Multi-View Actionable Patterns for Managing Traffic Bottleneck

Xiaodong Yue
School of Computer Eng & Sci
Shanghai University
Shanghai, 200444, China

Longbing Cao
Advanced Analytics Institute
University of Technology, Sydney
NSW, 2007, Australia

Yufei Chen
School of Electronics & Info
Tongji University
Shanghai, 201804, China

Bin Xu
School of Transportation Eng
Tongji University
Shanghai, 201804, China

Abstract

Discovering congestion patterns from table-formed traffic reports is critical for traffic bottleneck analysis. However, patterns mined by existing algorithms often do not satisfy user requirements and are not actionable for traffic management. Traffic officers may not pursue the most frequent patterns but expect mining outcomes showing the dependence between congestion and various kinds of road properties for traffic planning. Such multi-view analysis requires to integrate user preferences of data attributes into pattern mining process. To tackle this problem, we propose a multi-view attributes reduction model for discovering the patterns of user interests, in which user views are interpreted as preferred attributes and formulated by attribute orders. Based on the pattern discovery model, a workflow is built for traffic bottleneck analysis, which consists of data preprocessing, preference representation and congestion pattern mining. Our approach is validated on the reports of road conditions from Shanghai, which shows that the resultant multi-view findings are effective for analyzing congestion causes and traffic management.

Introduction

Although *knowledge discovery and data mining* (KDD) has become an active research area in information technology fields, there is still a big gap between the research deliverables and real business needs (Fayyad, Piatetsky-Shapiro, and Uthurusamy 2003). Often the data-centric patterns mined by many existing KDD algorithms cannot satisfy user expectations and application environments (Cao et al. 2009). To bridge this gap, traditional data mining objective was transferred to *actional knowledge discovery* (AKD), AKD aims to find the actionable patterns which are friendly enough for business people to interpret, validate and action (Wang, Jiang, and Tuzhilin 2006; Yang et al. 2007). Different from the traditional patterns of statistical significance, the actionable patterns involve subjective interestingness measures (Chan, Yang, and Shen 2003; Wong, Fu, and Wang 2003) and coincide with particular user preferences (Liu et al. 2000; Tan, Kumar, and Srivastava 2002; Omiecinski 2003). In reality, user preferences of actionable patterns are essentially related to multiple views of data analysis, in which each view is formally represented by a group of data attributes (Lin, Yao, and Zadeh 2002; Chen and Yao 2006). Therefore, AKD methodologies should have the ability to analyze data from multiple views, i.e. the ability to reduce and organize data attributes according to user preferences to generate actionable patterns.

In the field of *intelligent transportation* (Ma, Zheng, and Wolfson 2013; Yuan, Zheng, and Xie 2012), it is expected to analyze traffic bottlenecks through discovering congestion patterns from traffic reports (Zhang et al. 2011). Most existing works directly apply the techniques of association rule mining on reports to obtain congestion patterns (Barai 2003; Wong and Chung 2007). However, without considering user preferences and requirements, few patterns of only statistical significance are actionable enough to support traffic management. The congestion patterns obtained by traditional mining methods generally consist of the attributes about road construction, such as road width, length and average delay etc. But changing these factors are always difficult because reconstructing roads in modern cities are really high-cost tasks. Thus these patterns are not applicable for traffic improvement.

For outcomes from actionable congestion patterns, traffic management officers expect to study the causes of traffic bottlenecks from different perspectives. First, they hold the views to facilitate traffic planning. The discovered patterns should indicate the relationship between congestion and the road properties that are easy to be improved, such as car/bike separators, bus lanes and region environment. Second, officers hold the views to involve expert experiences into bottleneck analysis. The road properties being considered important for congestion formation should have high priority to emerge in the final patterns. The experts' experiences are helpful to discover, employ and interpret the actionable patterns. As mentioned above, each view for traffic bottleneck analysis actually denotes a kind of preference of road prop-

Copyright © 2015, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

erties. To make the patterns actionable, it is required to integrate user-defined attribute preferences into pattern mining process.

Aiming at the problems, we propose a multi-view pattern mining model for traffic bottleneck analysis. Our contributions are summarized as follows.

