Attribute-Oriented Induction based Data Generalization for Relational Databases

Zin May Oo, Dr. Sabai Phyu
University of Computer Studies, Yangon
zinmayoo11@gmail.com

Abstract

Data Mining is the task of discovering interesting patterns from large amounts of data where the data can be stored in relational databases. Data generalization is a process that abstracts a large set of data in a database from a relatively low level to higher conceptual levels. The attribute-oriented induction (AOI) is a data mining technique to extract generalized knowledge from relational databases and the user's background knowledge. AOI facilitates users in examining the general behavior of the data. AOI allows the users to view the data at more meaningful abstractions compared to the primitive level.

Keywords: Data Mining, Relational database, Data generalization, Attribute-oriented induction

1. Introduction

Data mining refers to extracting or mining knowledge from large amounts of data. Data mining can be classified into two categories: descriptive data mining and predictive data mining. Descriptive data mining describes the data set in a concise and summative manner and presents interesting general properties of the data. Predictive data mining analyzes the data in order to construct one or a set of models, and attempts to predict the behavior of new data sets. Data generalization is a descriptive data mining technique.

Many data mining approaches have already been proposed in discovering useful knowledge. One of the most important of these approaches is the attribute-oriented induction (AOI) method. Data generalization is a fundamental element of the attribute-oriented induction [1].

AOI is a set-oriented database mining method which transforms data stored in database relations into more general information on an attribute basis. AOI summarizes the information in a relational database by repeatedly replacing specific attribute values with more general concepts according to user defined concept hierarchies. Integrating with concept hierarchies, the AOI method can induce multi-level generalized knowledge, which can provide decision making process.

2. Related Work

There are many other related works in this field. The [1] represents two efficient methods, generalized positive knowledge induction and generalized negative knowledge induction. These methods provide a simple and efficient way for knowledge generalization from a relational database. The [2] proposes that AOI is a descriptive database mining technique allowing such a transformation. The [3] represents some algorithms for automatic generation of concept hierarchies for numerical attributes and for dynamic refinement of a given or generated concept hierarchy. The [4] demonstrates the effectiveness and efficiency of AOI methods in knowledge discovery in relational databases. Hoi-Yee Hwang and Wai-Chee Fu proposed the generalization step and an improved algorithm of O(N) in [5]. In [6], Liu Deqin and Ma Weijun proposed the fundamental principle in AOI that is to generalize the initial relation to a prime relation and then to a final relation using background knowledge and user-defined threshold. The [7] proposes a novel star schema attribute induction as a new attribute induction paradigm and as improving from current attribute oriented induction. The [8] represents the characteristics of object-oriented database and extends the AOI method to object-oriented paradigms, focusing on handling complex attributes, and present an algorithm for learning characteristic rules in an object-oriented database. The [9] represents an overall picture of the Data Mining field from a database researcher's point of view, introducing interesting data mining techniques and systems, and discussing applications and research directions.

3. Overview of the System

AOI is to be useful in abstraction large set of task relevant data in a database. The essential background knowledge applied in AOI is concept hierarchy associated with each attribute in the relational databases. Concept hierarchies help expressing knowledge and data relationships in databases in concise, high level terms, and thus, play an important role in knowledge discovery process [3]. Different levels of concepts can be organized into taxonomy of concepts. The concepts in taxonomy can be partially ordered according to general-to-specific ordering [8].

Each concept hierarchy often refers to as domain knowledge, and stores relationships of specific concepts and generalized concepts. Concept hierarchy can be specified based on the relationship among database attributes or by set groupings and be stored in the form of relations in the same database [7].

The following three important roles are played by concept hierarchies in attribute-oriented induction.

1. Concept hierarchies should be used to map the high level concepts into the constants at the concept level(s) matching those stored in the database.
2. Concept hierarchies should be used for concept tree climbing in the generalization process.
3. Concept hierarchies will be used for further ascension of the concepts in the generalized relation in order to achieve desired generalization results.

