MAPPING DATA BETWEEN PROBABILITY SPACES IN PROBABILISTIC DATABASES

Abdallah Alashqur

Faculty of Information Technology
Applied Science University
Amman, Jordan

ABSTRACT

Several advanced applications require that uncertain data be stored in the database. Applications with sensor data, data mining, and integrated data are just few examples in which probabilistic data is considered a first class citizen. In response to this demand for storing and managing probabilistic data, researchers have started in recent years addressing issues pertaining to the management uncertain data. Representation and modeling of probabilistic data is one of the areas that needs attention. In this paper, we summarize our previous work on how probabilistic data can be represented along three different probability spaces, namely, attribute probability space, record probability space, and database state probability space. Then we introduce techniques for mapping the data from the attribute probability space to record probability space and from record probability space to database state probability space. The ability to perform correct mappings of data between these probability spaces is important in order to preserve the integrity of the data and avoid any data loss during the mapping process.

KEYWORDS

Probability theory; Probabilistic databases; uncertainty; advanced database applications.

1. INTRODUCTION

In recent years there has been increasing interest in coming up with powerful ways to capture and represent probabilistic data in relational database systems. This has been motivated by many emerging applications that demand such capability. Examples of applications that need to support probabilistic data (i.e., data that is imprecise or uncertain) include sensor data, data extracted from text files, data integrated from a variety of sources, medical diagnosis data, and data mining. In [1] we give a brief description of applications that benefit from the capability of handling probabilistic data.

In response to this demand, the research database community started addressing issues related to probabilistic databases. Many experimental database systems that support probabilistic data have recently been built in order to explore and experiment with various aspects of probabilistic data. In addition, several research articles have been published to address different features that need to be supported in probabilistic database systems [2-7]. Some of the Prototype systems that have been reported in the literature include Trio [8], *MayBMS* [9], *MystiQ* [10], *Prob-View* [11], *Orion* system [12], MCDB [13], and BayesStore [14]. Trio, *MayBMS*, and *MystiQ* are systems that support discrete forms of uncertainty, in which a finite set of possible instances is represented.

DOI : 10.5121/ijdms.2015.7301

MayBMS expresses constraints between records in the form of “conditions.” The technique used by MayDMS resembles a technique called *lineage* that is used in Trio. MystiQ represents uncertainty of data in the form of existence probabilities over independent records. ProbView maintains what is called *probability intervals* for each record. The *Orion* system focuses on probabilistic data for applications that use sensors. This requires modeling *continuous* forms of probability, involving infinite sets of possible instances. MCDB and BayesStore are systems that incorporate AI inference mechanisms as well as statistical models for managing uncertainty. Other related work on databases with uncertainty can be found in [15-20].

In [1] we introduced a new approach for viewing and presenting probability in databases with uncertainty. In our approach, probability can be considered at three levels of granularity. We also introduced the definitions of three probability spaces where each probability space correspond to one of these three levels of granularity.

In this paper, we extend our work in [1] by introducing techniques to map data from one probability space to another. Sound mapping techniques are important in order to preserve data integrity and avoid any loss of information during the mapping. This research is part of an on-going research project called probabilistic data management and mining (PDMM).

The remainder of this paper is organized as follows. In section 2 we summarize our previous work in the area or probabilistic databases, since it serves as a basis for the research introduced in this paper. In section 3 mapping from attribute probability space to record probability space is described. In Section 4 we present mapping from record probability space to database state probability space. Conclusions are given in Section 5.

2. BACKGROUND INFORMATION

Imprecise or uncertain data in a relational database can be considered at the level of an attribute, record, or database. Uncertainty at the attribute level means that a value in a record can actually be a set of probable values instead of just a single deterministic value. Uncertainty at the level of an entire record indicates that the existence of the record is not certain (it may or may not exist in the database). The third type of uncertainty is at the level of the entire database. Each one of the above mentioned three types of uncertainty (attribute, record, or database) has its own probability space. In this section we provide a brief summary of probability spaces as introduced in [1]. To demonstrate the various probability spaces, we use the dataset shown in Table 1. In addition to the patient’s name, the dataset stores some vital signs of patients who are presently in the hospital’s intensive care unit. These vital signs are Temperature and Pulse-Rate (P_rate).

