

High Frequency Patterns in Data Mining

Tsau Young Lin

San Jose State University, USA

INTRODUCTION

The principal focus is to examine the foundation of association (rule) mining (AM) via granular computing (GrC). The main results is: The set of all high frequency patterns can be found by sloving linear inequalities within a polynomial time.

BACKGROUND

Some Foundation Issues in Data Mining

What is data mining? The following informal paraphrase of Fayad et al. (1996)'s definition seems quite universal:

Deriving useful patterns from data.

The keys are data, patterns, derivation system, and useful-ness. We will examine critically the current practices of AM.

Some Basic Terms in Association Mining (AM)

In AM, two measures, support and confidence, are the main criteria. It is well known among researchers the support is the main hurdle, in other words, high frequency patterns are the main focus. AM is originated from the market basket data (Agrawal, 1993). However, we will be interested in AM for relational tables. For definitive, we assert:

1. A relational table is a bag relation, that is, repetitions of tuples are permissible (Garcia-Monila et al. 2002)
2. An item is an attribute value,
3. A q-itemset is a subtuple of length q,
4. A high frequency pattern of length q is a q-subtuple if its number of occurrences is greater than or equal to a given threshold.

Emerging Data Mining Method - Granular Computing

Bitmap index is a common notion in database theory. The advantage of bitmap representation is computationally efficient (Louis & Lin, 2000), and the drawback is the order of the table has to be fixed (Garcia-Molina, 2002). Based on granular computing, we propose a new method, called granular representations, that avoids this drawback. We will illustrate the idea by examples. The following example is modified from the text cited above (p. 702). A relational table K is viewed as a knowledge representation of a set V, called the universe, of real world entities by tuples of data; see Table 1.

A bitmap index for an attribute is a collection of bit-vectors, one for each possible value that may appear in the attribute. For the first attribute, *BusinesSize* (the amount of business in millions), the bitmap index would have nine bit-vectors. The first bit-vector, for value TWENTY, is 100011100, because the first, fifth, sixth, and seventh tuple have *BusinesSize* = TWENTY. The other two, for values TEN and THIRTY, are 011100000 and 000000011 respectively; Table 1 shows both the original

Table 1. K and B are isomorphic

V		BusinesSize	Bmonth	City	BusinesSize	Bmonth	City
v ₁		TWENTY	MAR	NY	100011100	110011000	101000000
v ₂		TEN	MAR	SJ	011100000	110011000	010011100
v ₃		TEN	FEB	NY	011100000	001100000	101000000
v ₄	K	TEN	FEB	LA	011100000	001100000	000100011
v ₅	→	TWENTY	MAR	SJ	100011100	110011000	010011100
v ₆		TWENTY	MAR	SJ	100011100	110011000	010011100
v ₇		TWENTY	APR	SJ	100011100	000000100	010011100
v ₈		THIRTY	JAN	LA	000000011	000000011	000100011
v ₉		THIRTY	JAN	LA	000000011	000000011	000100011
Relational Table K					Bitmap Table B		

Table 2a. Granular data model (GDM) for *BusinesSize* attribute

BusinesSize	Granular Representation	Bitmap Representation
TWENTY	= {v ₁ , v ₅ , v ₆ , v ₇ }	=100011100
TEN	= {v ₂ , v ₃ , v ₄ }	=011100000
THIRTY	= {v ₈ , v ₉ }	=000000011
	GDM in Granules	GDM in Bitmaps

Table 2b. Granular data model (GDM) for *Bmonth* attribute

Bmonth	Granular Representation	Bitmap Representation
Jan	= {v ₈ , v ₉ }	=000000011
Feb	= {v ₃ , v ₄ }	=001100000
Mar	= {v ₁ , v ₂ , v ₅ , v ₆ }	=110011000
APR	= {v ₇ }	=000000100
	GDM in Granules	GDM in Bitmaps

Table 2c. Granular data model (GDM) for *CITY* attribute

City	Granular Representation	Bitmap Representation
LA	= {v ₄ , v ₈ , v ₉ }	=000100011
NY	= {v ₁ , v ₃ }	= {v ₁ , v ₃ }
SJ	= {v ₂ , v ₅ , v ₆ , v ₇ }	=010011100
	GDM in Granules	GDM in Bitmaps