3.1 Concept Hierarchy Generation for Numerical Data

In this system, Intuitive Partition method is used for concept hierarchy generation for numerical data. In this system, a simply 3-4-5 rule can be used to segment numerical data into relatively uniform, “natural” intervals using intuitive partition. If an interval covers 3,6,7 or 9 distinct values at the most significant digit, partition the range into 3-equals width intervals. If it covers 2,4, or 8 distinct values at the most significant digit, partition the range into 4 equal-width intervals. If it covers 1,5, or 10 distinct values at the most significant digit, partition the range into 5 equal-width intervals.

3.2 Concept Hierarchy Generation for Categorical Data

It performs the specification of a set of attributes, but not of their partial ordering. A concept hierarchy can be automatically generated based on the number of distinct values per attribute in the given attribute set. The most distinct values are placed at the lowest level.

3.3 Methods of AOI based Data Generalization

The idea of AOI is to perform generalization based on the examination of the number of distinct values of each attribute in the relevant set of data. Data generalization is performed by either attribute removal or attribute generalization.

- Attribute removal

  If there is a large set of distinct values for an attribute of the initial working relation, but either there is no higher level on the attribute or its higher-level concepts are expressed in terms of other attributes, then the attribute should be removed from the working relation.
- Attribute generalization

  If there is a large set of distinct values for an attribute in the initial working relation and there exists a set of generalization operators on the attribute then generalize the attribute [4].

3.4 Two approaches to control AOI based Data Generalization

Data generalization is controlled by two approaches: the attribute generalization threshold and the generalized relation threshold. These two approaches can be applied to generate a set of generalized tuples to describe the target relation.

- Attribute generalization threshold control
  - sets one generalization threshold for each attribute.
  - if the number of distinct values in an attribute is greater than the attribute threshold, attribute removal or attribute generalization should be performed.
- Generalized relation threshold control
  - sets a threshold for the generalized relation.
  - if the number of (distinct) tuples in the generalized relation is greater than the threshold, further generalization should be performed.
  - by further generalization on selected attribute(s) and merging of identical tuples, the size of the generalized relation will be reduced [5].

4. Attribute-Oriented Induction Algorithm

Input: (i) A relational database (ii) a data mining query (iii) a list of attributes (iv) a set of concept hierarchies (v) generalization thresholds.
Output: A generalized relation.

Method: The method is outlined as follows.
Step 1. Collect the task-relevant data,
Step 2. Prepare for generalization, and
Step 3. Generalize on relevant data

Notice that Step 2 and Step 3 are performed as follows.

```
begin
 for each attribute $A_i$ (1 <= i <= n, where n = # of
          attributes) in the generalized relation (GR) do
   if #_of_distinct_values_in_$A_i$ > threshold     {
      if   no higher level concept in the concept hierarchy
           table for $A_i$
      then remove $A_i$
      else substitute for the values of the $A_i$'s by its
           corresponding minimal generalized concept;
           merge identical tuples }
  while #_of_tuples in GR > threshold do
      {
          selectively generalize attributes;
          merge identical tuples
      }
end [5].
```

5. Design and Implementation of the System

The design and implementation of the system is shown as follows.