- *Propose a multi-view attributes reduction model to produce patterns of user interests.* In the model, user views are expressed with attribute preferences and formulated by attribute orders. To implement the order-based attributes reduction, we design a data structure of 2D linked list to store item-discerning elements and set discernibility thresholds to filter out trivial attributes in patterns.
- *Propose a workflow for traffic bottleneck analysis based on the multi-view reduction model.* The workflow consists of data collection & preprocessing of road conditions, user preference representation and attributes reduction for congestion pattern mining. It supports users to analyze the causes of urban traffic bottlenecks from the views of traffic management.

Multi-view Pattern Mining

User View Representation

The views of user in data analysis actually represent the users' preferences of data attributes. Attributes preferences express various user interests and lead to the multi-view analysis results. The user bias of data attributes can be formally represented in the form of strict order of attributes, i.e. attribute priority.

Definition 1. Attribute Order. *Given an information system $IS = (U, C, V, f)$, in which U is a finite set of data items, V denotes the domain of attribute values, $C = \{a_1, a_2, ..., a_m\}$ is the attribute set and $m = |C|$. A strict attribute order S on set C is defined as,*

$$S : a_1 \prec a_2 \prec ... \prec a_m \tag{1}$$

The order S is an attribute sequence which defines the priority of attributes, such as $a_1 \prec a_2$ denotes the attribute a_1 is prior to a_2.

Based on the attribute order, we can further define the order of attribute sets. For an attribute set, the set priority is decided by the priority of its member attributes.

Definition 2. Token Attribute. *Given an information system $IS = (U, C, V, f)$, S is an attribute order on set C. For a subset of attributes $A \subset C$, a token attribute is the first element of S contained in A. Obviously, the token attribute owns the highest priority in attribute set A.*

Definition 3. Attribute Set Order. *Given two attribute subsets $A, B \subset C$, sort the attributes of A and B with order S and obtain $A^S = \{a_i, ...\}$ and $B^S = \{a_j, ...\}$, a_i, a_j are the token attributes of A and B respectively. If $a_i \neq a_j$ and $a_i \prec a_j$, the sequence $A^S \prec B^S$ and thus the attribute set A is prior to B ($A \prec B$); if $a_i = a_j$, A and B have the same priority, i.e. $A \cong B$.*

Attributes Reduction for Pattern Mining

Based on the definitions of attribute preference, the user interests can be involved into pattern mining process through attributes reduction. Attributes reduction aims at finding a minimum set of conditional attributes that preserves a certain classification property, i.e. *attribute reduct* (Pawlak 1991). The attribute reducts with preferences can induce the patterns from different user views. The reduction algorithm is built on the basis of discerning elements. Each discerning element is a set of attributes to distinguish a pair of items belonging to different classes.

Data Structure Considering spatial complexity, we use a 2-dimensional list to storage the discerning elements. The first-dimension list represents M conditional attributes of system. On the second dimension, each attribute a_i has a linked list of discerning elements, in which all the elements have the same token attribute a_i. To further simplify the 2-dimensional list, we use the following strategies to remove redundant discerning elements.

- Because the comparison of two items is symmetric, i.e. the discerning element $e(i, j) = e(j, i)$, for each item pair, just one copy of their different attributes is stored in the list.
- Reduce redundant discerning elements based on Absorption Law in Set Theory. For two elements e and $e^{'}$, if $e \subseteq e^{'}$, $e^{'}$ can be replaced by e in the list.

Moreover, before being inserted into the linked list, the attributes in every discerning element are sorted by the given order. This strategy facilitates the computation of token attributes and the comparison of attributes in further reduction process. To balance computational time and spatial complexity, Absorption Law is applied to only the elements having the same token attribute rather than the total discerning elements to remove redundancy.

Reduction Algorithm The attributes reduction algorithm involving attribute preferences is constructed based on the partition of discerning elements. Given an attribute order $a_1 \prec ... \prec a_m$ for a decision system, referring to Definition 3, the binary relation of equal priority '$\cong$' between two attribute sets depends on their token attributes. Obviously, the relation '$\cong$' is reflexive, symmetric and transitive, thus can partition all the discerning elements (attribute subsets) M into m disjoint equivalence classes, $M/\cong$: $\{[a_1], ..., [a_m]\}$. Each equivalence class $[a_i]$ is denoted by the common token attribute and all the elements of it have the same priority. Thus we also have $[a_1] \prec ... \prec [a_m]$. Through partitioning the discerning elements, item discernibility is graded to different levels according to the predefined attribute preference. Recalling the data structure, the second-dimension lists represent the partition of discerning elements. Based on this, the reduction algorithm with attribute preference is designed as follows.

