Mining Travel Data with a Visualiser

Colin Ho
BT Asia-Pacific
hoc@hongkong.btap.bt.com

Ben Azvine
BTexact Technologies
ben.azvine@bt.com

Abstract: The rapid advances of mapping technology have provided new challenges to the KDD communities in exploiting its potential for visual data mining on spatial data. This paper discusses the issue of integrating this technology with data mining methods to provide an interactive environment for exploratory data analysis. We demonstrate our ideas by building highly interactive interfaces that integrate these tools together to enable collaboration between human experts and the system in the process of mining travel data. More specifically, we describe a travel visualiser and the role it plays in the knowledge discovery process.

1 Introduction

Visual data mining has emerged as one of the most popular and powerful techniques to discover hidden patterns in large volumes of data. This is not surprising as visual pattern recognition skills far exceed our ability to comprehend collections of texts and numbers. This sole benefit usually results in active user participation in the process of knowledge discovery, which in turn facilitates the development of better algorithms and processes for data mining in revealing interesting, hidden patterns and relevant anomalies in the data.

Over the past two decades, we have seen rapid advances in the development of visualisation methods and tools for exploration of large amounts of spatial data (Andrienko and Andrienko1999, Kraak and MacEachren 1999). Many digital data generated today has embedded geographical information like coordinates (latitude and longitude) and postal codes. This information can be fully exploited to provide a highly interactive environment to explore and present dynamic spatial data.

This paper discusses the issue of integrating spatial visualisation and data mining tools with the database. The goal of this integration is to provide a highly interactive tool that facilitates both the process of uncovering patterns and relationships in large, complex data and providing explanation of those patterns and relationships.

To address this issue of bringing these technologies together we need to carefully design an interface that will provide an environment for exploratory visual data analysis. However it would be difficult to build an interface that will fit all types of database as each has its own unique characteristics. Our approach is to develop domain-specific manipulation tools that integrates data mining methods and visualisation tools that enable human and machine to work together in the process of pattern discovery, and integrating databases with visualisation to query for relevant information to enable thinking, hypothesis generation, and problem solving.

Applications using this technology have been used to solve a wide range of business-critical problems, including detecting telephone calling fraud, estimating the traffic flows in cities, and managing resources in a tightly controlled environment. In this paper, we illustrates our ideas by describing the use of this technique to improve the estimation of travel times based on information gathered by BT field engineers.

In the next section we describe the application domain and the rationale behind the work. Section 3 describes the travel visualiser which uses the mapping technology to display travel patterns that let the user quickly see unusual patterns, Section 4 covers the role of the visualisation system in the knowledge discovery process. The last section contains conclusions and recommendations.

2 Travel Time Estimation

Any organisation with a large mobile workforce needs to ensure efficient utilisation of its resources as they move between tasks distributed over a large geographical area. BT employs around 20000 engineers in the UK who provide services for business and resident customers such as network maintenance, line provision and fault repairs. In order to manage its resources efficiently and effectively, BT uses a sophisticated dynamic scheduling system to build proposed sequence of work for field engineers. This system is typically developed to schedule tasks and activities for field engineers in accordance with predetermined rules governing cost, travel and business targets. The scheduler has the ability to modify the sequence in real-time to accommodate the dynamics of resource availability if new high priority tasks appear on the system.

A typical schedule for a field engineer contains a sequence of time windows for travel and task. To generate accurate schedules the system must have accurate estimates for time taken between tasks referred to in this paper as "travel time" and estimates for task duration. By using visual data-mining techniques we have implemented a system that improves the accuracy of "travel time" estimates by 30% compared to the previous system. Under the old "travel time" estimation system many engineers mainly due to underestimation of "travel time" were not able to arrive on-time for their next task resulting in knock-on effect and inefficient schedules. This was evidenced by the our preliminary analysis that for some end-of-day tours field engineers travelled from region *A* to region *B* and then back to a location close to region *A*, causing a criss-crossing effect on the overall schedule. This deficiency points to an underlying problem of providing accurate estimates for "travel times" between jobs.