In this table, a star “*” is used to denote uncertain records. The “*” next to record r4 means that this record may or may not exist in the database (e.g., it is uncertain if the patient has been discharged from the ICU). On the other hand, a square bracket is used to enclose a set of probable values for an attribute. For example, the temperature in record r2 can be either 38 or 39. Similarly, pulse rate in record r3 can be either 75 or 85.

Table 1. Example Probabilistic Database

RID	Name	Temp	P_rate	
r1	Ahmed	38	91	
r2	Huda	[38,39]	85	
r3	Andy	39	[75,85]	
r4	Samir	37	70	*

2.1 Database states in probabilistic databases

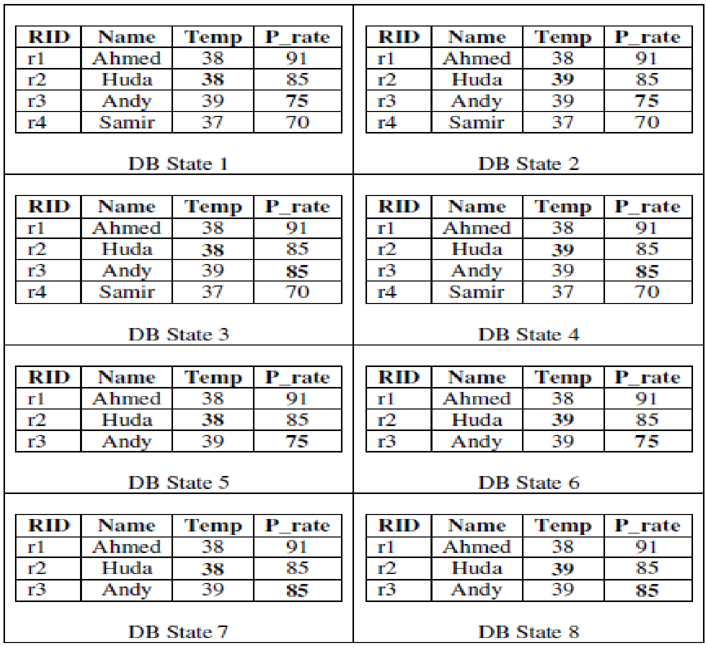
In database terminology, data that exist in the database at a particular moment in time is referred to as *database state*. In deterministic databases (databases that have no probabilistic data) there can be exactly one database state at any particular moment in time. On the other hand, in databases with uncertain data, there can be multiple possible database states at any given time. The number of possible database states depends on how many pieces of uncertain data there exist in the database. Table 2 shows all possible states corresponding to the data represented in Table 1. Each database state contains a unique combination of probabilistic data. Since there are two possible values for each of Andy's pulse rate and Huda's temperature and there are two possibilities for record r4 (i.e., that record may or may not exist in the database), the total number of possible states is 2 × 2 × 2 = 8. These eight states are shown in Table 2. However, the fact that there are multiple possible states at any moment in time does not mean that all these possible states are physically explicitly stored in the database. The representation shown in Table 2 is not a storage representation but a conceptual representation.

2.2 Attribute Probability Space

In the example shown in Table 1, we assume that the default is that probable values have equal probabilities. For example, the temperature of record r2 has a 0.5 probability of being 38 and a 0.5 probability of being 39. Similarly, the probability that record r4 exists in the database is 0.5 and the probability it does not exist is also 0.5.

In many cases the probability is not necessarily equally distributed. In these cases, explicit probability values need to be assigned. Table 3 is similar to Table 1, but with explicit probabilities assigned to uncertain attribute values. As shown in Table 3, each value is followed by a colon then its probability. For example, Andy's P_RATE has a 0.6 probability of being 75 and a 0.4 probability of being 85. Formally, we can express these probability assignments as follows.

Table 2. The Eight Probable DB States Corresponding to Table 1



RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	**38**	85
r3	Andy	39	**75**
r4	Samir	37	70

DB State 1

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	**39**	85
r3	Andy	39	**75**
r4	Samir	37	70

DB State 2

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	**38**	85
r3	Andy	39	**85**
r4	Samir	37	70

DB State 3

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	**39**	85
r3	Andy	39	**85**
r4	Samir	37	70

DB State 4

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	**38**	85
r3	Andy	39	**75**

DB State 5

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	**39**	85
r3	Andy	39	**75**

DB State 6

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	**38**	85
r3	Andy	39	**85**

DB State 7

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	**39**	85
r3	Andy	39	**85**