As shown in Algorithm 1, in reduction process, attributes are selected according to their priority and discernibility. For the attribute with high priority, it takes precedence to distinguish item pairs and thus is more likely to occur in the

Algorithm 1 Reduction with Attribute Preference (RAP)

Input:
Decision system $DS = (U, C \bigcup D, V, f)$, $|C| = m$,
Attribute preference, $S : a_1 \prec a_2 \prec ... \prec a_m$
Output: Attribute reduct R based on the preference S

```
Initialize reduct R = ∅ ;
Construct the 2-dimensional list M to partition all the
  discerning elements. m linked lists represent the equiva-
  lence classes of m token attributes, {[a_1], ..., [a_m]}, and
  class priority [a_1] ≺ ... ≺ [a_m];
while M ≠ ∅ do
  Choose the class one by one from low to high priority;
  for i = m to 1 do
    Check the irreplaceable discernibility of a_i;
    if |[a_i]| < T_id then
      Delete all the elements of [a_i] from M;
    else
      Browse M forward and count the number of ele-
        ments which contain the attribute a_i, the number
        is denoted by K_{a_i};
      Check the general discernibility of a_i;
      if K_{a_i} < T_d then
        Delete all the elements of [a_i] from M;
      else
        Add attribute a_i to reduct R and delete all the
          elements containing a_i from M;
      end if
    end if
  end for
end while
Output attribute reduct R.
```

reduct. On the other hand, for the attribute with low priority, if it is the token attribute of many discerning elements, this indicates the attribute is irreplaceable and should be added to the reduct. Furthermore, the discernibility of an attribute consists of the general and irreplaceable discerning ability. For an attribute a_i, the number of the elements in class $[a_i]$ denotes its irreplaceable discernibility, and the number of all the discerning elements containing a_i represents its general discernibility. To filter out the trivial attributes, we adopt two thresholds of discernibility to guarantee reducts are concise. The thresholds T_d and T_{id} are respectively used to evaluate the general and irreplaceable discernibility of an attribute. In algorithm implementation, we set $T_d = \lceil T/m \rceil$ and $T_{id} = \lceil 0.02T \rceil$, in which m is the number of attributes and T is the number of all the discerning elements.

Through attributes reduction process, the original conditional attributes are reduced to attribute reduct for classification. Integrating attribute values into the reduct, we can obtain the corresponding data patterns. With different attribute preferences, RAP generates different attribute reducts and thus leads to the patterns of various kinds of user interests.

Model Analysis

Involving the attributes of user interests may lead to the attribute reducts including more attributes than the traditional patterns. Thus it is required to analyze the redundancy of the attribute reducts with user preferences. Next we prove that the attribute reducts obtained by the proposed reduction model are still *independent*, i.e. all the attributes in a reduct are necessary for classification (Pawlak 1991). This means the reducts contain no redundant attributes and guarantee the high-quality patterns.

Theorem 1. *Given a decision system $DS = (U, C \cup D, V, f)$, $|C| = m$ and an attribute preference $S : a_1 \prec a_2 \prec ... \prec a_m$, suppose the partition of discerning elements induced by S is $\{[a_1], ..., [a_m]\}$. For any attributes $a_p, a_q \in C$, if the attribute priority $a_p \prec a_q$, then $\forall e \in [a_q]$, $e \cap \{a_p\} = \emptyset$.*

Proof. For any element $\forall e \in [a_q]$, e is a subset of attributes to discern an item pair and its token attribute is a_q. According to Definition 2, any attribute $a \in e$ cannot be prior to the attribute a_q, $a_q \preceq a$. Because $a_p \prec a_q$, we have $a_p \prec a_q \preceq a$, and $a_p \prec a$, the attribute a_p is prior to any attribute in e. Thus for any element e in class $[a_q]$, does not contain the attribute a_p, i.e. $e \cap \{a_p\} = \emptyset$. □

As shown in Theorem 1, for a class of discerning elements, all its members cannot contain the attributes which are prior to its token attribute. This indicates that if the discerning class $[a]$ is not empty, the discernibility of attribute a cannot be replaced by the attributes prior to it. Generally, the more elements $[a]$ has, the more important attribute a is for discerning item pairs. Even owning low priority, attribute a should be selected into reduct due to its irreplaceable discernibility. Based on Theorem 1, we can further check the dependency of the reduct attributes.

