose of this section is to confirm that algorithm PRS2PDTD and PRD2PT are correct by comparing the results of the algorithms with the teacher probabilistic relation.

Example 1: The teacher probabilistic data about rank, department and course are represented in the probabilistic relational data table. The teacher probabilistic relational schema is composed of Table 1-4 where no is the key attribute in Table 1. The <rank,ps> attribute in Table 2 represents the rank and its distribution data, the <dept,ps> attribute in Table 3 represents the department and its distribution data, and the <course,ps> attribute in Table 4 represents the teaching course and its distribution data.

From the tables, the probabilistic relational schema is shown as follows.

teacher = { teacher-name, teacher-rank,
teacher-department, teacher-course}

The attribute set of teacher is shown as follows.

C={no,name,<rank,ps>,<dept,ps>,<course,ps>}

So, the basic absolute path set is computed which is shown as follows.

APATH ={teacher/no,teacher/name,teacher/rank,teacher/
dept,teacher/course}

The elements of APATH is also shown in Fig. 1.

The extended absolute path set is computed which is shown as follows.

EPATH = {teacher/no,teacher/name,teacher/rank/
rank/dist/mux/poss, teacher/dept/dist/mux/poss,
teacher/course/dist/mux/poss}

The elements of EPATH is also shown in Fig. 2.

The candidate absolute path set is computed which is shown as follows.

CPATH={ teacher/no,teacher/name}

The elements of CPATH is also shown in Fig. 3.

Table 1: Teacher-name

No.	Name
0001	Wang Ming
0002	Zhang Min
0003	Zhao Li
0004	Feng Yan
0005	He Lu

Table 2: Teacher-rank

No.	<rank,ps>
0001	<ass,0.3>
0001	<ins,0.7>
0002	<a.prof,0.7>
0002	<prof,0.3>
0003	<ass,0.5>
0003	<ins,0.5>
0004	<a.prof,04>
0004	<prof,0.6>
0005	<ins,0.6>
0005	<a.prof,04>

Table 3: Teacher-department

No.	<dept,ps>
001	<d1,0.2>
0001	<d2,0.8>
0002	<d1,0.3>
0002	<d4,0.7>
0003	<d4,0.2>
0003	<d5,0.8>
0004	<d2,0.6>
0004	<d3,0.4>
0005	<d2,0.6>
0005	<d4,0.4>

Table 4: Teacher-course

No.	<course,ps>
0001	<c1,0.3>
0001	<c2,0.5>
0001	<c3,0.2>
0002	<c2,0.3>
0002	<c3,0.4>
0002	<c4,0.3>
0003	<c3,0.5>
0003	<c4,0.2>
0003	<c5,0.3>
0004	<c1,0.3>
0004	<c2,0.7>
0005	<c3,0.2>
0005	<c4,0.3>
0005	<c5,0.5>

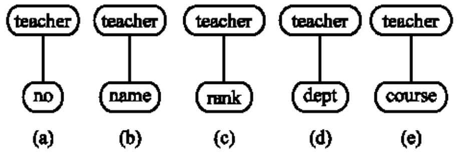

Fig. 1: Basic absolute path set, (a) teacher/no, (b) teacher/name, (c) teacher/rank, (d) teacher/dept and (e) teacher/course

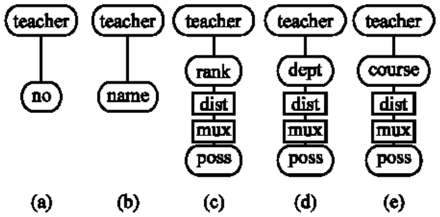

Fig. 2: Extended absolute path set, (a) teacher/no, (b) teacher/name, (c) teacher/rank/dist/mux/poss, (d) teacher/dept/dist/mux/poss and (e) teacher/course/dist/mux/poss

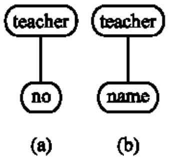

Fig. 3: Candidate absolute path set, (a) teacher/no and (b) teacher/name

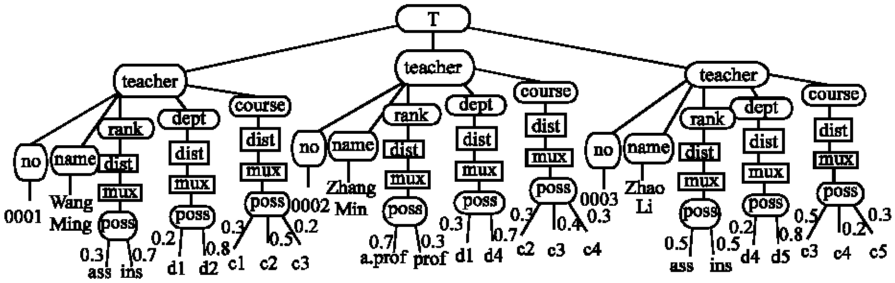

Fig. 4: Teacher probabilistic XML data tree PT

Based on the analysis of path type, the PDTD document is computed by algorithm PRS2PDTD. PDTD document of teacher is shown as follows:

```
PDTD document of teacher:
<!ELEMENT T (teacher+)>
<!ELEMENT teacher (no, name, rank, dept, course)>
<!ELEMENT no (#PCDATA)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT rank (dist)>
<!ELEMENT dept (dist)>
<!ELEMENT course (dist)>
<!ELEMENT dist (mux)>
<!ELEMENT mux (poss+)>
<!ELEMENT poss (#PCDATA)>
<!ATTLIST possCDATA #REQUIRED>
```

The probabilistic XML data tree is computed by algo- rithm PRD2PT which is shown in Fig. 4.

