COMPLEX EVENT DETECTION AND PREDICTION IN TRAFFIC

Blaž Kažič, Dunja Mladenić, Luka Bradeško
Artificial Intelligence Laboratory,
Jožef Stefan Institute and Jožef Stefan International Postgraduate School,
Jamova cesta 39, 1000 Ljubljana, Slovenia
e-mail: blaz.kazic@ijs.si

ABSTRACT

When dealing with large amounts of heterogeneous traffic data streams, how a Complex Event Processing (CEP) system, which can efficiently process and predict complex events in traffic, is set up is a crucial matter. In this paper, several issues and methods related to finding different rules that can be used to develop such a system are presented.

Statistical methods to detect complex events from traffic data are first described. Two types of techniques are used to research relations between complex events: descriptive and predictive data mining. First, association rules are used to analyze data and express regularities in data. Second, decision trees and decision rules algorithms are used for the prediction of complex events.

All the algorithms were tested with regards to how different social events affect the traffic system near the Stozice stadium in Ljubljana, Slovenia. The results show that methods described in this paper are feasible and can be used for developing an advanced traveler information system.

1 INTRODUCTION

This work is inspired by the need for the development of methods for real time complex event detection and processing in urban mobility networks, as proposed is the Mobis [1] project. This is usually done with specialized programs for complex event processing (CEP), such as ESPER [2], ETALIS [3], VANET [4]. These programs are capable of receiving different on-line data streams from which they detect changes in real time. These changes can be specified as events and complex events, and can be used for event processing and stream reasoning in real time. However, all these programs require some background knowledge (e.g. an ontology) and rules preprogrammed into the system.

The main concept of this paper is to investigate methods for extracting an ontology from heterogeneous data and finding useful rules for processing and predicting complex events in traffic. Since traffic anomalies are usually caused by series of other unpredictable events, they are usually extremely difficult to explain and even harder to predict. Nevertheless, there are some cases in which it is obvious that one event can affect the other. Here, we have attempted to determine how large social events in Stozice stadium in Ljubljana, Slovenia, influence other complex events in nearby traffic systems.

2 DATA

In this work, three different types of data were used:

- **Traffic loop sensor data** from a sensor near the stadium. These data contain information on how many cars have passed by in the last hour (for every 5 minutes, from 2011 to 2013).
- **Parking spots sensor data** at Stozice stadium were used for the same period as loop sensors. The sensors returns information about availability of free parking spots every 5 minutes between 8am and 10pm.
- **Data on 50 major social events** at Stozice stadium were collected for the same period as the other data.

2.1 Detecting Complex Events

Complex events had to be extracted from this time series dataset. In this research, a complex event is defined as an anomaly in traffic. Therefore, congested traffic can be considered to be a complex event. For example, traffic congestion in the evening, when there is usually low traffic, can be considered to be a complex traffic event. In contrast, traffic congestion during the morning and afternoon rush hours on workdays are quite common; therefore, they are not considered to be complex traffic events.

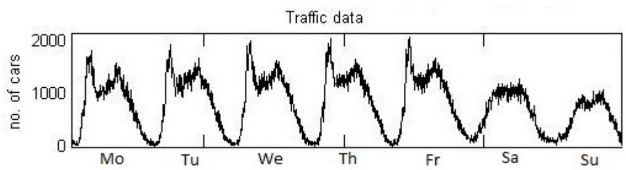

Figure 1: *Traffic loop sensor data for one week.*

There has been a great deal of research in detecting anomalies in data sets [5]. One of the basic methods is to capture the basic statistics of sensor data, for every 30-minute segment of time within each day of the week [6]. This marginal statistic is used to describe “usual” traffic. In order to identify complex events, marginal statistics are compared to real time data. If the difference is greater than a certain set threshold, these states can be marked as anomalous and therefore complex events.