The system collects event logs on the activities undertaken by a field engineer. Typically, this information comes from a system that monitors the workflow from the moment it issues a task to a field engineer till the task is completed. These event logs are also recorded and stored in a central database. A typical event log includes the field engineer information, the time a new task is issued, the location and region code of the destination site, the arrival time on site, the time the engineer accepts to carry on the task, and the time the task is completed.

Note that the "travel time" is calculated as the difference between the time a task is issued and the arrival time on site. Specifically, a travel time includes the time required to leave the current site, walk to the car-park, start the car, drive to the destination site, park the car, and arrive at the door-step of the next customer. In most cases a large proportion of the travel time is driving from one site to another and the car-park is within the premises of the site. Unfortunately this may not be the case in the urban area like London or city centres where the engineers may take a substantial amount of time to find a car-park and to walk from the car-park to the door-step of the customer.

Travel time is typically treated as a fixed overhead when scheduling jobs, and is extremely difficult to quantify, because factors such as road conditions, weather, vehicle type, route disruption, driving behaviour, traffic peak periods, etc. all contribute to journey times, making it difficult to prescribe an expected inter-job

journey time. More specifically, given a site location *A* and the destination location *B*, we want to predict the time, under a typical condition, it will take for a field engineer to travel from site *A* to *B*.

Our own experiences in driving tells us that a collection of our past journey experiences, taking into consideration the above factors and discarding those abnormal travels where unusual events like accident occurred, gives a good prediction of travel times. Hence we define a theory to predict travel time as follow:

> Definition 1: If *m* journeys take *n* minutes to travel from site *A* to *B*, then it is likely that it would take *n* ±15 minutes to travel from *A* to *B*.

From this theory, we know that it is crucial to carefully select the typical journeys to build a prediction model. For example, we should include journeys where there are road works that last a period of time, but should exclude those where an accident has occurred. We use the visual data mining approach to solve this problem as it would provide a useful tool to identify unusual travels.

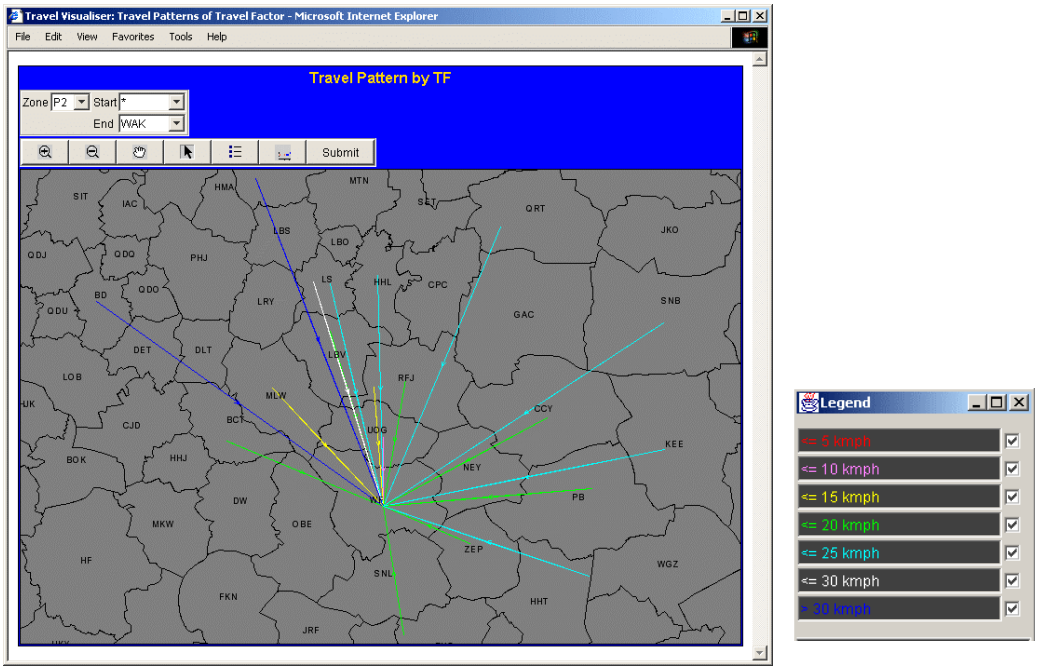

Figure 1: A visualiser displays travel patterns based on estimation model. It shows all travels ending in the region "WAK". The legend on the right shows the colour-codes representing each travel.