DB State 8

$P(r3.Pulse\_Rate = 75) = 0.6$
$P(r3.Pulse\_Rate = 85) = 0.4$

Based on probability theory, the sum of the probabilities of the set of possible attribute values should be one. In other words,
$P(r3.Pulse\_Rate = 75) + P(r3.Pulse\_Rate = 85) = 1.$

From probability theory, the set of possible simple events representing the outcomes of a repeatable experiment is referred to as a *sample space*. Borrowing from probability theory terminology, we use the term *attribute probability space (APS)* to refer to the set of possible attribute values along with their probabilities. Hence the data in the field r2.Temp of Table 3 is actually presented in attribute probability space. From probability theory, the sum of probabilities of the values in r2.Temp should add up to 1. In other words:

$P(r2.Temp = 38) + P(r2.Temp = 39) = 0.3 + 0.7 = 1$

The same applies to the data in the field r3.P_rate.

Table 3. Probabilistic Data with Assigned Probabilities

RID	**Name**	**Temp**	**P_rate**	
r1	Ahmed	38	91	
r2	Huda	[38:0.3,39:0.7]	85	
r3	Andy	39	[75:0.6,85:0.4]	
r4	Samir	37	70	0.8

2.3 Record Probability Space

Explicit probabilities can also be assigned to capture the uncertainty associated with the existence of records. Instead of using a "*" to indicate that the existence of record r4 is not certain as in Table 1, a specific probability value can be assigned. This is shown in Table 3 where a probability of 0.8 is used to indicate that the probability that r4 exists is 80%. We refer to this probability as *record existential probability*. Formally, we can state this as follows.

$P(r4) = 0.8$

Where P(r4) denotes the existential probability of record r4. The probability that a record does not exist and the probability that it exists should add up to one. Therefore,

$P(r4) + P(\neg r4) = 1$

Where P(¬r4) denotes the probability that record r4 does not exist in the database. To compute P(¬r4), we can rearrange the above equation to obtain:

$P(\neg r4) = 1 - P(r4) = 1- 0.8 = 0.2$

We refer to the set of different probabilities pertaining to the existence of a given record as *record probability space (RPS).* Therefore the *RPS* of record r4 consists of the set {P(r4) = 0.8, P(¬r4)= 0.2}. The sum of probabilities in a record probability space should be exactly one.

2.4 State probability space

Similar to the attribute probability space and record probability space, we also define *state probability space (SPS).* A state probability space represents the set of probable database states along with their probability values. For example, Table 2 shows the database state probability space corresponding to Table 1. Since in Table 1 equal probability distribution is assumed, the database states shown in Table 3 have equal probabilities.

The sum of probabilities of these different database states should be exactly one as represented by the following equation.

$$\sum_{i}^{n} P_i = 1$$

Where n is the total number of database states and P_i is the probability of state *i*. Since the eight states shown in Table 2 have equal probabilities, we can compute the probability of each state as follows.

$P_i = (1/8) = 0.125$

Briefly stated, in our representation of uncertain data we have introduced in [1] the definition of three different probability spaces. These probability spaces are:

- *Attribute probability space (APS)* which represents the set of possible values of an attribute along with their probabilities.

- *Record portability space (RPS)* which represents the different possible instances of a record along with their existential probabilities.

- *State probability space (SPS)* which represents the different possible database states and their probabilities.

3. MAPPING FROM APS TO RPS

In this section we introduce mapping techniques and describe how data can be mapped from attribute probability space to record probability space. Probabilistic attribute values in the *APS* can be mapped to records and existential probabilities in the *RPS*. The existential probabilities of these records are computed based on the probabilities that exist in the attribute probability space. Table 4 shows an equivalent relation to that of Table 3, but with data in the attribute probability space mapped to data in the record probability space. The way this mapping is achieved is as follows. There are two different possible Temperature values in record r2 of Table 3. Hence we map r2 to two different records, r2.1 and r2.2, in Table 4 where each record has one of the two Temperature values. The probability of that value in Table 3 becomes the existential probability of the corresponding record in Table 4. In other words, P(r2.1) in Table 4 is equal to P(r2.Temp = 38) of Table 3. The same approach applies to record r2.2 in Table 4.

