International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM)
ISSN: 2150-7988 Vol.2 (2010), pp.243-251
http://www.mirlabs.org/ijcisim

SOM Neural Network - a Piece of Intelligence in Disaster Management

Petr Klement
MEDIUM SOFT a.s.,
Ostrava, Czech Republic
petr.klement@mediumsoft.cz

Václav Snášel
Department of Computer Science
FEECS, VŠB – Technical University of Ostrava
Ostrava, Czech Republic
vaclav.snasel@vsb.cz

Abstract

A collaborative emergency call-taking information system in the Czech Republic processes calls from the European 112 emergency number. Large amounts of various incident records are stored in its databases. The data can be used for mining spatial and temporal anomalies. When such an anomalous situation is detected so that the system could suffer from local or temporal performance decrease, either a person, or an automatic management module could take measures to reconfigure the system traffic and balance its load. In this paper we describe a method for knowledge discovery and visualization with respect to the emergency call taking information system database characteristics. The method is based on the Kohonen Self Organizing Map (SOM) algorithm. Transformations of categorical attributes into numeric values are proposed to prepare training set for successful SOM generation.

1. Introduction

Statistical tools are exploited in public safety information systems to deal with historical incident-related data [3], [15]. Knowledge hidden in this data has to be derived by experts using common outputs like tables and graphs. Intelligent unsupervised knowledge discovery and visualization applied to the emergency situation data are not generally known. Even if research communities are encouraged to tackle the topic [8], the response is poor.

Emergency call taking in the Czech Republic is supported by a distributed collaborative information system operated at fourteen regional emergency call centres, or PSAPs (Public Safety Answering Points). Each of these PSAPs serves emergency calls from its home region primarily. If the home PSAP is occupied or out of order, the system automatically reroutes emergency calls to another PSAP, where the call is processed in the same way as it would be processed by the PSAP of the home region. Every single operator knows the actual operational status and language skills of all the other operators logged into the system. Thanks to the cooperative functionality based on an instant messaging subsystem, which is transparent to the user, operators can ask for help or offer their free capacity and skills in conference mode to the other operators. As all the descriptive and operational data are shared or replicated between system nodes in the background, every operator can receive an emergency call from any region, regardless of his/her position with respect to the location of the incident.

Experience from the operation of the emergency call taking system suggests that in a normal situation calls are smoothly processed by operators in the region where the incident originated, without any special demands on the system settings. In highly critical situations, when many incidents happen in a short period of time (e.g. during storms or floods), or many people are announcing the same incident (e.g. a plane crash, gas explosion or large fire), the system could be locally overloaded. In this case some intelligent reconfiguring scheme would help to balance the system load with respect to the resources available. By means of routing schemes via which calls are distributed, the quality the of service affecting the network throughput and the prioritization of critical services, as well as postponing the replication of less important data to lower network traffic in overloaded regions, can be managed dynamically, with the goal of improving system responses in critical situations.

In order to be able to apply proper and timely management actions, the system must first recognize the critical or anomaly situation. There is a central database, containing all records of emergencies, or incidents, from the whole territory of the Czech Republic. This database can be used for the monitoring and analysis of the current situation as well as for learning from historical incidents and mining spatial and temporal anomalies.

This paper is focused on a practical proof of the basic precondition of the process described above, the detection of an anomaly situation from the historical emergency data. Measures taken after the critical situation is detected are supposed to be applied by a person in this first stage and are not discussed here.

The goal of the presented work is:

(i) Design of a method of searching for patterns in the database of incidents, which would point out certain interesting situation in a time and place.

(ii) The method should reveal these patterns without human involvement and present results to the user in a way allowing fast comprehension and evaluation of the presented output.

In this paper we discuss knowledge discovery in database by unsupervised machine learning, namely the application of the Kohonen Self-Organizing Maps (SOM) algorithm.

In Section (2) we compare our approach with related work; Section (3) describes the principles of the Kohonen SOM algorithm used for clustering; Section (4) deals with features of the SW tools used in our experiments. In Section (5) we present a subset of experiments with SOM algorithm tuning and data transformations applied to the training set to form a suitable search space and reach satisfactory outputs. While the experiments in Section (5) are focused on the nature of the incident, Section (6) concentrates on incident report attributes describing rather the architecture and technology in the system background and suggests the SOM and corresponding analytical tool usage for the PSAP technology performance analysis. In Section (7) we discuss the results and conclude.