5.1 System Flow Diagram of the System

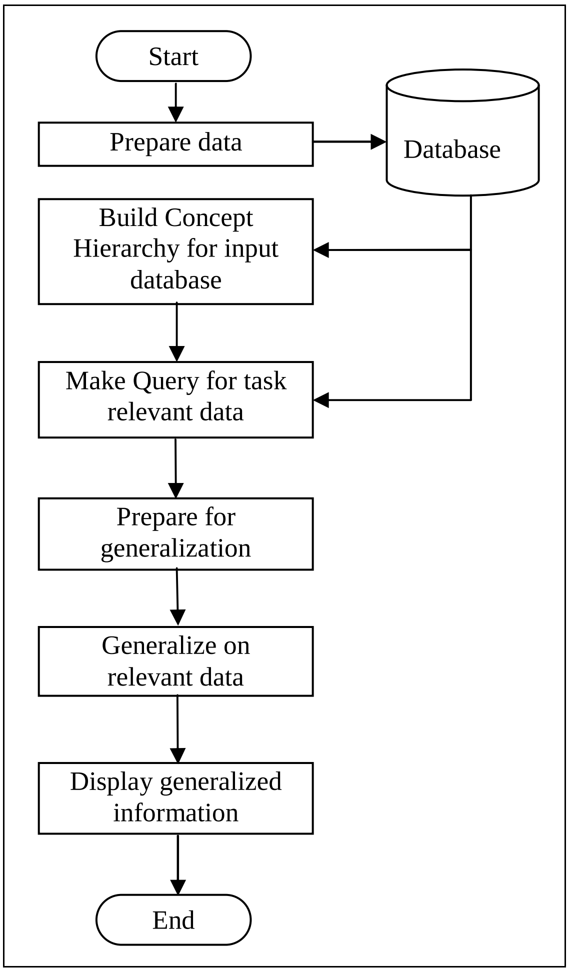
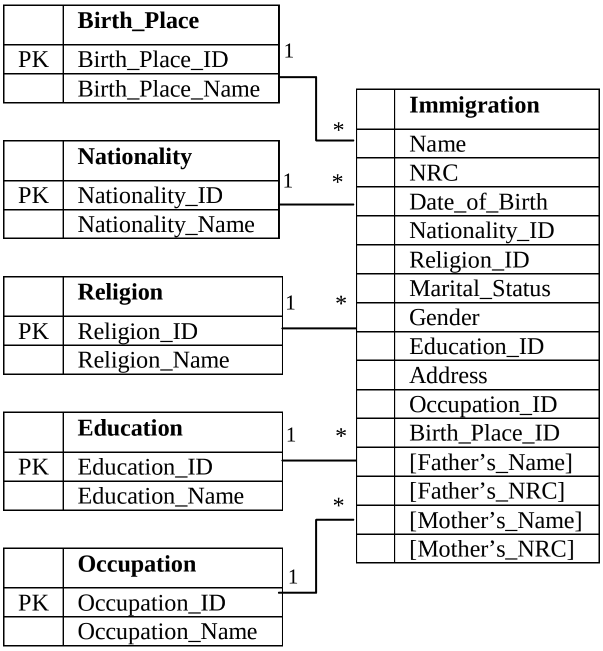

Start
Prepare data
Database
Build Concept Hierarchy for input database
Make Query for task relevant data
Prepare for generalization
Generalize on relevant data
Display generalized information
End

Figure 1. System flow diagram of the System

Firstly, this system builds concept hierarchy based on the examination of the number of distinct values of each attribute in the importing database. And then prepares query for task relevant data and the generalization process is performed. Finally, this system displays generalized information.

5.2 Database Design of the System

This system has the main database named "Immigration" in SQL Server(2005) or Microsoft Access. Immigration database contains six tables. "Immigration" table has one-to-many relationship with "Nationality" table, "Religion" table, "Education" table, "Occupation" table and "Birth_Place" table. The relationship of these tables is shown in Figure 2.



	Birth_Place
PK	Birth_Place_ID
	Birth_Place_Name

	Nationality
PK	Nationality_ID
	Nationality_Name

	Religion
PK	Religion_ID
	Religion_Name

	Education
PK	Education_ID
	Education_Name

	Occupation
PK	Occupation_ID
	Occupation_Name

	Immigration
	Name
	NRC
	Date_of_Birth
	Nationality_ID
	Religion_ID
	Marital_Status
	Gender
	Education_ID
	Address
	Occupation_ID
	Birth_Place_ID
	[Father's_Name]
	[Father's_NRC]
	[Mother's_Name]
	[Mother's_NRC]

Figure 2. Database Design of the System

5.3 Implementation and Experimental Results of the System

AOI method basically involves three primitives. These are collection of initial task-relevant data (Data Collection), use of background knowledge (domain knowledge) during the mining process and representation of the data result [6].

This system uses Immigration database but allows any databases to import to the system at run time. After importing the database, this system will build concepts hierarchy tree as shown in Figure 3. To build concepts hierarchy tree, first choose attributes that is to be added to tree. String and Number data type is used to build concepts hierarchy.

There are two processes to build concepts hierarchy:

- Automatic
- Existing

In Automatic process, concept hierarchy tree is automatically built for choosing attributes. In Existing process, concept hierarchy tree for choosing attribute is chosen from existing database.