Table 4. Data in Record Probability Space

RID	Name	Temp	P_rate	
r1	Ahmed	38	91	
r2.1	Huda	38	85	0.3
r2.2	Huda	39	85	0.7
r3.1	Andy	39	75	0.6
r3.2	Andy	39	85	0.4
r4	Samir	37	70	0.8

Record r3 in Table 3 is mapped to records r3.1 and r3.2 in Table 4 in a similar way but with respect to the P_rate possible values. In Table 4, we show the record existential probability next to each record. Record r1 has an existential probability of 100%, which is the default.

In addition to the simple mappings from APS to RPS shown in Table 4, there can be several other special cases. Below, we discuss three important cases.

3.1 Case 1: Mapping a record that has probabilistic values for more than one attribute.

An example of this case is demonstrated in Table 5. Attributes B and C both have multiple possible values for the first record (record r8), whereas attribute A has one deterministic value.

Table 5: More than one attribute have probabilistic values

ID	A	B	C
r8	a1	{b1: 0.4, b2:0.6}	{c1:0.3, c2:0.7}
r9	a9	b9	c9

Let T be a table in the attribute probability space that has *n* attributes. Let N_1, N_2, …., N_n represent the number of probabilistic values for each of the attributes for a given record *r* in T (note that in this discussion we are not including the ID attribute). We need to map record *r* from the APS to a number of records in RPS. Let N_r be the number of records in RPS that correspond to *r* in APS. N_r can be computed using the following formula.

$$N_r = N_1 * N_2 * .... * N_n$$

Or

$$N_r = \prod_{i=1}^{n} N_i$$

Where i is the attribute number in table T.

By applying the above formulae to record r8 of Table 5, we can obtain N_{r8} , the number of probabilistic records in RPS corresponding to *r8* in APS.

$$N_{r8} = N_A * N_B * N_C = 1 * 2 * 2 = 4$$

Where N_A, N_B, and N_C represent the number of probabilistic values for attributes A, B, and C respectively, for record *r8*. These four records in the RPS are shown in Table 6. The existential probability of each record is shown next to it.

Table 6: data in Table 5 mapped to RPS

ID	A	B	C	
r8.1	a1	b1	c1	0.12
r8.2	a1	b1	c2	0.28
r8.3	a1	b2	c1	0.18
r8.4	a1	b2	c2	0.42
r9	a9	b9	c9	

The record existential probability is obtained by multiplying the probabilities of the attribute values appearing in that record. For example, record *r8.1* has the two values b1 and c1 whose probabilities in Table 5 are 0.4 and 0.3 respectively. Therefore,

$$P(r8.1) = 0.3 * 0.4 = 0.12$$

The same approach is used to compute the existential probabilities of records *r8.2, r8.3,* and *r8.4* of Table 6.

3.2 Case 2: Probabilistic Values and Null Values.

In this case an attribute in table T in the APS may have several possible values, but one of the values is Null. This is represented in the APS by having a set of possible values for an attribute, but the sum of their probabilities is less than 1. For example in Table 7, attribute E for record *r5* has two possible values e1 and e2 with probabilities of 0.4 and 0.5, respectively. Since the sum of these probabilities is less than 1, this indicates that Null is a possible value. The probability of the Null value completes the sum of probabilities to 1. Therefore we can compute the probability of a Null value for attribute *r5.E* as follows.

$$P(r5.E \ \ IS\ NULL) = 1 - (0.4 + 0.5) = 0.1$$

Table 7: Representing a NULL value for an attribute

ID	D	E
r5	d1	{e1: 0.4, e2:0.5}
r6	d2	e9

When mapping record r5 to the RPS, we need to take into consideration this Null value just like any other possible value. Table 8 shows the mapping of Table 7 from the APS to the RPS. Record *r5.3* has a Null value for E and the existential probability of this record is 0.1, which is the probability *P(r5.E IS NULL)* in Table 7.

Table 8: Mapping Data in Table 7 to RPS

ID	D	E	
r5.1	d1	e1	0.4
r5.2	d1	e2	0.5
r5.3	d1		0.1
r6	d2	e9	

3.3 Case 3: A Record has Existential Probability as well as an Attribute-Level Uncertainty

In some cases the record existence is uncertain and, at the same time, some attribute values within the same record are probabilistic. To demonstrate this case, assume that the existence of record r3 is uncertain with existential probability of 0.9 as shown in Table 9. Also assume p_rate probabilistic values [75:0.6, 85:0.4] exist in the same record. We interpret this to mean that: *if the record exists*, then the P_rate values of 75 and 85 for that record have probabilities 0.6 and 0.4 respectively.

