Learning Probabilistic Relational Models based Aggregation

By
Dr. Saleh Al-Qaraawy Dr. Jane Jaleel Stephan

And Miss Lamia Abed Noor

Abstract

Aggregation is a tool that is used in order to present the multiple instances of an individual attribute in a single value that characterizes the groups represent. However, probabilistic relational models are constructed from relational data base; these data are interrelated with different cardinalities so it is need for aggregation in some situations in order to convert the relation cardinality from “many” to “one”. This paper will discuss the learning probabilistic relational models that adopt aggregation and the suitable procedure that is required for this task and describe it through a practical application on test data that would be explained in this paper.

Keywords: Aggregation, Probabilistic Relational Model

1. Introduction

The construction of Probabilistic Relational Models (PRMs) is done using raw data through learning process. Where the input of this process is relational data base that contains the interested data and the output is a model which encodes the dependencies between the features of input data, descriptive attributes of the entities in data base represent some features.

The learning process consists of different steps and in result the learning is divided into different stations. The initial step is to locate the features and find the suitable forms of them and in result it facilitates the next step that seeks about the dependencies between them and so on.

The descriptive attributes as features exist in relational data base are spread on different tables; these attributes can be related through multiple tables that have relations cardinalities such as one-one, one-many, and

many-many. At the same time, the searching for dependencies between attributes is performed as one-to-one, so this means the “many” must be converted into single, aggregation is the tool that convert “many” to single values that as possible keep the features of “many”.

2. Probabilistic Relational Models(PRMs)

Probabilistic relational models (PRMs) are probabilistic graphical models that are used with the relational database, extended on the traditional models that are standard attribute-based Bayesian network representation with flat table. PRMs, in contrast to the traditional models devoted to the attributes of a single table, exploit the structure of relational database in addition to the attributes that they contain in order to extract the regular relationships of interesting data and formulate them in suitable forms that are represented by probabilistic models.

Construction of PRM is based on the existence of possible dependencies between different types of attributes that are belong to different entities, are stored in different tables.

It is possible to search the dependency between attributes in the same table or between attributes in different tables, whether they are connected with cardinality relations; many or one-to-one.

3. Learning Probabilistic Relational Models

The probabilistic relational models are induced from the observed data that are stored in relational data set. As we know the interesting data are not trivial, so it is impossible that we can build these models by hand. Therefore, the approaches have been used to perform this task, are automated ways that are called learning.

However PRMs are extended Bayesian network, the learner procedure for PRM is not different from learning methods of Bayesian network except the steps that are concerned with the relation distinction. One central question is how to appropriately summarize the complex structure of relational data in ways that are useful to a learning algorithm.

The space hypothesis of PRMs must be prepared to assemble random variables and their parent that can represent different elements; propositional attributes, relational attributes, existence of link, and aggregate values that are constructed from aggregate function.

The candidate structures are the possible structure that can be evaluated and then may be returned by the learner procedure [33], and constitute the hypothesis space. Construction hypothesis space is an important step to generate the possible structures using the training data. The elements used to form the structure are nodes. Therefore, typically all the possible nodes that hold random variables must be available in the hypothesis space framework. Then the structure would be established by adding edges or deleting others and computed CPD for each node has a new parent.

An evaluation measures serve to assess the quality of a given candidate graphical model a given database of samples cases, so that it can be determined which of a set of candidate graphical models best fits the given data[26]. A desirable property of an evaluation measure is decomposability i.e. the total network quality should be computable as sum or product of local scores, while local score is concerned with pairs (child | parents). This can enable to maximize the quality of each pair independently then in result the total network.

The evaluation measures are split into two classes, first the measures that are concerned with determining the structures of model by allocating the nodes and arcs, and the other are concerned with the evaluation the best parameters of models.

Most evaluation measures of structure model are based on measures of dependence, it is necessary to measure the strength of dependence of two or more variables, either in order to test of conditional independence or in order to find the strongest dependences. One of the dependences measures is Mutual Information (MI); it can be used to measure the strength of interaction between two or more variables, so it has a threshold (0.001) [27]. If MI for random variable is over this threshold, these variables have some dependence, else the variables are independent. The random variable with high score of dependence can be introduced in the candidates' structures.

4. Aggregate and PRMs

The relational model in relational database introduced has rich information about associations. The relationships that connect between different tables are identified as one-many or many- many, more features can be added to be exploited in the dependent network using a suitable function that is called aggregate function. However, the aggregate functions are varied; the features provided will be different. They are used in order to introduce a new value which describes multiple values.

Aggregate functions are functions on sets of records. Given a set of records and some instance of an aggregate function, a feature of the set which summarizes the set is computed. Then, aggregates are used to characterize the structural information that is stored in tables and

associations between them. Consider two tables P and Q, linked by a one-to-many association A. For each record in P, there are multiple records in Q, and A defines a grouping over Q. An aggregate function can now be used to describe each group by a single value. Because a single value is specified for every record in P, we can think of the aggregate as a virtual attribute of P[37]. As result, P will have additional attribute and can be searched for existence of its interaction with other attributes in P.