{Shan, Myanmar, Chin, Yakhine, KaChin, Mon, Kayin, Kayar} Є Native

{Myanmar + Chinese, Myanmar +Indian, Myanmar + Pakistan, Mon + Pakistan, Chin + Pakistan , Yakhine + Pakistan, Yakhine + Indian , KaChin + Pakistan, Kayin + Chinese, Shan + Chinese, Kayin + Pakistan, KaChin + Chinese} Є Sino

{Pakistan, Indian, Chinese} Є Foreign

{Native, Sino, Foreign} Є Nationality

{Islam, Buddhism, Christian, Hindu} Є Religion

{Grade 1, Grade 2, Grade 3, Grade 4, Grade 5} Є Primary

{Grade 6, Grade 7, Grade 8, Grade 9} Є Middle

{Grade 10, Grade 11} ∈ High
{First Year, Second Year, Third Year, A.G.T.I} ∈ Undergraduate
{B.A, B.C.Sc, B.C.Sc(Hons), M.C.Sc, B.Tech, B.E, B.Sc, B.Sc(Hons), M.Sc, B.A(Hons), M.A, M.E } ∈ Graduate
{Primary, Middle, High, Undergraduate, Graduate} ∈ Education
{Shopkeeper, Tailor, Baker, Programmer, Trader, Artist, Photographer, Writer, Carpenter, Designer, Farmer} ∈ Business
{Teacher, Office Staff, Fireman, Postman, Policeman, Engineer} ∈ Staff
{Dependent, Student, Business, Staff} ∈ Occupation

Figure 3. A Concept Hierarchy table of the Immigration Database

After performing concept hierarchy tree, prepare query to collect task relevant data for performing generalization process. The user can choose the attributes and the value of that attributes by using column filter. The user can choose "and" or "or" in column filter. This system automatically performs query operation based on user-selection.

Table 1. Initial working relation: a collection of task-relevant data

Name	Nationality	Religion	Gender
Lin Lin Oo	Shan	Buddhism	M
Khin Zaw	Myanmar	Christian	M
Nwe Ni Win	Myanmar	Buddhism	F
Hla Aye	KaChin	Buddhism	F
Mie Mie	Myanmar	Buddhism	F
Tin Tun	Myanmar	Buddhism	M
Aye Aye	Myanmar	Buddhism	F
Phyu Phyu Nwe	Kayar	Buddhism	F
Khin Khin Phyu	Kayin	Christian	F

In Table 1, the user selects Name, Nationality, Religion and Gender attributes to perform generalization process and attributes' value are filtered by where clause such that Education is equal to Grade 1.

And then the system allows users to set the attribute threshold values to perform the first generalization control approach, called attribute generalization threshold control.

Table 2. Specifying attributes generalization threshold

Attribute Name	Distinct Values	Attribute Threshold
Name	9	9
Nationality	5	3
Religion	2	1
Gender	2	2

In Table 2, the user sets the threshold value for each attribute to replace lower level hierarchy tree values to upper level hierarchy tree values. If a user feels that the generalization reaches too high a level for a particular attribute, the threshold must be increased. This corresponds to drilling down along the attribute. Also, to further generalize a relation, the user must reduce the threshold of a particular attribute which corresponds to rolling up along the attribute. If the user doesn't want to perform the generalization process for the attribute, the user must set the equal attribute threshold value with the distinct value of the attribute.

The distinct value is the number of distinct values in the attribute. According to setting the threshold value for each attribute in Table 2, the system determine whether attribute should be removed, and if not, compute its minimum desired level in corresponding concept hierarchy. For Name attribute, the distinct value is equal to the threshold value that the user sets. So the attribute is not performed in the generalization process. In second attribute Nationality, the distinct value is greater than the attribute's threshold and the attribute has higher level concept, the substitution of the value by its higher level concept generalizes the attribute. A distinct value of the third attribute Religion is greater than the threshold's value. But there is no higher level concept provided for this attribute. Thus, the Religion attribute should be removed in generalization. The distinct value of fourth attribute Gender is equal to the attribute's threshold and so the generalization process is not performed for this Gender attribute.