3 The Travel Visualiser

Existing visualisation tools, like scatter plot and histogram, are powerful visual aids to show the relationship between two attributes in a data. Over the past few years, we have seen a new generation of computerised visualisation tools is represented including MineSet™, a data mining software from Silicon Graphics [Brunk et al., 1997] and XGobi [Swayne et al., 1998]. The advanced visualising models of MineSet™ include Scatter, Map, Tree, and Evidence Visualiser. Although these tools are useful in many ways, they do not meet our needs in visualising travels, where geographical information is one important source of knowledge.

In order to help users actively explore and interpret data of their interest, we have designed a tightly coupled interface of an interactive system that provides the visualisation of travel patterns with facilities for geographical information, scatter plot, colour-coding, direct data querying, data drill-down, identifying hot-spots, and travel explanation. The visualiser is also integrated to other travel visualisation tools like AutoRoute. In this section, we briefly describe the use of each of these facilities.

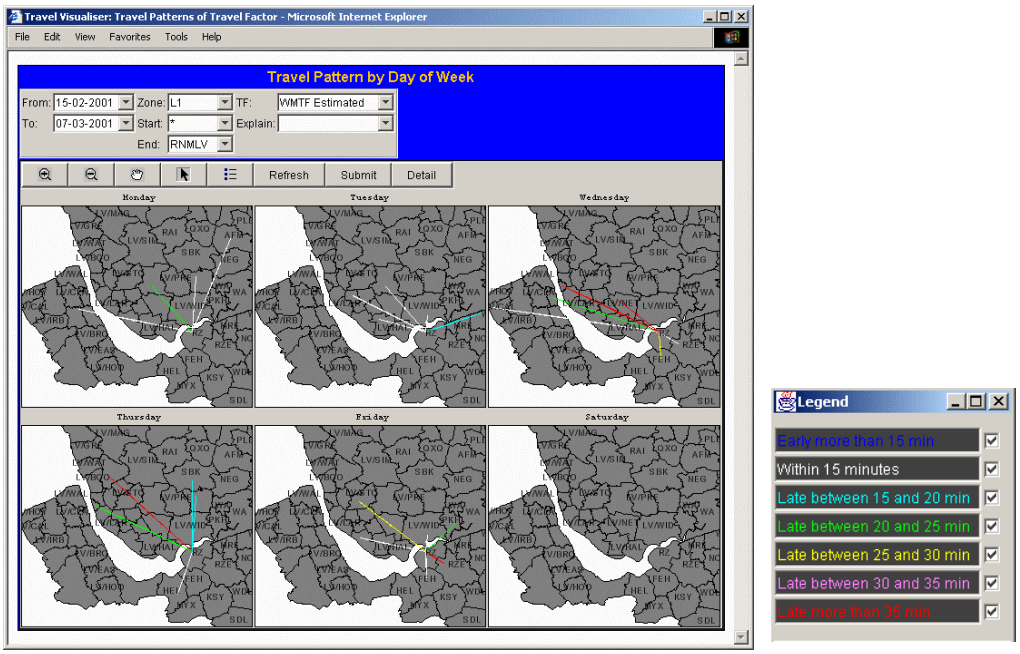

Figure 2: A visualiser displays travel patterns based on day of week. It shows all travels recorded for the period from 15 Feb to 7th March that starts from all regions and ends in "RNMLV". Each map represents all travels done for the day, for example 19/2, 26/2, and 5/3 are Monday. Each coloured line represents the performance of a travel based on the setting specified in the legend, shown on the right, for example a red line shows that the travel is late for more than 35 minutes.