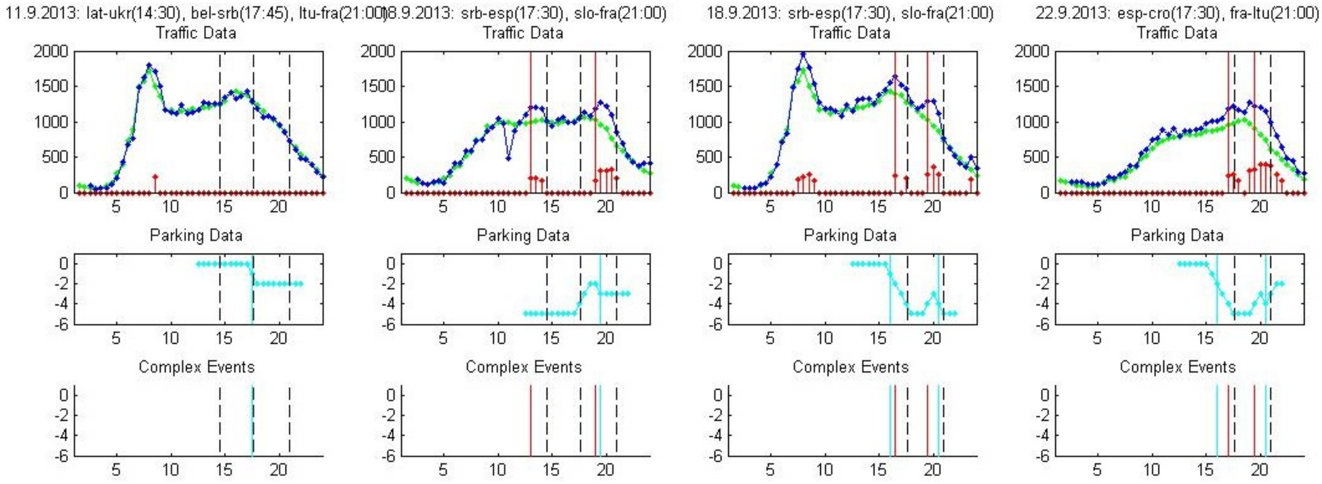

Figure 2: *Complex event detection from traffic loop sensor data and parking sensor data for four bigger social events.*

This relatively straightforward method can be used for anomaly detection in any count data, such as web access logging, security logs in buildings, etc. In the case of this paper, this technique was used to detect congested traffic from traffic loop sensors, as can be seen on Figure 2. Unfortunately, this method could not be used for parking sensor data, because of an inaccurate operation. Complex events for parking data are, therefore, detected as the start of decreasing free parking spots before a social event.

2.2 Data Base

The database was created from extracted complex events. Part of it is seen in Figure 3. Every instance represents one social event in Stozice, which is described with several attributes. For further work in this paper, only the last three attributes were used, since they contain information from all other attributes and are derived from them.

- **Demand attribute** is derived from information on how many visitors had visited one event. It has four possible values, with "1" indicating less than 50% occupancy and "4" indicating a sold out event.
- **Parking sensor attribute** contains information on how soon before a social event, the first complex event happened. It has five possible values; "no" indicating no complex event and "t-0", "t-30", "t-60" and "t-90", meaning complex events happened 0, 30, 60 and 90 minutes before the social event, respectively.
- **Traffic sensor attribute** contains information on events in traffic and has the same possible attribute values as parking sensor. For the predictive data mining part, this attribute is also marked as a target value.

Event Description	Date	Hour	Visitors	Demand	Parking Sensor	Traffic Sensor
…	…	…	…	…	…	…
SLO - UKR	21.09.2013	21:00	10000	4	t-90	t-90
ESP - CRO	22.09.2013	17:30	6050	2	t-90	t-30
FRA - LTU	22.09.2013	21:00	10000	4	t-30	t-90
Elton John	11. 11. 2011	21:00	8000	3	?	t-60
…	…	…	…	…	…	…

Figure 3: *Part of extracted complex events dataset*

3 DESCRIPTIVE DATA MINING

For descriptive data mining part, a well-known method of association rules was used to express regularities in complex events in the database. With these rules, we can better understand our problem and find some interesting unknown correlations between items in dataset. Two main algorithms (Apriori and Predictive apriori) were used with the help of Weka software [7].