Theorem 2. *Given a decision system $DS = (U, C \cup D, V, f)$, $|C| = m$ and an attribute preference $S : a_1 \prec a_2 \prec ... \prec a_m$, the attribute reduct obtained by algorithm RAP is independent.*

Proof. Suppose R is a reduct obtained by algorithm RAP, R consists of k attributes, $R : \{r_1, r_2, ..., r_k\}$, $r_i \in C$, $1 \leq i \leq k$, and the attribute priority $a_1 \preceq r_1 \prec r_2 \prec ... \prec r_k \preceq a_m$. In order to prove a reduct R is independent, referring to the definition of attribute reduct (Pawlak 1991), we should demonstrate that every attribute in reduct R is indispensable.

First we demonstrate attribute r_k, which has the lowest priority in R, is indispensable in reduct R. Since RAP selects attributes from low to high priority, r_k is the first selected attribute and $r_k = \min \text{priority}\{a_i | [a_i] \neq \emptyset\}$. $\forall r \in R - \{r_k\}$, $r \prec r_k$, according to Theorem 1, we have $\forall e \in [r_k]$, $e \cap \{r\} = \emptyset$. This means the item pairs discerned by r_k cannot be distinguished by the other attributes in reduct, thus r_k has irreplaceable discernibility and is indispensable in R.

Next we demonstrate the other attributes in R indispensable. $\forall r \in R - \{r_k\}$, suppose M^r be the set of discerning elements before selecting attribute r, referring to RAP algorithm, all the elements containing the reduct attributes of the priority lower than r have been removed from M^r, i.e. $\forall r' : r' \in R \wedge r \prec r'$, $\forall e \in M^r$, $e \cap \{r'\} = \emptyset$. Since $[r] \subset M^r$, we have $\forall e \in [r]$,$e \cap \{r'\} = \emptyset$. For the reduct attributes prior to r, $\forall r^* : r^* \in R \wedge r^* \prec r$, according to

Theorem 1, $\forall e \in [r], e \cap \{r^*\} = \emptyset$. As mentioned above, for the reduct attributes whether prior or posterior to r, r has the irreplaceable discernibility. Thus r is indispensable in R.

To sum up, all the reduct attributes $\{r_1, r_2, ..., r_k\}$ are indispensable in R. Thus the reduct obtained by RAP is independent. □

From Theorem 1 and 2, we know that adding the attribute preference as prior information to attributes reduction model is helpful to not only discover patterns from user views but also guarantee the patterns without redundancy.

Workflow of Traffic Bottleneck Analysis

Data Preparation

The target of traffic bottleneck analysis is to study the causes of road congestion. Urban traffic congestion is not only related to road construction, but also city planning and traffic management. In this paper, we adopt 14 attributes as candidate factors for fully analyzing urban traffic congestion, in which some factors are inherent road properties about construction and traffic capacity and the others are related to road programming and region environment. Using attributes reduction model to discover data patterns of traffic congestion, it is necessary to discretize attribute values to obtain a symbolic and semantic description of road properties. The criteria of discretization are made through referring to Urban Road Traffic Performance Index and expert experiences. Table 1 lists the description and discretization criteria of all attributes. Besides the conditional attributes of road properties, the decision attribute value to judge whether a road is a traffic bottleneck is assigned by domain experts.

User Preferences

The user preferences of road properties for traffic bottleneck analysis come from either application requirements or domain knowledge. As introduced above, the user preferences of road properties are formulated by attribute orders. Thus we can model the user views of traffic bottleneck analysis through setting the priority of road properties. To facilitate this setting, we divide 14 road properties into 5 groups from the aspects of road construction, infrastructure setup, traffic environment, traffic capacity and bus station. To obtain a total order of properties, users can first rank property groups and then adjust order within each group. This strategy is implemented through a friendly user interface.

For example, if traffic management personnel hold a view to avoid traffic congestion through optimizing road infrastructure setup and traffic environment, the following road property order can be used to express user preference.

