Applying data mining methods for cellular radio network planning

Piotr Gawrysiak, Michał Okoniewski
{gawrysia, okoniews}@ii.pw.edu.pl

Institute of Computer Science, Warsaw University of Technology
ul. Nowowiejska 15/19, 00-665
Warsaw, Poland

Abstract: This paper contains description of an knowledge discovery experiment performed in radio planning department of one of Polish celular telecom providers. The results of using various data mining methods for GSM cell traffic prediction are presented. The methods used include both standard and well established approaches such as decision trees and k-means clustering, and new methods invented for this experiment, such as regressional clustering. Remarks on importance of discretization methods for quantitative data mining are presented, together with general discussion on data mining of technical (i.e. mostly numeric and automatically generated) data.

Keywords: data mining, automatic knowledge discovery, cellular networks, clustering, multiple regression, discretization

1 Introduction

Typical applications of data mining are usually associated with financial or marketing problems. Market segmentation, customer practices, fraud detection are areas commonly aimed with discovery tools. These problems are similar for a number of organizations - and the data that may be found in these organizations look much alike. Finance and marketing are also places where knowledge may be turned into profit most easily. It is different with in engineering. There are no easy schemes like "customer - sales - profit". Quantitative data are more common than qualitative, because even smallest device may produce huge amounts of log files about its work parameters. On the other hand there is no fast and obvious way of converting discovered knowledge into marketable form. This knowledge may improve functionality of devices, or help to plan future technology investments. It may bring mostly long term profits, so it is not that easy to convince executive staff to invest in knowledge discovery software and services for engineering. Nonetheless, there is much room for new KDD research in these technology-related areas.

Data mining team, which was formed in the Institute of Computer Science at the Warsaw University of Technology, has been asked to investigate the possibility of applying data mining techniques for one of Polish cellular telecommunication companies. Because the idea of this project came out from technological departments of this company, the

mining team started its research in their area. The problems that have been identified in department, as good candidates for data mining, were not the typical applications of KDD in telecommunications such as described in [1] and [6]. Of course typical problems, such as churn analysis or customer segmentation are also very important in telecom companies, but as they are obviously related to marketing, we have not been yet investigating them. Moreover, the purely technical problems that were analyzed with data mining methodologies proved to be much more interesting, as the classical knowledge discovery methods could not be applied here.

This paper describes a first one in longer series of knowledge discovery experiments in this company as our team started with data mining in the sub-department of radio network planning division. The project consists of much more experiments, some of which are still ongoing, and may be described in more detail in further papers.

2 Problem definition

One of the most important areas for a young cellular telecom provider is network expansion. This creates a need for traffic prediction i.e. we would like to estimate the number of calls made during a certain time span, on an area where we want to build a new base station. Such information is crucial for station equipment design - there must be enough transceivers to ensure that every subscriber in the GSM cell created by this station is able to place or receive call. On the other hand, there should not be too much available - and unused - radio channels, because this would mean unnecessary costs.

Traffic prediction is a complex task, as the number of subscribers present on a certain area may vary. After all GSM is an abbreviation of name Global System for Mobile Telecommunications, and GSM subscribers travel between cells, for example moving into city centers at day, and going to suburbs (where their homes are) in the evening. Similar effect can be observed also for longer time periods. So called vacation traffic analysis shows that in the summer average traffic generally increases in popular resort areas - like mountains, seashore etc.

Fortunately our analysis showed that these variations are periodic and predictable, at least for regions with well developed GSM coverage. We can therefore try to predict traffic for a certain characteristic time period - say, for vacation time - using measurements of existing network elements, and than interpolate obtained values.

In this particular experiment we were able to extract two types of information from ERA GSM network monitoring system. First was the traffic information: for each cell we obtained average of weekly traffic measured at busy hour (usually around midday). Other type of information has been extracted from company's geographical information system (GIS): for each cell, the types of terrain occupied have been established. The GIS database contained information about nine terrain types (landuses) that may occur in particular cell. These were:

1={Fields}, 2={Forests}, 3={Water}, 4={Rocks, seashores, swamps}, 5={Roads, concrete, parks}, 6={Suburbs} 7={Urban area}, 8={Dense urban area}, 9={Industrial

area}

For each cell the amount of ground pixels occupied by every landuse have been measured, where one ground pixel width and length is approximately 5 arc seconds of parallel or meridian respectively.