We use the geographical components in the data to display the results on a map. Maps are more informative than simple charts and graphs, and can be interpreted more quickly and easily than spreadsheets or 2-D graphs. Each journey is represented on a map by a line using the X-Y co-ordinates of the start and end locations and an arrow to show the direction of the travel. This simple plot also implicitly reveals the distance between the two points. The performance of each travel is categorised into groups based on the speed of travel, for example <= 5kmph, <= 10kmph, etc. We then define the legend by assigning different colour codes for each of these categories. What makes it all come together is a visual display of the travels on the map. Figure 1 shows a visualiser displaying travel patterns generated by an estimation model. The user can select the scatter plot button to display the relationship between speed and distance.

3.1 Mapping Technology

We include the facility to provide basic geographical information of the region, which is an important source of knowledge, by integrating the mapping technology in the visualiser. The map is based on the same mapping technology used in products such

as MapInfo Professional and Microsoft Map. It adds powerful mapping capabilities to the visualiser as it can display information in a format that is easy for everyone to understand.

Figure 2 shows a visualiser that displays travel patterns based on days of week. By viewing the map, we can immediately know that a field engineer would need to make a detour to reach the destination because a river or canal separates the two regions, thus prolonging travel times. However, we might not be able to draw such a conclusion if the visualiser does not contain the geographical information. This feature lets the users see patterns and relationships in the mass of information quickly and easily without having to pore over the database.

The map was designed such that each region represents an area code of the telephone numbers. It gives an abstract view of the region and omits geographical details such as roads and highways so as not to overwhelm the user with too much information. This is consistent in practice as field engineers usually have local knowledge of the regions and may use different approach routes to the same destination.

3.2 Direct Data Querying

A fundamental function of a visualiser is to allow the user to interact with the data directly for additional information. Users can click on the map to select a journey and the system extracts and displays all information relating to the travel from the database. This facility is essential to exploratory data analysis as the user can generate and verify hypothesis about a travel pattern by performing data drill-down, which we will explain in the next section. For example, a user may want to request information for a peculiar travel so that he can verify it against travel patterns by period of day to see if other travels have the same behaviour.

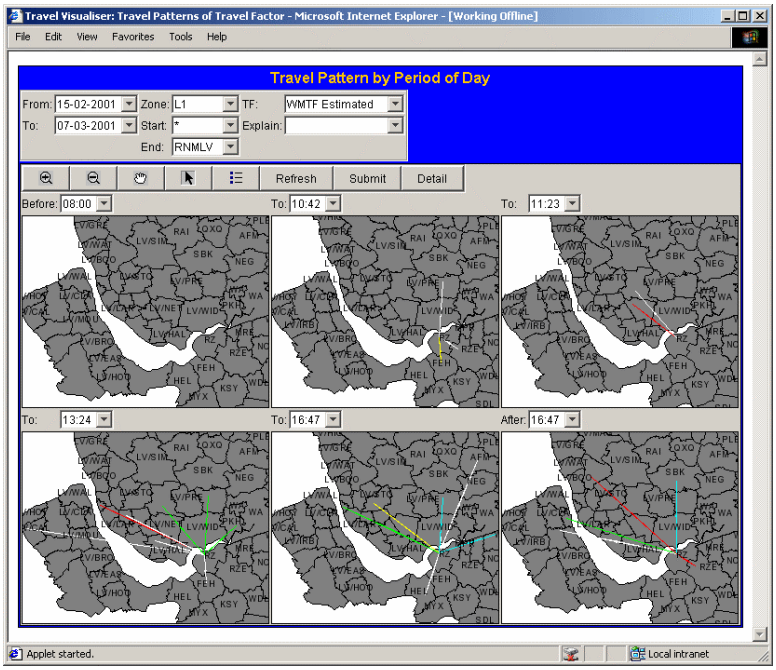

Figure 3: A visualiser displays travel patterns based on periods of day from 15 Feb to 7th March that starts from all regions and ends in "RNMLV". Each map represents all travels done for the period, for example the first map shows travels done before 8a.m. while the second map from 8a.m. to 10:42a.m.