- Preference for *infrastructure setup* and *environment*: {Car/bike separator $\prec$ Middle separator $\prec$ Zebra strips $\prec$ Exit/entrance } $\prec$ {Region $\prec$ Land type $\prec$ Traffic volume} $\prec$ {Bus lane $\prec$ Bus station} $\prec$ {Joint roads number $\prec$ Capacity difference} $\prec$ {Lane number $\prec$ Length $\prec$ Average delay}

Extract Patterns of Traffic Bottleneck

Given a user preference of road properties, we can use attributes reduction model to extract patterns of traffic bottlenecks from the corresponding view. For a decision system of traffic congestion, first RAP algorithm is used to obtain an attribute reduct of road properties with user preference. Next candidate patterns of traffic congestion are generated through integrating attribute values into the reduct. Finally, based on pattern evaluation criteria, the redundant patterns in candidate ones are filtered out. The workflow of discovering traffic bottleneck patterns is shown below.

Algorithm 2 Extracting Traffic Bottleneck Patterns

1: Preprocess the data of road conditions;
2: Formulate user preferences with the priority orders of road properties;
3: For a priority order, utilize RAP algorithm to compute an attribute reduct of road properties;
4: For every item of class 'congestion', integrate its property values into attribute reduct to form a candidate pattern;
5: Remove redundant patterns and filter out the candidate patterns according to their Confidences and Lifts.

Example 1. *Suppose users have a preference of infrastructure setup and region environment, the attribute reduct obtained by RAP algorithm consists of the following road properties: Traffic volume, Land type, Region, Zebra strips and Middle separator. For the items of class 'congestion', integrate property values to produce candidate patterns as*

1) *Traffic volume = 1 $\wedge$ Land type = 2 $\wedge$ Region = 2 $\wedge$ Zebra strips = 0 $\wedge$ Middle separator = 2 [Conf: 0.714, Lift: 2.187]*
2) *Traffic volume = 1 $\wedge$ Land type = 4 $\wedge$ Region = 3 $\wedge$ Zebra strips = 0 $\wedge$ Middle separator = 1 [Conf: 1, Lift: 3.061]*

.....

After filtering out the candidate patterns by Confidence and Lift, finally we obtain the patterns of traffic congestion as follows.

1) *Traffic volume = 1 $\wedge$ Land type = 4 $\wedge$ Region = 3 $\wedge$ Zebra strips = 0 $\wedge$ Middle separator = 1 [Conf: 1, Lift: 3.061]*

.....

The patterns above indicate the coupling effects of separator setting and region environments on traffic congestion. Users can study the dependence between congestion and other road properties through reformulating their preferences.

Experimental results

For experiments, we build up a decision system of the road conditions in urban areas of Shanghai. Each record in the decision system has 14 conditional attributes (see Table 1) to present the properties of an urban road and 1 decision attribute to judge congestion. We select 300 representative road sections from thousands of ones to form the experimental data. These road sections cover 9 districts in the urban area of Shanghai, which include Huangpu, Jing'an, Changning, Hongkou, etc. 17% of these road sections locate within the inner-ring city area and 83% locate around the middle-ring. The region types of these road sections involve Entertainment (2.7%), Residential(64.3%), Business(26%)

Table 1: Description of Road Properties

Length	Value	*Middle separator*	Value	*Car/bike separator*	Value
0-200	1	No separator	0	No separator	0
200-500	2	Divider	1	Divider	1
500-800	3	Pier		Pier	
>800	4	Strip	2	Strip	2
				No vehicle lane	3
Zebra strips	Value	*Traffic volume*	Value	*Region*	Value
No	0	0-18.75	1	Outer middle-ring	1
Yes	1	18.75-37.5	2	Around middle-ring	2
		>37.5	3	Within inner-ring	3
Lane number	Value	*Exit/entrance*	Value	*Bus lane*	Value
1	1	0 or 1	1	No	0
2	2	2	2	Yes	1
>2	3	>2	3		
Bus station	Value	*Land type*	Value	*Capacity difference*	Value
No station	0	Entertainment	1	0	0
Bus bay	1	Residential	2	0-1300	1
Roadside stop	2	Business	4	1300-5400	2
		Government	8	>5400	3
Joint roads number	Value	*Average Delay*	Value		
0-5	1	0-0.08	0		
6	2	0.08-0.14	1		
>6	3	>0.14	2		

and Government(7%). The data of road conditions were collected in a period of 12 months thus can reflect the city traffic situation comprehensively.