2. Related works

Anomaly detection methods for various professional domains have been designed and well described. SOM has been used here e.g. for network intrusion detection [12], [16] and [19], fraud detection [2], mechanical fault detection [21] and anomaly detection in generic time series data [5]. A common approach in using SOM for anomaly detection is to build a classifier distinguishing between anomalous and non-anomalous classes of data. The non-anomalous data are used to create a model of the correct situation. After that a single input vector is presented to the trained SOM and the winner neuron closest to the input vector is found. If the distance from the winner's representative is below a certain limit, the input is classified as belonging to the winner's cluster of non-anomalous data. Otherwise, the input is considered anomalous.

These classification methods assume that either a non-anomalous subspace is known before the learning starts [16] or the resulting clusters are compared with an expert classification [2]. Generating artificial anomalous cases with a Negative Selection Algorithm inspired by the human immune system in combination with a back-propagation neural network [5] falls into this category. Thus the anomaly detection in this traditional view is based on supervised learning.

Based on two facts, our approach is different. First, distinguishing between anomalous and non-anomalous cases within emergency calls is disputable. Second, if an emergency situation exceeded the "normal" scale, it would probably be reported by a set of single emergency cases having certain attributes in common. We are therefore interested in revealing certain patterns in the emergency call data, rather than deciding whether a fresh new case is somehow strange compared to the previous experiences. After the patterns are recognized automatically, the composition is always presented to a person for them to analyze the situation.

To enhance this concept, providing that the algorithm is being run periodically, the new composition will be shown to the supervisor or to the network management module if the composition of the patterns found in the current run is different from the composition formed in the previous run. SOM quality measures would be investigated like the index of the map's disorder [14] or the goodness of the map based on the distances between the winner and the second best-match node [9].

While the SOM algorithm has been widely used for classification tasks, using it for clustering data analysis has been relatively outside the focus of the research community [20]. This paper describes another application of SOM in cluster analysis.

3. Clustering and Self Organizing Maps

Cluster analysis groups objects (data records) into classes (clusters) in such a way that objects in the same cluster are very similar, while objects in different classes are quite distinct.

One of the possible clustering methods is competitive learning [4]. Given the training set of objects, competitive learning finds an artificial object (representative) most similar to the objects of a certain cluster.

A commonly used application of competitive learning is the Kohonen Self-Organizing Map [10], or SOM, described by Teuvo Kohonen in 1982.

SOM is inspired by the cortex of the human brain, where information is represented in structures of 2D or 3D grids. Formally, SOM is a type of artificial neural network [7] with two fully interconnected layers of neurons, the input layer and the output or Kohonen layer.

The first step of Kohonen learning is **competition**. Given the training vector on the network's input and weight vector for each neuron of the Kohonen layer, the neuron with the minimal (usually Euclidean) distance between weight and input vectors is excited or selected as the winner of the competition [4], [7].

The second step is **adaptation**. The neurons of the Kohonen layer are organized in a one-, two-, or three-dimensional lattice, reflecting its biological inspiration. A topological neighbour-affecting function is defined on the Kohonen layer, assigning a degree of participation in the learning process to the neurons neighbouring the winning neuron. In every learning step the weight vectors of the winning neuron and its neighbours are adjusted to move closer to the input training vector.

In the batch version of the SOM algorithm [18], equivalent to Lloyd's vector quantization [13], the winning neuron weights are not adapted immediately after the competition step. When all the training set is consumed, the weight vector of the output neuron N_i, $i = 1, \ldots, n$ is replaced by the weighted mean value of the training cases assigned to the clusters represented by the neuron N_i and its neighbours, using the neighbour-affecting function as the weight function for the mean calculation.

The trained network finally sets its weights in such a way that the topologically near neurons represent similar training cases while distant ones reflect different cases. This is analogous with the cortex of the human brain, where similar knowledge is represented by adjacent parts of the cortex. The topology of a trained SOM forms an inherently useful base for clustering.