3.3 Data Drill-Down

In order not to overwhelm the user with too much information on a single map, we provide multiple views of travel patterns including evaluation of estimation model (see Figure 1), days of week (see Figure 2), periods of day (see Figure 3), and driving behaviour of engineers (see Figure 4). Each view has been designed to be used by different user groups with distinct requirement for what analysis should be done and how data should be displayed.

This is done by providing numerous filtering utilities in each view. For example, an abstract view can be displayed by selecting all travels from all start and end regions, whereas a filtered view can be obtained by selecting a specific start or end region as shown in Figure 2. The flexibility to select a date range would allow a user to view travel patterns by days, weeks or months. This is particularly useful when it is used to monitor the trend of travels in a region where a road work begins or ends. An operation staff may also use the visualiser to aid in decision making when assigning a high-priority task to an engineer who has a faster approach route to the site.

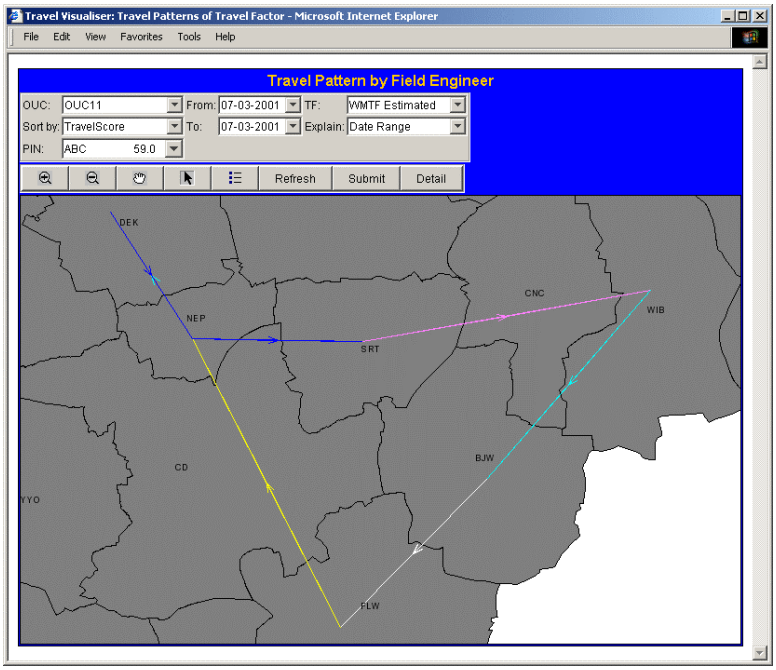

Figure 4: A visualiser displays the end-of-day tour of a field engineer, ABC, on 7th March 2001. It also displays the travel behaviour as compared to the overall engineers. In this case, ABC has a travel score of 59%, where 100% represents the best travel ranking.

3.4 Identify Hot-Spots

The ability to display hot-spots is an important issue in many visualisation systems. To address this issue, we provide facilities to allow the user to switch on travel hot-spots. The system also suggests period hot-spots intelligently.

A legend window, as shown in Figure 2, shows the representation of the colour-codes on travels. The user can select the travels they want to see on the map. For example if the user want to identify hot-spots, which represent travels that are late for more than 35 minutes, then all colour-codes, except red, in the legend are

unselected. This flexibility allows the user to quickly identify travel hot-spots and performs further investigation by using one or more drill-down views.

Another feature of the travel visualiser (see Figure 2) is to automatically identify period hot-spots and display travels according to their respective periods. The system first reads the time-related attributes, like start and arrival time, and performs a supervised discretization (Fayyad and Irani, 1993) using the travel categories as the teacher. This feature shows the relationship between time-related attributes and travel and the user can easily identify on-peak and off-peak periods in a region.

3.5 Travel Explanation

One of the most fundamental issue in knowledge discovery is the ability to distinguish the similarities or differences between a particular record against the numerous patterns discovered from the mass of data. For example in the case of travel patterns, the user identifies a peculiarity on a particular travel and would like to know which other travels share the same behaviour. This would allow the user to gain deeper insight into the underlying patterns in the data.