Our initial data about existing network have been collected and recorded in a table with following attributes:

- cell identification number
- landuse type (a number $\{1..9\}$ corresponding to above landuse types list)
- number of pixels occupied by this landuse
- area number that allows to determine region in which the cell is situated
- average weekly traffic value in Erlangs for this cell

In the latest experiments additional data about the population on every cell area was added. These experiments are still in progress, so they are not fully described in this paper. Data preprocessing included converting above representation into table that contains percentages of landuses for each cell. Both relative and non-relative distributions of landuses were supposed to be meaningful.

3 Statistical foundations of the solution.

In an inception phase of research, a multiple regression have been proposed as a problem solution (see [3] and [5]. Indeed, if we treat the ground pixels as entities generating traffic for a cell than we can represent traffic value as a linear combination of landuse distribution values

$$t(cellnr) = l_1(cellnr) \cdot d_1 + \dots + l_9(cellnr) \cdot d_9 \quad (1)$$

where t represents traffic generated by this pixel, l_n represents the amount of pixels of n -th landuse type in $cellnr$ GSM cell, and d_n traffic density for this landuse type

Above equation can be also written in matrix representation:

$$T = L \cdot N \quad (2)$$

Where T is a traffic vector, L is a matrix of landuses for each cell and D is a vector of density coefficients $[d_1 \dots d_9]$.

In this particular problem, the vector D has as many rows as landuses. This fact is caused by problem definition, that does not specify any constant addend in equation

(1). Of course, landuses are not only factors that may be taken into consideration when estimating traffic. Other factors may be population density, wealth of a region or number of landline telephones installed, so the traffic equation may be formulated in a more complex way. In fact our further experiments involve working with such parameters, but the overall estimation method remains the same.

Using multiple regression we were able to calculate all density coefficient values. In the case presented above, the multiple regression hyperplane which is best approximation in least square sense, may be calculated with the estimator:

$$N = (L^T \cdot L)^{-1} \cdot L^T \cdot T \quad (3)$$

that is a vector of approximated traffic density coefficients in particular landuses. The quality of approximation may be assessed here with estimators

$$\hat{R} = \sqrt{1 - \frac{\sum_{i=1}^n (t_i - \hat{t}_i)^2}{\sum_{i=1}^n (t_i - \bar{t})^2}} \quad (4)$$

$$\hat{R} = \sqrt{1 - \frac{T^T T - N^T L^T T}{T^T T - \frac{1}{n} (1^T T)^2}} \quad (5)$$

R^2 is an estimator of correlation coefficient, that has value range between 0 and 1. $R^2 = 0$ means total lack of correlation - variable T is not correlated with $L_1, L_2, \dots, L_k$. On the other hand $R^2 = 1$ means perfect fit - each point from the population belongs to regression hyperplane.

A subsidiary may be also the estimator of convergence coefficient:

$$\varphi^2 = \frac{\sigma^2}{D^2(T)} \quad (6)$$

Above estimators: (4) and (6) are good measures for comparison of regression approximation quality in different samples of traffic results, and such comparison is necessary as we operate on data collected on various time periods.

4 Initial problems and requirements

The problem that arisen was the unacceptably high regression error rate. This is not surprising as analyzed cells had various sizes and characteristics, and therefore the same amount of landuse pixels can have different impact on total traffic in different cells. Consider for example such three cells:

Cell 1 is a very small cell, probably an inter-urban one. This means that the urban area within is densely populated, and therefore generates lots of traffic. Urban characteristics are probably similar in *Cell 2*, albeit here total traffic will be sum of traffics generated by urban area, and forest area (which probably will have minor contribution to total value). The situation is very different in *Cell 3*. The urban area is very small here and completely surrounded by forest area (it could be for example a forester's house). This means that the number of people residing in this area will be smaller than number of subscribers located in the same type and size area in *Cells 2* and *3*. That suggests the need to categorize all cells into several groups, in which such differences as depicted above will be minor. If we had more information about the cells (such as map of landuse distribution etc.), and less cells to classify, it could possibly be done manually. As such data were not available and the number of cells in the whole network is several thousand, the use of automatic knowledge discovery methods here is fully justified.