3.1 Association Rules

The Apriori algorithm iteratively reduces the minimum support until it finds the required number of rules with the given minimum confidence that can be set as an input parameter. In this work, minimum confidence was set to 0.8.

```
Apriori
=======

Minimum support: 0.1 (5 instances)
Minimum metric <confidence>: 0.8
Number of cycles performed: 18

Generated sets of large itemsets:

Size of set of large itemsets L(1): 12

Size of set of large itemsets L(2): 8

Size of set of large itemsets L(3): 2

Best rules found:

 1. parking=no traffic=no 11 ==> demand=1 11    conf:(1)
 2. parking=t-90 traffic=t-90 6 ==> demand=4 6    conf:(1)
 3. traffic=no 18 ==> demand=1 17    conf:(0.94)
 4. traffic=t-90 8 ==> demand=4 7    conf:(0.88)
 5. demand=4 traffic=t-90 7 ==> parking=t-90 6    conf:(0.86)
 6. parking=no 17 ==> demand=1 14    conf:(0.82)
 7. demand=3 10 ==> traffic=t-60 8    conf:(0.8)
```

Figure 4: *Association rules from Apriori algorithm.*

From the Apriori algorithm results on Figure 4, we can see that the rules with the highest confidence level are the trivial ones. We read this rules like this: if we know that there is a social event in Stozice tonight, and if there is no complex event in parking sensor stream, and there is no complex event in traffic sensor stream two hours or less before the

game, then we can assume that the demand of the game is "1". From Rule 7, we can explain a complex traffic event. If there is a social event with demand 3 in Stozice, it is likely that there will be a congested traffic 60 minutes before the event.
The second algorithm used was Predictive Apriori. It searches for n number of rules (that we define as an input parameter) with an increasing support threshold, concerning a support-based corrected confidence value.

```
 1. parking=no traffic=no 11 ==> demand=1 11    acc:(0.98487)
 2. parking=t-90 traffic=t-90 6 ==> demand=4 6    acc:(0.96475)
 3. traffic=no 18 ==> demand=1 17    acc:(0.93259)
 4. demand=3 parking=t-90 3 ==> traffic=t-60 3    acc:(0.91412)
 5. demand=1 parking=t-60 2 ==> traffic=no 2    acc:(0.86503)
 6. parking=no traffic=t-30 2 ==> demand=1 2    acc:(0.86503)
 7. parking=no traffic=t-60 2 ==> demand=3 2    acc:(0.86503)
 8. traffic=t-90 8 ==> demand=4 7    acc:(0.8065)
 9. parking=no 17 ==> demand=1 14    acc:(0.78475)
10. demand=4 traffic=t-90 7 ==> parking=t-90 6    acc:(0.78107)
11. demand=1 22 ==> traffic=no 17    acc:(0.75941)
12. demand=3 10 ==> traffic=t-60 8    acc:(0.74427)
13. demand=2 5 ==> traffic=t-30 4    acc:(0.70329)
14. traffic=t-90 8 ==> demand=4 parking=t-90 6    acc:(0.69036)
15. demand=4 parking=t-90 8 ==> traffic=t-90 6    acc:(0.69036)
16. demand=4 12 ==> parking=t-90 8    acc:(0.63589)
17. traffic=t-60 12 ==> demand=3 8    acc:(0.63589)
18. parking=no 17 ==> demand=1 traffic=no 11    acc:(0.62519)
19. demand=1 traffic=no 17 ==> parking=no 11    acc:(0.62519)
20. demand=1 22 ==> parking=no 14    acc:(0.61724)
21. traffic=no 18 ==> demand=1 parking=no 11    acc:(0.58557)
22. demand=4 12 ==> traffic=t-90 7    acc:(0.54765)
23. parking=t-90 14 ==> demand=4 8    acc:(0.53877)
24. parking=t-0 7 ==> demand=1 4    acc:(0.51846)
25. demand=1 22 ==> parking=no traffic=no 11    acc:(0.47803)
```

Figure 5: Association r*ules extracted with Predictive Apriori algorithm.*

From Figure 5, we can see even more rules that are explaining complex traffic events, e.g. Rule 4 also includes complex event regarding parking places. We can understand this rule this way: if we know that there is a social event with demand "3", and we know that there is a complex event in parking sensor streams, it is highly likely that there will be traffic congestion 60 minutes before the game. We can also see some other interesting rules (11, 12, and 13) that can explain complex traffic events.