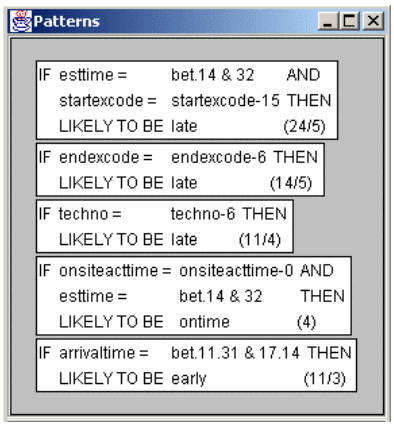

Figure 5: The system offers 5 likely explanations with respect to the travel selected. Each box is a rule whose conditions match the values of the travel. The bracket at the end of each rule represent (total classified / incorrectly classified).

We address this issue by designing an explanation module and integrating it into the visualiser. The process of generating an explanation model including data selection, transformation, attribute selection, model building, and rules matching. This module has been designed to be used by different user groups. The user first decides the scope of data to build the model by selecting the values provided. For example in Figure 4, the user selects the date range to scope the explanation model, which in this case all travels recorded on the date, 7th March, will be used to build the model. Alternatively, the user, who is a team leader, may want to scope the data to his own OUC group. An explanation model will then be build based on the values selected.

The system then extracts the data from the data warehouse. The data then undergoes a series of transformation process including global discretization of all numerical attributes (Fayyad and Irani, 1993), global grouping of attribute values (Ho and Scott, 2000), and removal of irrelevant attributes. The system then use the transformed data to generate an explanation model which is in the form of production rules (Quinlan 1993).

When the user selects a travel on the map, the system will pass through all rules in the model and display those rules whose conditions match the values of the

travel. Figure 5 shows a typical explanation window displaying the patterns. The user may treat this as new hypotheses and investigate them using one or more drill-down views. Hence this module augments the data drill-down facilities in the system.

3.6 Integration to other tools

Another auxiliary feature of the visualiser is the integration of other commercial travel visualisation products like AutoRoute 2001, which provide driving direction from address-to-address. The user can select a travel on the map and request AutoRoute to suggest a detailed route. This facility would further facilitate the patterns searching process as the data analyst would have a rough idea of what route an engineer might take. It can also be used as a coaching tool for field engineers who do not have domain knowledge of the region.

4 Knowledge Discovery Process

We have described the travel visualiser and highlighted some of its features. As can be seen, this visualiser has been designed as a tool to be used for mining travel patterns. In this section, we will briefly describe the role it plays in each stage of the knowledge discovery process.

4.1 Data pre-processing: Cleaning and Transformations

As many data miners will testify with us that a large proportion of the knowledge discovery time is taken up within this stage. Visual data mining attempts to shorten this time by providing opportunities for closer interaction between the data, domain experts, and data miners. Hence, it is important that the visualiser is designed in such a way that domain experts, at different levels, could easily gain insights on the data.

From Definition 1 we know that it is crucial that we accurately select m journeys to simulate "normal" travels, which include routes that have road works over a period of time, peak periods. However, we should exclude "abnormal" travels, including those that do not meet the legal travel requirements, for example, speed at 100 miles per hour, and travels that were held up by an accident. Such events are difficult to spot as there is no travel description recorded in the data.

By using the visualiser, the domain experts could easily point out travels that are classified as "abnormal". For example field engineers are required to sign on to the network to register their start-of-work for the day. The system keeps a record of the default sign-on location for each engineer and travel times are calculated from this location to the first task location. However, the visualiser shows that a large proportion of these travels are either late or very early, and one domain expert explains that an engineer may not be in the default location as registered in the system when they signed on to the network. Such a revelation prompts us to remove all first job travels from the training data. With such active user participation and the aid of the visualiser, we are able to implement rules to remove unreliable data from the training set.

After the data cleaning process, the data are transformed, reformatted, and the visualiser is used to verify the correctness of the new data. One such transformation is the discretization of time-related variables like start and arrival time. As shown in Figure 2, we could use the visualiser to evaluate the effectiveness of such data transformation.