Another controversy between network engineers and data mining team at the beginning of research was the interpretation of negative density coefficients. The cellular network experts claimed that each pixel should generate certain amount of traffic within its cell. Nonetheless, the coefficients obtained with regression method often happened to be negative, especially for landuses 1-4 (fields, forests, water, swamp and rocks). The results partially convinced them to change their interpretation: some landuses may deter subscribers from using a cellular phone and generate "negative traffic" (for example forest area in *Cell 3* from above figure could possibly have such coefficient). Of course it is possible to approximate the traffic function with hyperplane that has only positive coefficients, but the regression error (6) would grow in such case.

5 Data mining solutions

The preprocessed data, with traffic and percentages of landuses, were then discretized using quartiles and chi-square discretization techniques. Generally dividing both landuse percentages and traffic into four levels, as in quartile approach, proved to be best for retrieval and visualization of qualitative knowledge about groups of cells. The fastest available data mining approach was the use of existing software tools for the classification of cells. For this purpose such tools as IBM Intelligent Miner for Data and SGI Mine Set were used. The main idea was to find first a classification of cells, and then calculate

multiple regression function for each group. Algorithms used in this experiment were classic k-means clustering and C4.5 decision trees.

5.1 Decision trees

The decision tree based on discretized traffic as a decisive attribute, after pruning, has brought about a result of 12 decision groups (leaves). The number of cells in leaves varied from 12 to 700 cells. The only problem was the purity of classification that was only 61%. For each group of cells multiple regression function was calculated. Most important tests were based on the value of the landuse 5, 6 and 9.

5.2 K-means clustering

Classic clustering did not seem much promising as a method of obtaining a meaningful classification. We anticipated that the euclidean distance measure is not very relevant to desired classification that should follow regressional hyperplane. Anyway, the results were more valuable for the experts than it was expected at the beginning. The population of cells was distributed among various number of clusters: from 2 clusters up to 10 clusters. For each cluster of cells multiple regression function was calculated. Using Intelligent Miner visualization tools we presented results to cellular network experts. They interpreted clusters as "rural", "urban", "suburban" or "industrialized" and agreed that traffic density coefficients in regression function calculated for discovered clusters are meaningful and differences between clusters are significant.

5.3 Regression results for clusters and decision groups.

The regression calculated for groups obtained from decision tree and clustering differs significantly from the regression calculated for the whole population. Below we present few examples, $d_1, \dots, d_9$ are density coefficients in erlangs per pixel:

Group	d_9	d_7	d_6	d_5	d_4	d_3	d_2	d_1
All cells	0.045425	0.254308	0.022617	0.026505	-0.00276	2.99E-05	-9.02E-05	-0.00012
Warsaw	0.144189	1.2673	0.35689	0.015791	0.185147	0.006157	-0.02149	-0.00393
Cluster 1	-0.01539	0.474486	0.011674	0.017678	-0.07205	0.000592	-0.00031	2.61E-05
Cluster 2	0.085386	0.429112	0.265819	0.068346	Not present	Not present	-0.03502	-0.06868
Leaf 1	0.048883	1.376153	0.014656	0.006502	-0.00096	2.67E-05	7.52E-05	4.74E-05
Leaf 2	0.007647	0.036766	0.014682	0.011339	-0.00697	0.00027	0.000182	6.75E-05
Leaf 3	0.027476	0.089261	0.00795	0.017669	-0.00057	0.000141	8.35E-05	1.70E-05
Leaf 4	-0.01431	0.799676	0.012607	0.045727	-0.15842	0.003023	0.000992	3.06E-05
Leaf 10	0.008492	2.632975	-1.49948	1.924176	-0.45692	-0.06189	0.094073	-0.20499
Leaf 12	0.061382	0.425565	0.084185	0.036595	Not present	0.030456	-0.00292	-0.0018

where:

Cluster 1 = $\{l_4 l_7, l_9 = \text{medium}; l_1 = \text{huge}\}$; Cluster 2 = $\{l_5 l_6 = \text{huge}; l_1, l_3, l_4 = \text{small}\}$

Leaf 1 = $\{l_5 = \text{medium} \cup \text{small}; l_6 = \text{small}; T = \text{small}\}$
 Leaf 2 = $\{l_6 = \text{medium} \cup \text{big}; l_5 = \text{medium}\}$
 Leaf 3 = $\{l_6 = \text{medium} \cup \text{big}; l_5 = \text{small}\}$; Leaf 4 = $\{l_5 = \text{medium} \cup \text{small}; l_6 = \text{huge}\}$
 Leaf 10 = $\{l_5 = \text{big}; l_6 = \text{big}; l_9 = \text{huge}\}$; Leaf 12 = $\{l_5 = \text{huge}; l_6 = \text{big}\}$