4 PREDICTIVE DATA MINING

Rules can also be extracted with certain predictive data mining methods. In this work, decision trees and rule learner methods were used and evaluated.

4.1 Decision Trees

It is possible to transform any decision tree into a set of rules. In this research, decision tree was built with the help of Weka's J48 algorithm. Complex traffic events were set as the target class. The visualized tree is seen in Figure 6. From this decision tree, we simply derive rules for every leaf. For example, we can assume that traffic will be congested 90 minutes before the game if the game is labeled with demand 4, and so on.

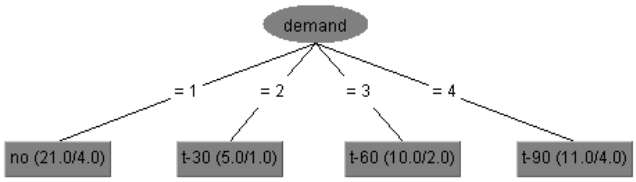

Figure 6: *Pruned decision tree, from which we can extract decision rules*

How complex the rules will be depends on where the tree is pruned. For demonstrating purposes, an unpruned tree was built, which can be seen in Figure 7. In this case, it is seen that parking sensor complex events are also included; therefore, the rules are more complex. Because of the very small dataset (only 50 data samples), we can also see that the number of instances in particular leafs is very low or even zero. Therefore, these rules are not very useful and are shown solely for demonstration of how pruning affects the complexity of rules.

4.2 Rule Learner

Another predictive method that was tested is the decision rule learner. In this case, Weka's JRip (RIPPER) algorithm was used, according to which classes are examined in increasing size, and an initial set of rules for the class is generated using incremental reduced error. The algorithm proceeds by treating all the examples of a particular judgment in the training data as a class, and finding a set of rules that covers all the members of that class. Thereafter, it proceeds to the next class and does the same, repeating this until all classes have been covered.
Figure 8 shows the rules extracted by JRip Algorithm. It can be seen that the rules JRip are the same as those obtained with pruned decision tree.
An unpruned version of JRip algorithm was also tested, and the results are shown in Figure 9. It is seen that, like in an

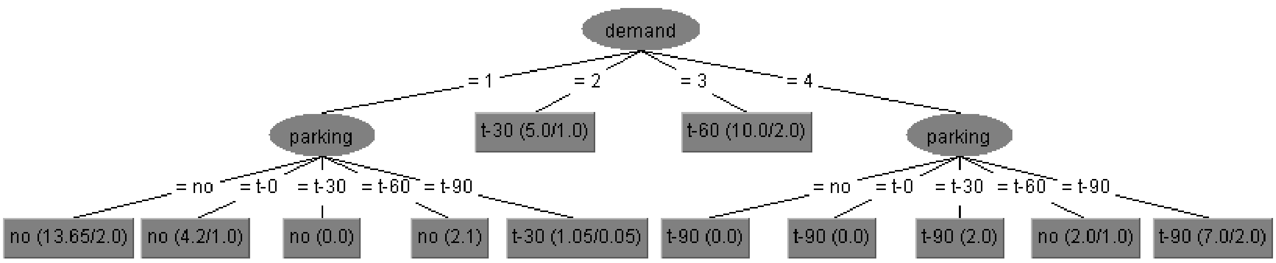

Figure 7: *Unpruned decision tree. Rules are more complex, but number of instances in the leaves is very low because of small dataset.*

unpruned decision tree, the rules are more complex, but also with very few instances in the rules (see Figure 9) because of the small dataset.

```
JRIP rules:
===========

(demand = 4) => traffic=t-90 (11.0/4.0)
(demand = 2) => traffic=t-30 (5.0/1.0)
(demand = 3) => traffic=t-60 (10.0/2.0)
 => traffic=no (21.0/4.0)
```

Figure 8: *Rules extracted with pruned JRip algorithm.*

```
JRIP rules:
===========

(demand = 4) and (parking = t-30) => traffic=t-90 (2.0/0.0)
(demand = 4) and (parking = t-90) => traffic=t-90 (7.0/2.0)
(demand = 2) => traffic=t-30 (5.0/1.0)
(demand = 3) and (parking = t-90) => traffic=t-60 (3.0/0.0)
(demand = 3) => traffic=t-60 (7.0/2.0)
 => traffic=no (23.0/5.0)
```