4.2 Data Mining and Evaluation

This step can be viewed as the automated application of data mining algorithms to build a predictive model that fit to the data (for example, a regression tree, a linear function, a set of fuzzy rules, etc.). The predictive model is then subjected to critical evaluations on the quality of the output, but unfortunately this step is often ignored and usually limited only to the statistical evaluation.

The travel visualiser, integrated with the explanation module, is a useful tool for evaluating the model generated by the data mining algorithms. As can be seen in Figure 5, the value of the *esttime* is estimated by the output model. This offers some indications to the quality of the model. The user can use one of more views to verify their evaluations and provide feedback to the data miner. This may suggest a need to further clean or transform the data, forming a feedback loop to the KDD process. This cycle is repeated until the user is confident with the results.

4.3 Deployment

The travel trends change from day to day, and there is a need to update the estimation model to match the latest condition on the road. We automate the overall KDD process from data selection to model construction so that the model can be used continuously in real-time for estimating travels.

The visualiser has the facility to allow the user to select the type of model for estimation, i.e. current and recommended. When the system activates the model update process, the current and recommended will produce the same travel estimations. As more travels are added to the system, the recommended model will gradually be different from the current. The user can use such facility to help them access the models and decide when to perform the next model update. This flexibility increases the confidence of the user in the overall system.

5 Conclusions and Future Work

We have demonstrated that visual data mining offers many benefits to the KDD process. We have carefully designed a visualiser which has complete integration to the database and other "off-the-shelf" visualisation tools.

The use of the mapping technology in the visualiser has added another dimension to visual data mining. By providing multiple maps in a single view, the user could easily recognise complex dependencies between many attributes. This encourages active user participation as they could apply their perceptual abilities to gain insights to the underlying patterns of the large data sets. By using the visualiser as a tool for dialogues between users and developers, it speeds up the overall KDD process. When a visualiser is designed to be used at every stage of the KDD process, it increases the quality of the output model. This is evidence in our project as the accuracy of our proposed system significantly outperforms the existing system by thirty percent, which in turn causes the scheduler to produce a higher quality tours of work. This also improves customer service as the company could allocate a smaller waiting time window for the customer.

Researchers should look beyond scatter plots in designing a visualisation tool. The integration of the mapping technology in a visualiser opens up many new challenges in visual data mining, particularly when there are geographical components in the data. Data visualisation is often restricted to general-purpose tools provided by most commercial data mining packages. Many such tools are usually catered for scientists but not domain experts. In many situations, developing an

application specific visualiser would significantly improve the environment for the data exploration process.

Acknowledgement

We wish to acknowledge Richard Maxwell from BT Retail and Ted Lawson from BT Wholesale for their help and guidance during the trial and implementation stages of the system.

References

G. L. Andrienko & N. V. Andrienko. Interactive Maps for Visual Data Exploration, in *International Journal Geographic Information Science*, 13 (4), pp. 355-374,1999.

C, Brunk, J. Kelly, and R. Kohavi. Mineset: An Integrated System for Data Mining, In *Proc. KDD-1997: The Third International Conference on Knowledge Discovery and Data Mining*, Menlo Park, CA, 1997.

U. M. Fayyad and K. B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In *Proc. IJCAI-1993: Thirteenth International Joint Conference on Artificial Intelligence*, Morgan Kaufmann Publishers, Los Altos, CA, pp. 1022-1027, 1993.

K. M. Ho, and P. D. Scott. Reducing Decision Tree Fragmentation Through Attribute Value Grouping: A Comparative Study, in *Intelligent Data Analysis Journal*, 4(1), pp.1-20, 2000.

M. J. Kraak and A. MacEachren. Visualization for exploration of spatial data. In *International Journal of Geographical Information Science*, 13(4): pp. 285-287, 1999.

J. R. Quinlan. Programs for Machine Learning. Morgan Kaufmann Publishers, Los Altos, CA, 1993.

D. F. Swayne, D. Cook, and A. Buja. `XGobi: Interactive Dynamic Data Visualization in the X Window System. *In Journal of Computational and Graphical Statistics*, **7**(1), pp. 113-130, 1998.