Table 2. *K* and *G* are isomorphic

V		BusinesSize	Bmonth	City	BusinesSize	Bmonth	City
v ₁		TWENTY	MAR	NY	{v ₁ ,v ₅ ,v ₆ ,v ₇ }	{v ₁ ,v ₂ ,v ₅ ,v ₆ }	{v ₁ ,v ₃ }
v ₂		TEN	MAR	SJ	{v ₂ ,v ₃ ,v ₄ }	{v ₁ ,v ₂ ,v ₅ ,v ₆ }	{v ₂ ,v ₅ ,v ₆ ,v ₇ }
v ₃		TEN	FEB	NY	{v ₂ ,v ₃ ,v ₄ }	{v ₃ ,v ₄ }	{v ₁ ,v ₃ }
v ₄	K	TEN	FEB	LA	{v ₂ ,v ₃ ,v ₄ }	{v ₃ ,v ₄ }	{v ₄ ,v ₈ ,v ₉ }
v ₅	→	TWENTY	MAR	SJ	{v ₁ ,v ₅ ,v ₆ ,v ₇ }	{v ₁ ,v ₂ ,v ₅ ,v ₆ }	{v ₂ ,v ₅ ,v ₆ ,v ₇ }
v ₆		TWENTY	MAR	SJ	{v ₁ ,v ₅ ,v ₆ ,v ₇ }	{v ₁ ,v ₂ ,v ₅ ,v ₆ }	{v ₂ ,v ₅ ,v ₆ ,v ₇ }
v ₇		TWENTY	APR	SJ	{v ₁ ,v ₅ ,v ₆ ,v ₇ }	{v ₇ }	{v ₂ ,v ₅ ,v ₆ ,v ₇ }
v ₈		THIRTY	JAN	LA	{v ₈ ,v ₉ }	{v ₈ ,v ₉ }	{v ₄ ,v ₈ ,v ₉ }
v ₉		THIRTY	JAN	LA	{v ₈ ,v ₉ }	{v ₈ ,v ₉ }	{v ₄ ,v ₈ ,v ₉ }
Bag Relation K					GRANULR TABLE G		

table and bitmap table. Bmonth means Birth month; City means the location of the entities.

Next, we will interpret the bit-vectors in terms of set theory. A bit-vector can be viewed as a representation of a subset of V. For example, the bit-vector, 100011100, of *BusinesSize* = TWENTY says that the first, fifth, sixth, and seventh entities have been selected, in other words, the bit-vector represents the subset {v₁, v₅, v₆, v₇}. The other two bi-vectors, for values TEN and THIRTY, represent the subsets {v₂, v₃, v₄} and {v₈, v₉} respectively. We summarize such translations in Table 2a,b,c. and refer to these subsets as elementary granules.

Some easy observations:

1. The collection of elementary granules of an attribute (column) forms a partition, that is, all granules of this attribute are pairwise disjoint. This fact was observed by Pawlak (1982) and Tony Lee (1983).
2. From Tables 1 and 2, one can easily conclude that the relational table K, the bitmap table B and granular table G are isomorphic. Two tables are isomorphic if one can transform a table to the other by renaming all attribute values in a one-to-one fashion.

4 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/high-frequency-patterns-data-mining/10660

Related Content

Ontology Query Languages for Ontology-Based Databases: A Survey

Stéphane Jean, Yamine Aït Ameur and Guy Pierra (2010). *Data Warehousing Design and Advanced Engineering Applications: Methods for Complex Construction* (pp. 227-247).

www.irma-international.org/chapter/ontology-query-languages-ontology-based/36617

Data Mining in Franchise Organizations

Ye-Sho Chen, Robert Justis and P. Pete Chong (2008). *Data Warehousing and Mining: Concepts, Methodologies, Tools, and Applications* (pp. 2722-2733).

www.irma-international.org/chapter/data-mining-franchise-organizations/7795

On Querying Data and Metadata in Multiversion Data Warehouse

Wojciech Leja, Robert Wrembel and Robert Ziembicki (2010). *Data Warehousing Design and Advanced Engineering Applications: Methods for Complex Construction* (pp. 206-226).

www.irma-international.org/chapter/querying-data-metadata-multiversion-data/36616

Big Data, Semantics, and Policy-Making: How Can Data Dynamics Lead to Wiser Governance?

Lamyaa El Bassiti (2019). *Big Data Governance and Perspectives in Knowledge Management* (pp. 154-178).

www.irma-international.org/chapter/big-data-semantics-and-policy-making/216807

Data Management in Three-Dimensional Structures

Xiong Wang (2005). *Encyclopedia of Data Warehousing and Mining* (pp. 228-232).

www.irma-international.org/chapter/data-management-three-dimensional-structures/10598