To validate the capability of our approach for mining actionable traffic patterns, we evaluate patterns from the aspects of *accuracy*, *concision* and *user interests*. The popular measures of Confidence and Lift are used to evaluate pattern accuracy. The pattern concision is measured by the compression ratio of conditional attributes. Finally, the degree-of-interest of a pattern is evaluated by the average of preference scores of all pattern attributes. The preference scores of attributes are determined by priority order. The ith attribute can be simply assigned the score $|C| - i + 1$. The higher priority an attribute has, the higher score it is assigned. We present an overall evaluation of RAP patterns through comparing with the following popular pattern mining algorithms: Apriori (Han, Kamber, and Pei 2011), Predictive Apriori (Scheffer 2001) and Tertius (Flach and Lachiche 1999). Suppose a user holds the preference for road infrastructure setup and environment, Table 2 lists the best 3 congestion patterns generated by different pattern mining methods.

As shown in Table 2, RAP can extract the precise congestion patterns with high Confidences and Lifts as most existing methods. Focusing on the co-occurrence of items, the patterns mined by the traditional algorithms always consist of the attributes of road construction. From the patterns, we find that the roads of long length and high average delay are more likely to be congested. But in a modern city, it is difficult to change these road conditions directly. Thus these patterns are not actionable to support traffic planning. For the traffic officers who are interested in the roles of separator setting and region environment in traffic congestion, these patterns cannot coincide with his expectation. Through involving attribute preferences into pattern mining, RAP captures the correlation between congestion and the road properties of user interests. RAP patterns indicate the effects of separator setting and region environment on congestion. In the business and residential regions within the middle-ring city area, the deficiency of road separator setting easily leads to traffic block even if the surrounding traffic volume is normal. Meanwhile, in the central city area of high traffic volume, the effective way to ease the traffic congestion is crowd decentralization. Moreover, capturing comprehensive correlation among road properties, RAP patterns generally contain more attributes than frequent patterns. But it should be noticed that no attribute in RAP pattern is redundant, because the attributes reduct obtained by RAP is independent (See Theorem 2).

To further validate the ability of RAP for multi-view actionable pattern mining, we suppose 4 users with different views for traffic bottleneck analysis. These views focus on infrastructure setup, bus station, traffic capacity and road construction respectively. Figure 1 illustrates the average degree-of-interest of the patterns obtained by different pattern mining methods. Obviously, without considering user interests, traditional algorithms generate the same patterns for different users and thus result in low degree-of-interest. Especially for the users being interested in the road properties not occurring frequently in the congestion records, such as bus lane, traditional algorithms over focus on road construction and cannot provide the satisfactory results. We also find that when the user has an attribute preference for road construction, the degree-of-interest of the traditional patterns will be close to RAP patterns. This indicates that adopting the specific attribute preferences, RAP can obtain

Table 2: Evaluation of Traffic Bottleneck Patterns

Methods	Traffic Bottleneck Patterns	Conf	Lift	Conc	Int
Apriori	Length=3 $\wedge$ aveDelay=1 $\rightarrow$ Congestion	1	3.061	14.3%	1.5
	Length=3 $\wedge$ trfVol=2 $\wedge$ landType=2 $\rightarrow$ Congestion	1	3.061	21.4%	6.3
	Length=3 $\wedge$ Region=3 $\wedge$ aveDelay=1 $\rightarrow$ Congestion	1	3.061	21.4%	4.3
Predictive Apriori	Length=3 $\wedge$ aveDelay=1 $\rightarrow$ Congestion	1	3.061	14.3%	1.5
	Length=3 $\wedge$ trfVol=2 $\wedge$ landType=2 $\rightarrow$ Congestion	1	3.061	21.4%	6.3
	Length=3 $\wedge$ CapDif=3 $\rightarrow$ Congestion	1	3.061	14.3%	3
Tertius	Length=3 $\wedge$ aveDelay=1 $\rightarrow$ Congestion	1	3.061	14.3%	1.5
	Length=2 $\wedge$ busLane=0 $\wedge$ aveDelay=2 $\rightarrow$ Congestion	0.810	2.478	21.4%	3.3
	busLane=0 $\wedge$ aveDelay=2 $\rightarrow$ Congestion	0.622	1.904	14.3%	4
RAP	trfVol=1 $\wedge$ landType=4 $\wedge$ Region=3 $\wedge$ zebStrip=0 $\wedge$ midSep=1 $\rightarrow$ Congestion	1	3.061	35.7%	10.4
	trfVol=2 $\wedge$ landType=2 $\wedge$ Region=2 $\wedge$ zebStrip=1 $\wedge$ midSep=0 $\rightarrow$ Congestion	1	3.061	35.7%	10.4
	trfVol=3 $\wedge$ landType=4 $\wedge$ Region=3 $\wedge$ zebStrip=1 $\wedge$ midSep=2 $\rightarrow$ Congestion	1	3.061	35.7%	10.4