To get a satisfactory approximation of a data set with higher variance, the number of neurons in the static SOM exceeds the number of potential clusters. Agglomerative clustering [20] is therefore used over the trained SOM.

Initially, each of the SOM neurons represents a separate cluster. In each iteration a distance function is computed for every couple of clusters and those with the shortest distance are merged together to form a new cluster. The iteration process stops when the specified number of clusters is reached. Examples of distance functions being used with SOM are the overall variance of the map [20], the Ward and the SOM-Ward distance [18].

In our work, the SOM algorithm with the clustering extension performs unsupervised cluster analysis over the training set. If there is an anomaly situation described in the training set of records, we suppose that this anomaly would be detected by some subset of records which would be revealed in the form of an isolated cluster.

In our case, SOM realises the transformation of the relations of the objects from the m-dimensional input space into the two-dimensional map of nodes (neurons) of the resulting Kohonen network. The complexity of the input space is reduced significantly and, in conjunction with colouring the nodes of the resulting network, data clusters can be effectively visualised.

Thus the SOM-based method could satisfy both conditions of the goal stated in the beginning of this paper.

4. Characteristics of the exploited SW

One of the most advanced applications of the SOM method currently available, Viscovery® SOMine 5.0., was used in the practical part of this work [18]. SOMine uses the batch version of the SOM algorithm exploiting the Gaussian neighbour-affecting function and the short cut winner heuristic search. The short cut winner search is based on the assumption that the winner for the given input vector in the current iteration is close to the winner for the same input vector in the previous iteration. Competition starts from the previous iterations' winner and its neighbourhood. If there is a better match within the neighbourhood, the new winner is registered and the competition continues in its neighbourhood. Otherwise the competition stops without searching the rest of the network. This heuristic significantly speeds up the competition phase and the algorithm itself, with a certain risk of neglecting clusters formed by small sets of data.

SOMine also performs agglomerative hierarchical clustering with the Ward and SOM-Ward distance function. The SOM-Ward function, used for experiments, computes the Ward distance for adjacent clusters only, omitting pairs of non-adjacent clusters, resulting in faster clustering process.

Numeric attributes are normalized by SOMine into the ‹0, 1› interval to eliminate differences in magnitudes of the input data attributes.

Categorical attributes have to be transformed into numeric values to be usable in the SOM algorithm. SOMine applies the following method [18]:

Let $a_1, \ldots, a_k$ be values of the categorical attribute x. Attribute x is replaced by k new attributes $x_1, \ldots, x_k$, set to:

$x_i = 1$ in case that x has value a_i,

$x_i = 0$ otherwise.

Obviously, this kind of a transformation raises the time complexity of the algorithm. It has also another substantial disadvantage. In Euclidean distance between two vectors with numeric and categorical attributes the difference in the categorical attribute adds 1 to the result while the difference in the normalized numeric attribute adds only fractions far less than 1. Euclidean metric can then evaluate as distant the vectors having similar numeric attributes but different categorical attribute and the vectors matching in categorical attributes as close, even if they may differ substantially in the numeric attributes.

Proposition 1: Described transformation of categorical attributes confuses competition phase and deteriorates results of the SOM learning process.

A SOM learning module and a corresponding SOM analytical tool, called EDAS, have been developed at the Technical University of Ostrava [1]. Authors adapted these tools according to the emergency call data characteristics and enhanced the EDAS explorer by the capability of displaying nodes according to the values of the selected attribute.

EDAS exploits a classical SOM learning algorithm with the neighbourhood radius initially set to 50% of the largest SOM dimension and the exponential radius shrinking. Learning times are thus 3 to 10 times longer then those seen with Viscovery®, but still in the order of minutes. On the contrary, EDAS was able to reveal more tiny anomalies than Viscovery®, validating thus the assumption regarding the potential drawback of the short cut winner search.

Comparing to the Viscovery® SOMine, EDAS does not have the capability to build clusters automatically from the trained SOM. Instead, it generates a SOM density map, which shows for every node the mean distance to its neighbouring nodes. The lines formed by nodes having relatively distant neighbours can be in fact considered the cluster borders.