Table 3. Simplification and Count Propagation using the attribute generalization threshold control

Name	Nationality	Gender	Count
Lin Lin Oo	Native	M	1
Khin Zaw	Native	M	1
Nwe Ni Win	Native	F	1
Hla Aye	Native	F	1
Mie Mie	Native	F	1
Tin Tun	Native	M	1
Aye Aye	Native	F	1
Phyu Phyu Nwe	Native	F	1
Khin Khin Phyu	Native	F	1

After performing attributes generalization threshold process, the system will result in groups of identical tuples. Such identical tuples are then merged into one, with their counts accumulated. Table 3 shows the generalized relation with count propagation that applies attribute generalization threshold control approach.

And then the system performs the second approach, called generalized relation threshold control. The user can set generalized relation threshold value to further reduce the size of the generalized relation.

In this process, if the generalized relation threshold value sets to 5, this system performs attribute removal process for each attribute because there is no higher level on each attribute. If the Name attribute is selected for attribute removal process, merge identical values for other attributes - Nationality and Gender.

Table 4. Final generalized relation using the generalized relation threshold control

Nationality	Gender	Count
Native	M	3
Native	F	6

Table 4 shows the results of the final generalized relation. The user can adjust the generalization thresholds in order to obtain interesting concepts.

6. Advantages of the System

This system facilitates in examining the general behavior of the data from importing databases. This system reduces the computational complexity of database learning process using AOI-based Data Generalization method. This system makes large amounts of data to be more meaningful and easier to interpret. This system provides decision making process. It allows the users to view the data at more meaningful abstractions compared to the primitive level. This system is more efficient than mining for larger and ungeneralized data set using AOI.

7. Conclusion

Attribute-Oriented Generalization in data mining performs grouping of tuples based on the similarity of the attribute values. Domain knowledge in the form of concept hierarchies helps to generalize the concept of the attributes in the database relations. Although detailed information is lost using AOI-based Data Generalization, this system presents interesting general properties of the data.

Another approach of data generalization, Data Cube or OLAP (Online Analytical Processing), confines dimensions to nonnumeric data and measures to numeric data. In reality, the database can include attributes of various data types, including numeric, nonnumeric, spatial, text, or image. Moreover, OLAP is a purely user-controlled process. Although the control in most OLAP systems is quite user-friendly, users do require a good understanding of the role each dimension. Furthermore, in order to find a satisfactory description of the data, users may need to specify a long sequence of OLAP operations [9].

AOI works for complex types of data and relies on a data-driven generalization process. AOI primarily adopts relational operations, such as selection, join, projection (extracting task-relevant data and removing attributes), tuple substitution (ascending concept trees). Such relational operations are set-oriented and have typically been optimized in most existing database systems. So AOI is not only efficient but can also easily be exported to other relational systems [9]. This system is applicable for all strategic level users of any domain (e.g. Immigration information system, Educational servicing system) in decision making process effectively. So this system is robust and extensible use to other databases using AOI based data generalization.

8. References

[1] Y.Y.Wu, Y.L.Chen, R.I.Chang, “Generalized Knowledge Discovery from Relational Databases”

[2] R.A.Angryk, F.E.Perty, “Knowledge discovery in Fuzzy database using Attribute-Oriented Induction”

[3] J.Han, Y.Fu, “Dynamic Generation and Refinement of Concept Hierarchies for Knowledge Discovery in Database”

[4] J.Han, Y.Cai, N.Cercone, “Knowledge Discovery in Databases: An Attribute-Oriented Approach”

[5] H.Y.Hwang, W.C.Fu, “Efficient Algorithms for Attribute-Oriented Induction”

[6] L.Deqin, M.Weijun, “Data Mining of Population Distribution Rules: an Attribute-Oriented Approach”

[7] Spits Warnars H.L.H, “Attribute oriented induction with star schema”

[8] Jinshi Xia, “Attribute-Oriented Induction in Object-Oriented Databases”

[9] J.Han, M.Kamber, “Data Mining Concepts and Techniques”, “Second Edition”, ISBN: 978-1-55860-901-3.