Table 9. Probabilistic data and probabilistic records

RID	Name	Temp	P_rate	
r1	Ahmed	38	91	
r2	Huda	[38:0.3,39:0.7]	85	
r3	Andy	39	[75:0.6,85:0.4]	0.9
r4	Samir	37	70	0.8

Table 10 below shows a mapping of r3 in Table 9 which is in APS to two records *r3.1* and *r3.2* in RPS. In Table 10, the record existential probabilities for *r3.1* and *r3.2* are computed by multiplying the record existential probability of r3 in Table 9 and the attribute value probability as shown below.

$$P(r3.1) = P(r3) * P(r3.P\_rate = 75) = 0.9 * 0.6 = 0.54$$
$$P(r3.2) = P(r3) * P(r3.P\_rate = 85) = 0.9 * 0.4 = 0.36$$

Table 10. Mapping from APS to RPS

RID	Name	Temp	P_rate	
r1	Ahmed	38	91	
r2	Huda	[38:0.3,39:0.7]	85	
r3.1	Andy	39	75	0.54
r3.2	Andy	39	85	0.36
r4	Samir	37	70	0.8

To re-compute P(¬r3) from Table 10 information, the following substitutions can be made.

$$P(\neg r3) = 1 – [P(r3.1) + P(r3.2)] = 1 – (0.54 + 0.36) = 1 – 0.9 = 0.1$$

4. MAPPING RPS TO SPS

In addition to the mapping of data from attribute probability space to record probability space, we can map probabilistic data from record probability space to database state probability space. This requires us to identify all possible database states and the probability associated with each state. The database state probabilities can be computed based on the record existential probabilities that exist in the record probability space.

To demonstrate how this mapping can be achieved, we map the data of Table 4 (which is in RPS) to state probability space. After the mapping is complete, we obtain the state probability space representation shown in Table 11. Note that Table 11 is different from Table 3 in that the database states do not have equal probabilities. In what follows we describe how the mapping is performed and how the probabilities of the database states are computed.

Each database state in Table 11 contains a unique combinations of records appearing in Table 4. Database state 1 in Table 11 contains records r1, r2.1, r3.1 and r4 from Table 4. Database state 2, on the other hand, contains records r1, r2.2, r3.1, and r4 from Table 4. Database states 3 and 4 are constructed in a similar way. Database states 5, 6, 7, and 8 are similar to the first four database states except that record r4 in Table 4 is not included in these states. This is because record r4 has a 0.2 probability of not existing. In other words, the first four database states in Table 11 represent the fact that record r4 exists with 0.8 probability, whereas the last four database states reflect the fact that record r4 has a 0.2 probability of not existing.

To compute the probability of the first database state in Table 11, we multiply the existential probabilities of the records appearing in that database state. The existential probability of record r2.1 in Table 4, which is appearing as record r2 in the first database state in Table 11, is 0.3. Similarly records r3.1 and r4 in Table 4 are appearing in the first database state of Table 11 as records r3 and r4. The existential probabilities of these two records in Table 4 are 0.6 and 0.8, respectively. By multiplying these three probabilities, we obtain the probability of the first database state as follows.