the patterns similar to the traditional ones. For the users with different interests of patterns, RAP actually provides a flexible tool for data analysis.

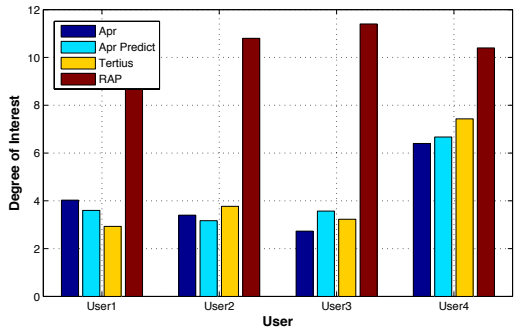

Figure 1: Degree-of-interest with multiple views

We briefly present how to use RAP patterns to analyze the causes of urban traffic bottleneck and further support the traffic programming. Due to the length limit, we just present a part of experimental results here. We also simulate multi-form preferences to study the coupling and contrast effects of road properties on traffic congestion and find some interesting rules. Synthesizing these rules, we can construct the comprehensive knowledge on traffic congestion. For the users holding different views for traffic bottleneck analysis, our approach produce actionable patterns which are consistent with real application requirements.

Related Work

For the report-type traffic data mining, most existing works focus on association rule mining. Barai employed association rules to explore the relationship between road types and traffic accidents (2003). Gong and Liu predicted traffic network flows based on the association rules with association analysis (2003). Wong and Chung employed rough sets to model the mechanism of traffic accidents as chains of attributes, which include driver properties, travel properties and environmental factors (2007). Chang et al. used algorithm of attributes reduction in to mine the rules to induce pavement maintenance and rehabilitation strategy (2007).

As an inductive learning tool, *attributes reduction* targets at reducing original attributes to an attribute reduct while preserving the classification property of system (Pawlak 1991; Lin, Yao, and Zadeh 2002). To involve prior information into attributes reduction, Wang first designed a reduction algorithm based on attribute orders and analyzed its properties. The reduction process is built up on the basis of discernibility matrix (2001). Yao proposed a formal framework to represent the attribute bias by both qualitative and quantitative judgements for machine learning (2008). Zhao also utilized attribute orders to find a particular reduct according to user interests. The quality of the output reduct relies on the free attributes selected in reduction process (2002). Han and Wang further analyzed the relationship between attribute orders and reducts to judge whether similar attribute orders produce the same reduct (2005).

Conclusion

How to make the outcomes from traffic data mining actionable for traffic bottleneck diagnosis and management is a challenging task. The challenge lies in the gap between diversified user requirements and classic data-driven methodologies. This paper proposes attributes reduction model for extracting actionable traffic congestion patterns that present multiple views of business and cater for traffic manager's interests. User views are expressed by attribute preferences, which are formally represented by attribute orders. We validate our approach based on the reports of road conditions from Shanghai. Experimental results show that the proposed approach is effective in analyzing congestion causes from the views of traffic management.

Acknowledgments

This work was supported by National Natural Science Foundation of China (Serial Nos. 61103067, 61103070, 61170155) and The Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University.

References

Barai, S. 2003. Data mining applications in transportation engineering. *Transport* 18(5):216–223.

Cao, L.; Yu, P. S.; Zhang, C.; and Zhao, Y. 2009. *Domain Driven Data Mining*. New York, USA: Springer Science+Business Media.

Chan, R.; Yang, Q.; and Shen, Y. 2003. Mining high utility itemsets. In *Proceedings of the Third IEEE International Conference on Data Mining (ICDM'03)*, 19–26.