5. Anomalous situation detection

Incident records from the period Feb. 1st – March 31st 2008 were used for the experiments. On March 1st 2008 the territory of the Czech Republic was affected by Hurricane Emma.

The input data set consists of about 25 000 records. The SOM-based procedure should find records related to Hurricane Emma (Emma-records), visualising the Emma-cluster formed by Emma-records in an automatic and effective way.

Emma records are characterized by:

- specific types in the incident classification (storm, danger status removal, obstacle removal);
- higher frequency of incidents, namely of the above-stated types, in the time period in question;
- higher frequency of incidents in certain regions (districts).

The training vectors presented to the SOM algorithm consist of the attributes “time of the incident beginning”, “incident classification”, “region of origin of the incident” and artificial attributes derived from these primary ones.

5.1.1. Experiment 1

We are starting with original attributes:

CALL_START (Date time) containing date and time of the call beginning, *CLASSIFICATION*, a categorical attribute describing the incident type and *DISTRICT*, again a categorical attribute storing territorial district where the incident happened.

For the result, see Figure 1. Every picture shows a trained SOM map of 20x20 nodes, depicted as small hexagons, projected onto values of the selected attribute or results of a mathematical expression. The nodes are coloured according to the mean values of the presented attribute in all the records assigned to the specific node. Scales are shown at the bottom of every picture. The black line inside a map defines a cluster of nodes with related attribute values.

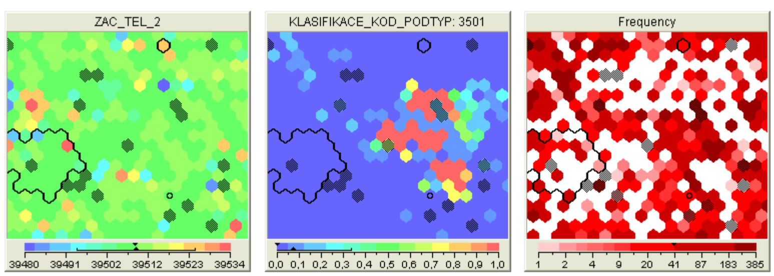

Figure 1. Experiment 1 results

The first and second maps show the CALL_START and CLASSIFICATION attributes, the rightmost map shows the frequency of records assigned to the nodes of the trained SOM.

The darker the node colours in the frequency map, the more records were classified to the node. Quite a lot of empty white nodes suggest a low quality of learning. Shadowed nodes contain Emma-records together with other records not belonging to Emma.

In compliance with Proposition 1, two categorical attributes used in this experiment make automatic Emma-cluster selection impossible.

5.1.2. Experiment 2

We are looking for a transformation of the categorical attributes into real numbers, which would preserve distribution characteristics of the original attribute values. We introduced frequency characteristics in a time quantum, the hour of the incident origin, as the basis for transformation.

The categorical attributes *CLASSIFICATION* and *DISTRICT* are used as the source for artificial numeric attributes produced by a function of relevance, combining the classification or district frequency within the time quantum of origin of the current incident and the occurrence of the same classification or district in the remaining quanta (adopted from the text classification TF-IDF weighing model [17]):

$$IMP\_CL = FREQ\_CL\_H * LOG\ (N_h / CL\_HRS)$$
$$IMP\_DS = FREQ\_DS\_H * LOG\ (N_h / DS\_HRS)$$

where N_h means the number of hours in the period analysed, *FREQ_CL_H* is the frequency of the current incident classification within the subset of incidents related to the hour of origin of the current incident, *FREQ_DS_H* denotes the frequency of the current incident district within the subset of incidents related to the hour of origin of the current incident, and *CL_HRS* or *DS_HRS* denote the number of hours in which the classification or district emerged.

Another attribute that is transformed is the district relevance:

$$IMP\_CL\_DS = FREQ\_CL\_DS * LOG\ (N_d / CL\_DS)$$

where N_d means the number of districts in the Czech Republic, *FREQ_CL_DS* is the frequency of the current incident classification in the current incident district during the whole period analysed, and *CL_DS* denote the number of districts where the current incident classification occurred.