CONCLUSIONS

In this study we present algorithms to solve the problem of conversion probabilistic relational schema to probabilistic DTD and conversion probabilistic relational data to probabilistic XML data. The contribution is original and useful in itself, additionally, it will be helpful to convert the existing probabilistic relational data to probabilistic XML data and probabilistic XML data become the probabilistic data exchange standard.

ACKNOWLEDGMENTS

Authors would like to thank National Natural Science Foundation of China and Hei longjiang National Science Grant No. 60673136 and No. F200702, for funding this research. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

Anthony, L., R. Alhajj and K. Barke, 2006. VIREX: Visual relational to XML conversion tool. J. Visual Languages Comput., 17: 25-45.

Barbar, D., H.G. Molina and D. Porter, 1992. The management of probabilistic data. IEEE Trans. Knowl. Data Eng., 4: 487-502.

Benny, K. and Y. Sagiv, 2007. Matching twigs in probabilistic XML. Proceedings of the 33rd International Conference on Very Large Data Bases, Sept. 23-27, Vienna, Austria, 27-38.

Cavallo, R. and M. Pittarelli, 1987. The theory of probabilistic databases. Proceedings of the 13th International Conference on Very Large Data Bases, Sept. 1-4, Brighton, England, pp: 71-81.

De Keijzer, A. and M. van Keulen, 2007. User feedback in probabilistic integration. Proceedings of the 18th International Workshop on Database and Expert Systems Applications,Proceedings of the 18th International Workshop on Database and Expert Systems Applications, Sept. 3-7, Regensburg, Germany, pp: 377-381.

Dekhtyar, A., K.K. Mathias and P. Gutti, 2006. Structured queries for semistructured probabilistic data. Proceedings of 2nd Twente Data Management Workshop on Uncertainty in Databases, June 6, Netherlands, pp: 11-18.

Dey, D. and S. Sarkar, 1996. A probabilistic relational model and algebra. ACM Trans. Database Syst., 21: 339-369.

Fong, J., F. Pang and C. Bloor, 2001. Converting relational database into XML document. Proceedings of the 12th International Workshop on Database and Expert Systems Applications, IEEE Computer Society. Sept. 3-7, Washington, DC, USA., pp: 61-65.

Haw, S.C. and C.S. Lee, 2007. Structural query optimization in native XML databases: A hybrid approach. J. Applied Sci., 7: 2934-2946.

Hung, E. and V.S. Subrahmanian, 2005. Managinguncertaintyand ontologies in databases. Ph.D. Thesis, University of Maryland

Jacinto and H. Marta, 2002. Bidirectional conversion between XML documents and relational data bases. Proceedings of the International Conference on CSCW in Design, Sept. 25-27, Brazil, pp: 4-9.

Kimelfeld, B., Y. Kosharovsky and Y. Sagiv, 2008. Query efficiency in probabilistic XML models. Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, June 10-12, Vancouver, Canada, pp: 701-714.

Lee, S.K., 1992. An extended relational database model for uncertain and imprecise information. Proceedings of the 18th International Conference on Very Large Data Bases, Aug. 23-27, Vancouver, Canada, pp: 211-220.

Li, T., Q. Shao and Y. Chen, 2006. PEPX: A query-friendly probabilistic XML database. Proceedings of the 15th ACM International Conference on Information and Knowledge Management, Nov. 6-11, ACM Press, New York, pp: 848-849.

Magnani, M. and D. Montesi, 2008. Management of interval probabilistic data. Acta Inform., 45: 93-130.

Nath, U.K.D. and D.N. Batanov, 2005. Comparative analysis of three promising XML query languages and some recommendations. Inform. Technol. J., 4: 439-444.

Nierman and H.V. Jagadish, 2002. ProTDB: Probabilistic data in XML. Proceedings of the 28th VLDB Conference, Aug. 20-23, Hong Kong, China, pp: 446-457.

Sengupta, G.J., 2010. Regression testing method based on XML schema for GUI components. J. Software Eng., 4: 137-146.

Teng, L., P. Yan and Q. Huang, 2005. Relational to XML Schema Conversion with Constraints. Vol. 3818, Springer, Berlin, Heidelberg, ISBN: 978-3-540-30767-9, pp: 278-279.

Van Maurice, K., A. de Keijzer and W. Alink, 2005. A probabilistic XML approach to data integration. Proceedings of the 21st International Conference on Data Engineering, April 5-8 Tokyo, Japan, pp: 459-470.

Yuan, M., 2005. Non_Classical Relational Database Theory. Tsinghua University Press, Beijing, China, pp: 203-226.

Zhao, Z., A. Dekhtyar and J. Goldsmith, 2005. A framework for management of semistructured probabilistic data. J. Intell. Inform. Syst., 25: 293-332.

Zhongxiao, H., 1996. Database Guide Under the Null Values Circumstances[M]. Machine Press, Beijing, China, pp: 3-131.