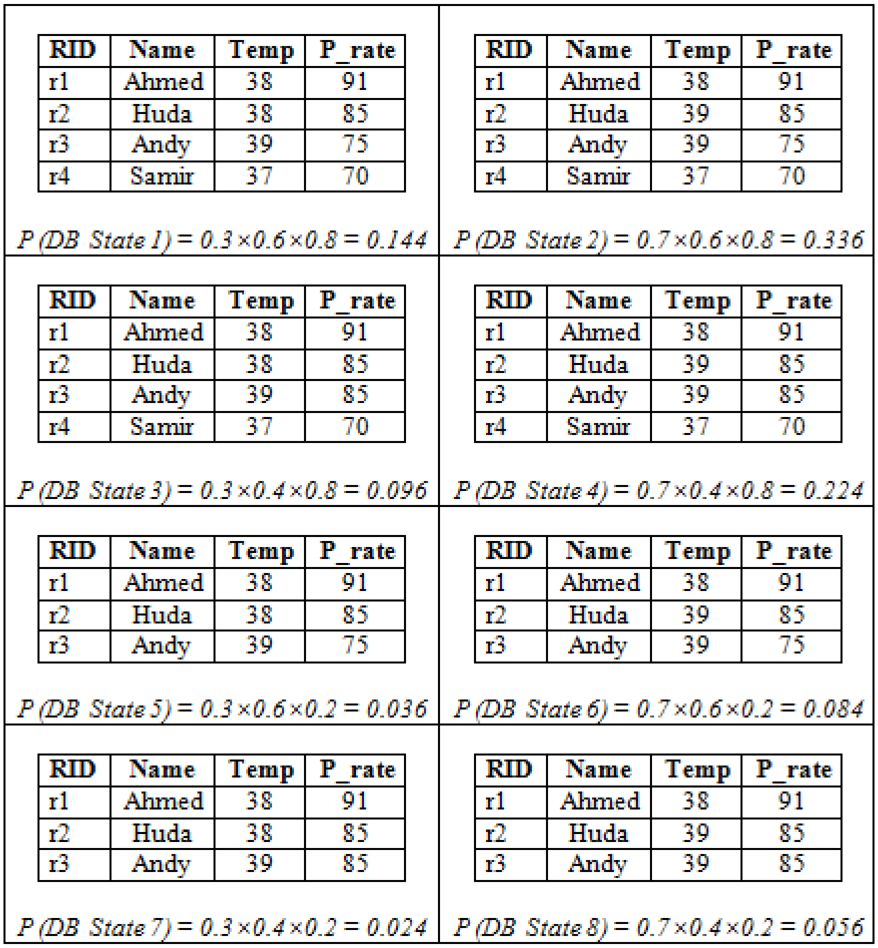
$$P\ (\text{DB State } 1) = P(r2.1) \times P(r3.1) \times P(r4) = 0.3 \times 0.6 \times 0.8 = 0.144$$

The probabilities of database states 2, 3, and 4 are computed in a similar way. In Table 11, we show the computation of the probability of each database state based on the probabilistic records appearing in in that state.

The probability of database state 5 is computed in a way similar to database state 1, except that we take into consideration the probability that record r4 may not exist ($P(\neg r4)$) in the database, which is 0.2. Therefore,

$$P\ (\text{DB State } 5) = P(r2.1) \times P(r3.1) \times P(\neg r4) = 0.3 \times 0.6 \times 0.2 = 0.036$$

The probabilities of the remaining database states 6, 7, and 8 are computed in a way similar to that of record r5, where we use $P(\neg r4)$.



RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	38	85
r3	Andy	39	75
r4	Samir	37	70

P (DB State 1) = 0.3×0.6×0.8 = 0.144

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	39	85
r3	Andy	39	75
r4	Samir	37	70

P (DB State 2) = 0.7×0.6×0.8 = 0.336

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	38	85
r3	Andy	39	85
r4	Samir	37	70

P (DB State 3) = 0.3×0.4×0.8 = 0.096

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	39	85
r3	Andy	39	85
r4	Samir	37	70

P (DB State 4) = 0.7×0.4×0.8 = 0.224

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	38	85
r3	Andy	39	75

P (DB State 5) = 0.3×0.6×0.2 = 0.036

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	39	85
r3	Andy	39	75

P (DB State 6) = 0.7×0.6×0.2 = 0.084

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	38	85
r3	Andy	39	85

P (DB State 7) = 0.3×0.4×0.2 = 0.024

RID	Name	Temp	P_rate
r1	Ahmed	38	91
r2	Huda	39	85
r3	Andy	39	85

P (DB State 8) = 0.7×0.4×0.2 = 0.056

5. CONCLUSION

Storing and managing probabilistic data in database systems have become a very much-needed capability in many advanced applications. Probabilistic data can be represented at different probability spaces. Each probability space corresponds to a level of granularity of the data. The three levels that we used are: attribute probability space, record probability space, and database state probability state.

As a consequence of this representation, performing correct mappings between these probability spaces become significant. Sound data mapping techniques are needed to avoid any loss of information or data integrity in the mapping process. In this paper we have described, first, how data can be mapped from attribute probability space to record probability space. In addition to the overall mapping approach, we described several special mapping cases. Second, we described how data can be mapped from record probability space to database state probability space. We demonstrated with examples how the probability of each database state can be computed based on the probabilities that exist in the record probability space.

ACKNOWLEDGEMENT

The author is grateful to the Applied Science Private University, Amman, Jordan, for the full financial support granted to this research project (Grant No. DRGS-2014-2015-77).