To get an optimal result, the width of the neighbourhood had to be raised, increasing the influence of neighbouring neurons in the learning process.

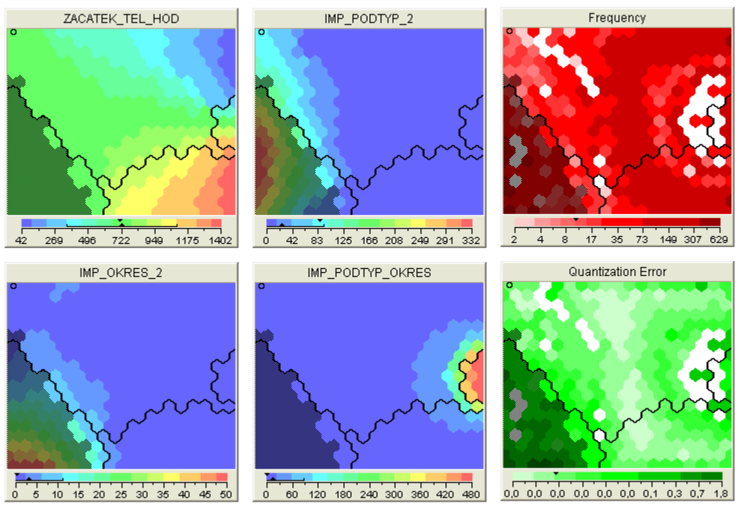

Figure 2. Experiment 2 results

The final output is in Figure 2. The learning quality within the Emma-cluster is satisfying as only 4 nodes are empty in the frequency map and the quantization error, defined as the average of the squared distance of all the data records associated with a node [18], depicted in the lowest right map has a negligible 1,8 compared to the standard deviation of 70 of the *IMP_CL_DS* attribute.

Moreover, an unexpected anomaly was identified by the smallest cluster on the right side of the map, collecting incidents assisted by the Prague Municipal Police and described by the classification which is not used anywhere except of Prague.

5.1.3. Experiment 3

The EDAS tool set [1], enhanced by authors to address the needs of this work, has been exploited in this experiment.

The training set and attributes from the previous experiment were used.

Figure 3 shows the final output of the SOM network of 40 x 40 nodes after 100 training cycles. Shadowed nodes contain Emma-records, the Emma-cluster is well bordered by the wavy line in the bottom left-hand corner of the SOM density map (bottom middle). The rainbow like stripes within the Emma-cluster, visible in the *IMP_DS* map (top right) point to the movement of Emma across the territory of particular districts.

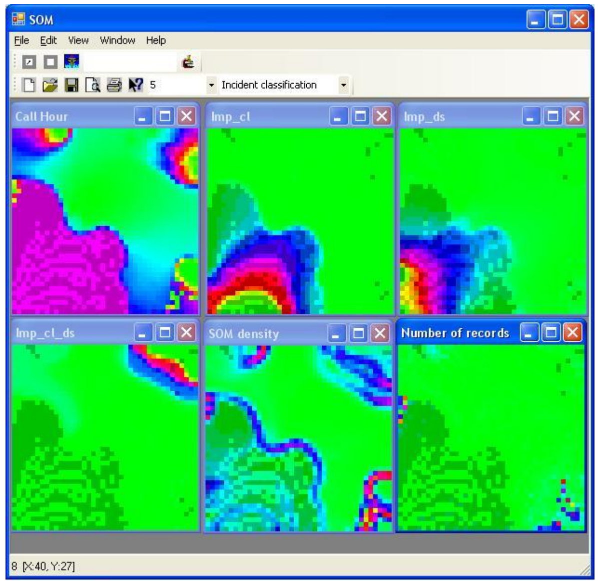

Figure 3. Emma-EDAS output

Figure 4, showing the output of the analysis of the nodes in one of the stripes, states that on March 1st, from 11 to 12 a.m., 90 Emma-related incidents of types 3331 (windstorm), 3501 (removing dangerous objects) and 3526 (removing obstacles) were reported from two districts in the central part of the Czech Republic.