5.4 Clustering with regressional distance measures

As classic clustering does not seem to optimize the value of estimator R for clusters, there was a need for a new clustering algorithm that will solve the main problem of this project. According to initial assumptions this clustering algorithm has to use estimators of regression quality to assess the quality of clusters. The problem is that we cannot determine the distance between clusters and the distance between records within a cluster. The only way to classify cells in this way is to move cells between two clusters and compare the estimator R before and after the operation. The decision whether to leave the moved cell in the cluster or move it back to the previous place should be taken according to changes in the R estimator for both clusters:

R11, R21, - the values of R before moving a cell record
 R12, R22, - the values of R after moving a cell record between clusters.

- If $R12 > R11$ and $R21 > R22$ - the records should definitely be moved between clusters
- If $R12 + R22 > R11 + R21$ - the cells should be moved
- If $R12 + R22 < R11 + R21$ - the cells should not be moved
- If $R12 > R11$ and $R21 > R22$ - the records definitely should not be moved between clusters

The general form of such algorithm may look like this:

```

1 regress_clustering(clusters_set cl[k], BOOL strict)
2 {
3   cluster a,b;
4   records ra,rb;
5   double R11,R12,R21,R22;
6
7   add_records_to_clusters(cl);
8   while (calculate_general_regression_results(cl) < MAXRESULT)
9   {
10    choose_clusters(a,b,clusters[k]);
11    choose_records_to_move(a,b,ra,rb);
12    R11 = correlation(a);
13    R21 = correlation(b);
14    move_records(a,b,ra,rb);

```

```

15  R12 = correlation(a);
16  R22 = correlation(b);
17  if (R12>R11 && R21>R22)
18    reverse_move(a,b,ra,rb);
19  else
20    if (R12+R22>R11+R21 && strict==TRUE)
21      reverse_move(a,b,ra,rb);
22  }
23  calculate_regression_coefficients_in_clusters (c1);
24 }

```

The function *correlation(cluster)* is used to calculate the quality of regression. It may use estimator formulas (4) or (6) - other ways to estimate the regression quality are also possible. The main loop (*line 8*) is trying to check if the overall clustering is good enough. If it is not, two clusters are chosen. In each cluster algorithm has to find two records - possibly having negative impact on regression quality (*line 11*). Then records are moved between clusters, and the correlation function is evaluated before and after this operation (*lines 12-16*). Results are compared, and if they do not fulfill normal or strict condition, records are moved back to native clusters. This algorithm creates requested number of clusters with requested quantity of records in every cluster. The final quantities are the same as created by function *add_records_to_clusters(cl)*. If necessary it is easy to rewrite above algorithm into one that creates clusters with variable quantity of records, by moving only one record between clusters instead of exchanging records.

These algorithms have been experimentally implemented, using SAS system, but further research is still necessary to find better and more efficient forms of sub-algorithms *calculate_general_regression_results()*, *choose_clusters()*, *choose_record_to_move()*, than those used by us. Although above algorithms are not the typical versions of clustering, they are the right and exact answer to network planners needs. The problem of efficiency is not critical here, because of the fact that cellular network planning is done only once - before the new cell starts to operate.

Another approach that we plan to implement for deriving clusters maximizing regression quality for within-cluster records is genetic algorithm. We can build population entities chromosomes in such a way, that each entity represents certain distribution of cells among predefined number of clusters. The sum of regression quality estimators for each cluster defines a fit function. Having such definitions a classic genetic algorithm procedures, as defined in [8] and [?], may be used. This approach should generate better results than above heuristic method, but the performance maybe worse.

5.5 Association rules

The classic data mining approach of association rules did not show much more than facts revealed by decision trees and clustering. As discretization used by us generated relatively few (three to six) qualitative values, and there were 8, 9 or 10 attributes, the experts of cellular network planning suggested that only rules with support over 10%

and confidence over 50% are meaningful. Such rules either confirmed the knowledge previously obtained by C4.5 tree or clustering or were similar to the experts' engineering intuition. For example association rules confirmed that cells that have large area usually have little traffic, and small cells have relatively more traffic - because they are typically within cities.