Figure 9: *Unpruned JRip rules.*

4.3 Evaluation

We can evaluate predictive algorithms and see how the rules extracted with these methods are useful for prediction. Algorithms were compared to the ZeroR algorithm that represents a type of baseline. From the results in Figure 10, we can see that all mentioned algorithms performed better than ZeroR. It is also seen that the evaluation results for the pruned decision tree and pruned JRip rules are entirely the same. This makes sense, since both algorithms extracted the same rules. However, unpruned versions of both algorithms are also the same, but this is more a coincidence since the rules are different. This is probably due to the small dataset. We can also see that pruned algorithms performed better than unpruned ones, which was expected since unpruned models were used only to demonstrate how to extract more complex rules.

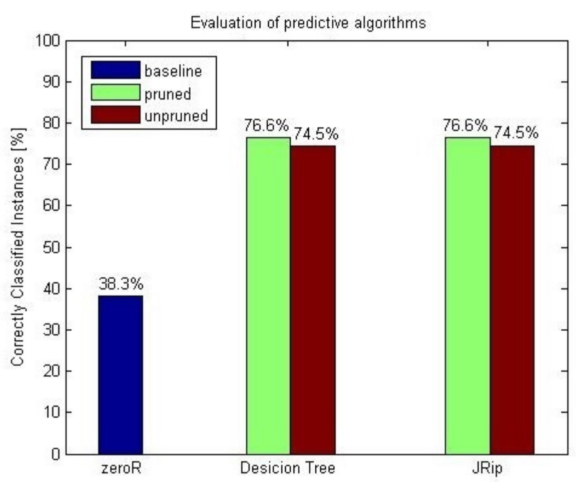

Figure 10: *Decision tree rules and JRip rules evaluation.*

5 CONCLUSIONS

We can conclude that both the descriptive data mining and the predictive data mining methods described in this paper are useful for better understanding of datasets as well as for extracting rules from the given dataset. The author is aware of the problems with the small dataset used in the presented work. Consequently, the next plan is to enlarge the dataset, if possible, and to use these methods on other larger datasets. Regardless of the small dataset, this paper shows that the suggested methods return feasible results and, therefore, can be used to develop advanced travel information systems that can detect, process and predict complex events in traffic.

ACKNOWLEDGEMENTS

This work was supported by the Slovenian Research Agency and the ICT Programme of the EC under MobiS (FP7-318452).

References

[1] Project Mobis, *Personalized Mobility Services for energy efficiency and security through advanced Artificial Intelligence techniques, FP7-ICT*

[2] E. Olmezogullari, Online Association Rule Mining over Fast Data. *Proc. 2013 IEEE International Congress on Big Data (BigData Congress), Santa Clara, CA*. 2013.

[3] D. Anicic, S. Rudolph, P. Fodor, N. Stojanovic. Stream Reasoning and Complex Event Processing in ETALIS, *Proc. Semantic Web – Interoperability, Usability, Applicability, Vol. unpublished: under review (2010), pp. 1-10*

[4] F. Terroso-Saenz, M Valdes-Vela, C. Sotomayor-Martinez, A cooperative approach to traffic congestion detection with complex event processing and VANET, *Proc. IEEE Transactions on Intelligent Transportation Systems, Vol. 13,* 2012.

[5] A. Ihler, J Hutchins, P Smyth, Adaptive Event Detection with Time-Varying Poisson Processes, *Proc. KDD '06 Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 207-216*

[6] E. Horvitz. J. Apacible, R. Sarin, L. Liao, Prediction, Expectation, and Surprise: Methods, Designs, and Study of a Deployed Traffic Forecasting Service, *Proc. Twenty-First Conference on Uncertainty in Artificial Intelligence, UAI-2005, Edinburgh, Scotland, July 2005.*

[7] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten (2009); The WEKA Data Mining Software: *An Update; SIGKDD Explorations, Volume 11, Issue 1.*