Chang, J.; Hung, C.; Tzeng, G.; and Kang, S. 2007. Using rough set theory to induce pavement maintenance and rehabilitation strategy. *RSKT'07, Lecture Notes in Computer Science* 4481:542–549.

Chen, Y., and Yao, Y. Y. 2006. Multiview intelligent data analysis based on granular computing. In *Proceedings of 2006 IEEE International Conference on Granular Computing*, 281–286.

Fayyad, U.; Piatetsky-Shapiro, G.; and Uthurusamy, R. 2003. Summary from the kdd-03 panel: data mining: the next 10 years. *ACM SIGKDD Explorations Newsletter* 5(2):191–196.

Flach, P. A., and Lachiche, N. 1999. Confirmation-guided discovery of first-order rules with tertius. *Machine Learning* 42:61–95.

Gong, X., and Liu, X. 2003. A data-mining-based algorithm for traffic network flow forecasting. In *Proceedings of 2003 IEEE Intelligent Transportation Systems*, 193–198.

Han, S., and Wang, J. 2005. The second attribute. *RSFDGrc'05, Lecture Notes in Computer Science* 3641:156–165.

Han, J.; Kamber, M.; and Pei, J. 2011. *Data Mining: Concepts and Techniques, 3rd edition*. San Francisco, CA, USA: Morgan Kaufmann Publishers.

Lin, T. Y.; Yao, Y. Y.; and Zadeh, L. A. 2002. *Data mining, rough sets and granular computing*. Heidelberg, Germany: Physica-Verlag.

Liu, B.; Hsu, W.; Chen, S.; and Ma, Y. 2000. Analyzing subjective interestingness of association rules. *IEEE Intelligent Systems* 15(5):47–55.

Ma, S.; Zheng, Y.; and Wolfson, O. 2013. T-share: A large scale dynamic taxi ridesharing service. In *Proceedings of IEEE International Conference on Data Engineering (ICDE'13)*, 410–421.

Omiecinski, E. 2003. Alternative interest measures for mining associations in databases. *IEEE Trans. on Knowledge and Data Engineering* 15(1):57–69.

Pawlak, Z. 1991. *Rough sets: Theoretical Aspects of Reasoning About Data*. Boston, USA: Kluwer Academic Publishers.

Scheffer, T. 2001. Finding association rules that trade support optimally against confidence. In *Proceedings of 5th European Conf. on Principles of Data Mining and Knowledge Discovery*, 424–435.

Tan, P.; Kumar, V.; and Srivastava, J. 2002. Selecting the right interestingness measure for association patterns. In *Proceedings of the eighth ACM SIGKDD international conference on knowledge discovery and data mining (KDD'02)*, 32–41.

Wang, J., and Wang, J. 2001. Reduction algorithms based on discernibility matrix: The ordered attributes method. *Journal of Computer Science and Technology* 16(6):489–504.

Wang, K.; Jiang, Y.; and Tuzhilin, A. 2006. Mining actionable patterns by role models. In *Proceedings of the 22nd International Conference on Data Engineering (ICDE'06)*, 16.

Wong, J., and Chung, Y. 2007. Rough set approach for accident chains exploration. *Accident Analysis and Prevention* 39(3):629–637.

Wong, R.; Fu, A.; and Wang, K. 2003. Mpis: Maximal-profit item selection with cross-selling considerations. In *Proceedings of the Third IEEE International Conference on Data Mining (ICDM'03)*, 371–378.

Yang, Q.; Yin, J.; Ling, C.; and Pan, R. 2007. Extracting actionable knowledge from decision trees. *IEEE Transactions on Knowledge and Data Engineering* 19(1):43–56.

Yao, Y. Y.; Zhao, Y.; Wang, J.; and Han, S. 2008. A model of user-oriented reduct construction for machine learning. *Transactions on Rough Sets VIII* 5084:332–351.

Yuan, J.; Zheng, Y.; and Xie, X. 2012. Discovering regions of different functions in a city using human mobility and pois. In *Proceedings of 18th SIGKDD conference on Knowledge Discovery and Data Mining (KDD'12)*, 186–194.

Zhang, J.; Wang, F.; Wang, K.; Lin, W.; Xu, X.; and Chen, C. 2011. Data-driven intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems* 12(4):1624–1639.

Zhao, K., and Wang, J. 2002. A reduction algorithm meeting users' requirements. *Journal of Computer Science and Technology* 17(5):578–593.