5.6 Neural networks approach

Members of the data mining team applied also some artificial intelligence tools to solve the problem. One of them was simple 3 layer non-linear neural network, built with 50 neurons. Network inputs were percentages of landuses, and output was the traffic level. After training the network, the mean square error of predicted traffic values was about 2 erlangs, while average traffic in a cell is equal or a little over 3 erlangs. However the experiments with the network proved the importance of the population data. Adding the population as an input to the network reduced the error rate by 10-15 %.

6 Conclusions

Although the amount of time for the experiment described in this paper was very limited, and the experiment is still running (as for December 1999), it has brought about some interesting outcome. Main results were in this case all the obtained clusters and decision trees with the calculations of multiple regression coefficients and error estimators. All this knowledge is ready to use by the network planning engineers for the purpose of planning new base stations and splitting existing cells. Even more immediate and spectacular results were changes in landuse classification that were made soon after the results of clustering and regression trees were presented by us to the planning engineers. The most important attribute in decision tree appeared to be landuses 5 (roads, concrete, parks) and 6 (suburbs). Experts were surprised finding out that landuse 5 has so decisive role in distribution of traffic. It turned out that the definition of this landuse is too general, and in future planning it will have to be split into two or more subclasses.

Another outcome of the experiment was noticing the importance of discretization methods for such data mining tasks. First applied discretization method was simple use of quartiles. Other preliminary methods used by us distributed records into sets of the same quantity for every attribute. The difficulty with such discretization is that it was useless for some attributes (those having mostly zero values). Because of this, another step was discretization for only these values that are not zero, and appending zero as additional class - in this way quartiles discretization results in five discretized values. Attributes that were equal zero proved to be very important in decision tree tests. For example cell that had no landuse type 6 pixels was mostly classified as "rural" and therefore with below-average traffic.

Another discretization tool that was used in this experiment was based on research results from Warsaw University [5]. We tested mostly chi-square method that for some, but not all of the attributes, seemed to be much better than quartiles - the purity of

a decision tree was better by 10%. This example proves that it is still need for new methods in mining numeric databases. To obtain rules or trees the attributes have to be discretized, and this still means some loss of information. It would be good to find methods of knowledge discovery for quantitative data, that does not need discretization - and it is another topic for future deeper research that is pointed out by this experiment.

The experiment also proved data mining reputation as an area where multi-methodology approach is common and advantageous. Many different techniques coexisted and were mutually supporting - no matter if they were from machine learning field, statistics or artificial intelligence area. For best support for planning traffic in designed cells all obtained models will be used. Four proposed by our team models are : decision tree with multiple regression, k-means clustering with multiple regression, regressional clustering (see 5.4), neural networks. Combining above models should help cellular network engineers to predict with traffic maximal accuracy.

This experiment also proved, that automatic knowledge discovery is a very promising tool also in purely technical applications. As we already mentioned, data mining is still perceived as mostly marketing and business decision tool, yet we think that experiments with applying data mining methodology to technical problems such as process control, systems planning etc. can be very advantageous, and represent an interesting and challenging research opportunity.

Acknowledgments : We would like to thank our scientific supervisors - prof. Mieczysław Muraszewicz and prof. Henryk Rybinski - and also other members of Data Mining Team for support and valuable discussions.

References

- [1] Rob Mattison, "Data Warehousing and Data Mining for Telecommunications", Artech House Computer Science Library, 1997
- [2] Daniel Keim, "Clustering Methods for Large Databases: From the Past to the Future", SIGMOD'99 Conference Tutorial, 1999
- [3] Statsoft Inc., "Electronic Statistics Textbook", Tulsa, 1999.
- [4] Lindgren, Bernard W., "Statistical Methods", McMillan Publishing, 1976
- [5] Michael J. A. Berry, Gordon Linoff, "Data Mining Techniques : For Marketing, Sales, and Customer Support", John Wiley & Sons 1997.
- [6] Andrzej Skowron, Son H. Nguyen, "Quantization of Real Value Attributes", IIPW Report 1995
- [7] D.E. Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning", Addison-Wesley, 1989
- [8] Z. Michalewicz, "Genetic Algorithms + Data Structures = Evolution Programs", Springer Verlag, 1992