The aggregate functions are classified according to the kind of the aggregated values, the continuous values; MAX, MIN, MEAN and so on. While the discrete values, “MODE” is the conventional function, which be used in order to summarize multiple discrete values in one through selection of the state that have higher frequency. Figure(1) describes an example of relation database that are specified to university, each record in relation “Lecture” may connect to one or more records in relation “Course”. Therefore the values of attribute “course_type” in Course must be converted from multiple values, which are related with one value in “Lecturer” relation to single value in order to enable it with face to attributes in “Lecturer” relation and introduce the summarized values in virtual attribute that can be imagined as additional attribute in relation “Lecture” as shown in Figure(2).

Course Relation

Course code	Course type	Lecturer code
C_0	Software Engineering	L_0
C_1	Programming	L_0
C_2	Programming	L_0
C_3	Programming	L_0
C_4	Programming	L_1
C_5	Data mining	L_1
C_6	Software engineering	L_2
C_7	Operating System	L_3
C_8	Communication	L_3
C_9	Programming	L_2
C-10	Data Mining	L_1
C_11	Programming	L_3
C_12	Software Engineering	L_2

Lecture Relation

Lecturer_code	Lecture_ name
L_0	X
L_1	Y
L_2	Z
L_3	M

Figure(3.3) university data set

Lecturer Relation

Lecturer_code	Lecturer_name	Aggregate “Course_type”
L_0	X	Programming
L_1	Y	Data Mining
L_2	Z	Software Engineering
L_3	M	Programming

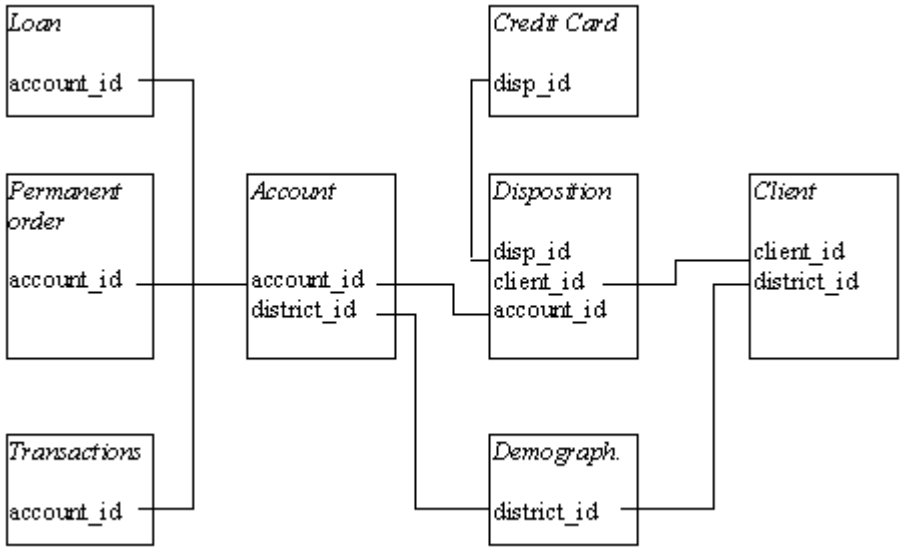
Figure(3.4) Lecturer Relation with “virtual attribute” hold aggregate values result from attribute”course_type” in “Course” relation

5. Practical Framework Learning PRMs Based Aggregate

The practical steps that must be applied in order to get the hypothesis space of local structures; are required to construct the PRM based aggregate.

5.1 Expirement Data Description

The experiment data we use are from PKDD'99 Discovery Challenge "Guide to the Financial Data Set" and are given free for research purposes[43] as shown in Figure(4.1).



Figure(3) Relational Structure Schema Of Experimented data

aprocedure for learning the PRMs based aggregation contains of several steps that must be achieved in order to get the hypothesis space of

The seeking for random variables in order to formulate the hypothesis space of candidate models in order to create PRMs is considered the main task of learning process, different possible values can be attached to them; description attributes, existence of possible

objects, links between different relations, and aggregate descriptive attributes.

The learning of PRMs aggregate based models require to prepare and care the hypothesis space that contain the random variables, have aggregate attributes so the early step in such learning process is the extraction the suitable aggregate values of specified attributes and present as a virtual attribute and then complete the learning process will be done in similar way to other probabilistic models.

Basically, Aggregation is the approach used to deal with the attributes across different relations with this type of relationships and aggregate functions are suggested to conclude information and convert into single value according to some features that functions perform. PRMs learning procedure formulates the resulted aggregate values in virtual attribute, which would be joined to the opposite relation. The new virtual attributes would be tested in order to reveal any dependencies with the attributes that are located in the opposite relation. The pairs of attributes (virtual and opposite) are input into dependency tests. Then the pair that overcomes threshold value of specified measure will be candidates to be into other tests that construct the model through training step. The virtual attributes holds aggregate values are dealt as same as the original attributes so the learning steps of model are the same in more of them for both types of attributes. Algorithm(1) contains the required steps to aggregate the interested attributes in one relation and formulate them into virtual attributes that holds aggregate values and join them to opposite relation.

Algorithm(1) Construction Virtual Attribute by Aggregation

Input : Two Relations X,Y are connected with cardinality(One_to_Many)
Output : Virtual attribute(V) would be joined with X as X.V
Processing:

1- Let Y.A be discrete attribute in Y to be aggregeated
2- Grouping Data(D) in Y relation into bags of record (Di), where the