REFERENCES

[1] A. Alashqur, "Probability Spaces in Databases with Uncertainty," International Journal of Engineering Science and Technology, Vol. 5, Issue 12, December, 2013.

[2] N. N. Dalvi, C. Ré, and D. Suciu (2009). "Probabilistic databases: diamonds in the dirt". Communication of the ACM, 52(7):86-94.

[3] N. N. Dalvi, K. Schnaitter, and D. Suciu (2010). "Computing query probability with incidence algebras". In PODS, pages 203-214.

[4] R. Fink, D. Olteanu, and S. Rath (2011). "Providing Support for Full Relational Algebra in Probabilistic Databases". In ICDE, pages 315-326.

[5] A. Jha, D. Olteanu, and D. Suciu (2010). "Bridging the Gap Between Intensional And Extensional Query Evaluation In Probabilistic Databases". In EDBT, pages 323-334.

[6] C. Koch (2009) "On Query Algebras for Probabilistic Databases". SIGMOD Rec., 37:78-85, March.

[7] J. Li, B. Saha, and A. Deshpande (2009) "A Unified Approach to Ranking in Probabilistic Databases". PVLDB, 2(1):502-513.

[8] P. Bosc and O. Pivert (2010). "Modeling and querying uncertain relational databases: A survey of approaches based on the possible worlds semantics", International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 18(5):565-603.

[9] L. Antova, C. Koch, and D. Olteanu (2007). "MayBMS: Managing Incomplete Information with Probabilistic World-Set Decompositions". In: Proc. of Intl. Conf. on Data Engineering (ICDE).

[10] J. Boulos, N. Dalvi, B. Mandhani, S. Mathur, C. Re, and D. Suciu (2005) "MYSTIQ: a system for finding more answers by using probabilities". In: Proc. of ACM SIGMOD Intl. Conf. on Management of Data.

[11] L. V. S. Lakshmanan, N. Leone, R. Ross, and V. Subrahmanian (1997). "ProbView: A Flexible Probabilistic Database System". ACM Transactions on Database Systems (TODS), Vol. 22, No. 3, pp. 419–469.

[12] R. Cheng, S. Singh, and S. Prabhakar (2005). "U-DBMS: A Database System for Managing Constantly-Evolving Data" In: Proc. of Conf. on Very Large Data Bases (VLDB).

[13] R. Jampani, L. Perez, M. Wu, F. Xu, C. Jermaine, and P. J. Haas (2008). "MCDB: A Monte Carlo Approach to Managing Uncertain Data". In: Proc. of ACM SIGMOD Intl. Conf. on Management of Data.

[14] D. Z. Wang, E. Michelakis, M. Garofalakis, and J. M. Hellerstein (2008). “BayesStore: Managing Large, Uncertain Data Repositories with Probabilistic Graphical Models”. In: Proc. of Conf. on Very Large Data Bases (VLDB).

[15] D. Olteanu, J. Huang, and C. Koch (2010). “Approximate Confidence Computation in Probabilistic Databases”. In ICDE, pages 145-156.

[16] C. Re and D. Suciu (2009). “The Trichotomy of HAVING Queries on a Probabilistic Database”. VLDB J., 18(5):1091-1116.

[17] S. Roy, V. Perduca, and V. Tannen (2011) “Faster Query Answering in Probabilistic Databases Using Read-Once Functions”. In ICDT, pages 232-243.

[18] P. Sen, A. Deshpande, and L. Getoor (2010). “Read-once Functions and Query Evaluation In Probabilistic Databases”. PVLDB, 3(1):1068-1079.

[19] M. A. Soliman, I. F. Ilyas, and S. Ben-David (2010). “Supporting Ranking Queries on Uncertain And Incomplete Data”. VLDB J., 19(4):477-501.

[20] L. Antova, C. Koch, and D. Olteanu (2009). “10^(10^6) worlds and beyond: Efficient representation and processing of incomplete information”. VLDB J. 18(5), 1021-1040.

AUTHOR

Abdallah Alashqur is currently an associate professor at the Applied Science University in Amman, Jordan. Dr. Alashqur holds a Master’s degree and a Ph.D. degree from the University of Florida. After obtaining his Ph.D. degree in 1989, he worked for around seventeen years in the USA (in industry) followed by nine years in Jordan (in academia). His research is mainly in the area of data mining and database systems.