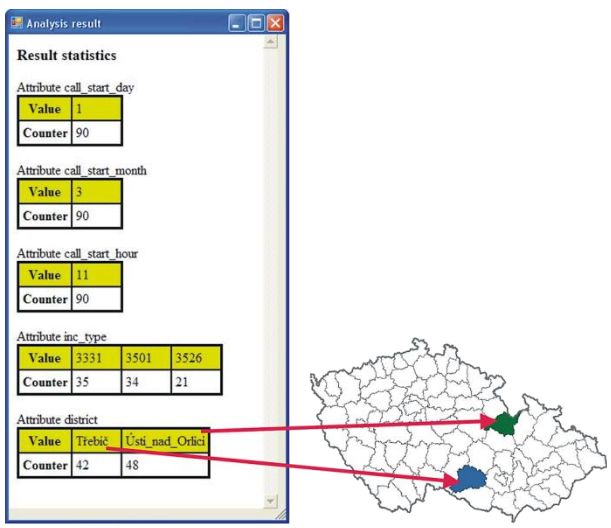

Figure 4. Emma-EDAS analysis

The anomalies not related to Emma are concentrated in the upper right-hand corner of the *IMP_CL_DS* map (Figure 3, bottom left). Except of the one described in the Experiment 2, this experiment revealed higher rate of car thefts in Prague and regular system tests being performed at the time of operators shifting at the particular PSAP.

6. PSAP Technology performance analysis

The performance monitoring relies on attributes describing the architecture and technology in the system background, rather than the nature of the incident as seen in the previous experiment.

In a normal situation emergency calls are processed by operators in the region where the related incident originated, e.g. in region A. Once the call is processed by an operator-agent of the PSAP, the incident report is stored in the database, which is a remote service to most of the centres, accessible through a Wide Area Network (WAN). A delay in the operation of the database may suggest a network overload or a DB service capacity problem. In parallel with the incident report being inserted into the DB, it is passed to the Emergency Response Agencies (ERA) via the respective proxy service. The proxy services, after passing the incident report to the ERA, expect a confirmation message. Delays in receiving this confirmation may give evidence of the ERA systems being overloaded. In the telephone subsystem we can measure the ringing time, i.e. the time interval between the moment when the call arrives at the agent and the moment when the operator picked up their phone. As picking up the phone is automatic (auto-answer mode is preset in the whole system), stretching of the ringing interval may suggest congestion in the telephone system. Moreover, two values characterising the operator's working style are measured, the length of the call and the time interval between the start of the call and the insertion of the incident report into the DB.

In critical situations, when many incidents happen in a short period of time, or many people are announcing the same incident and the operators of PSAP A are busy, calls are routed to and processed at the PSAP of another region (say B). In the course of processing the calls the DB and telephone call measures have the same meaning as in the previous case. But after the call is processed by PSAP B, the reports to the ERA are communicated via the WAN to the ECC of the original region, A. The further incident report flow is identical to the previous case. The time it takes to transfer the incident report between PSAP B and the proxy services of PSAP A is measured and evaluated.

For the performance analysis we used data from a special testing of the emergency call-taking system under peak load. In an hour the system in its full capacity (14 PSAP, 70 operators) received 3400 emergency calls and produced around 10,000 incident reports. As there was only one incident type reported during the tests (the "system test"), it made no sense to take the incident classification into account.

In the test of the emergency call taking system the aim was to show the system with a heavy load in unrealistic conditions in order to reveal its hidden flaws. Figure 5 shows the anomalies shown in the system technology.

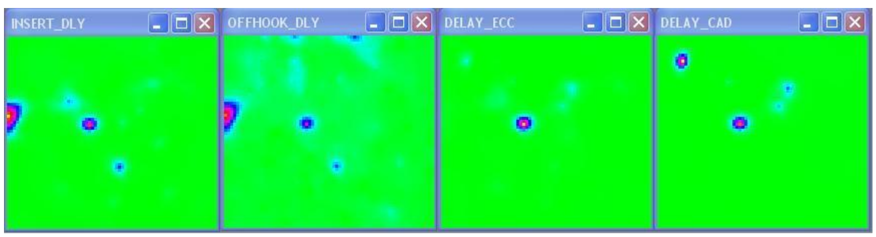

Figure 5. Anomalies in the PSAP technology under peak load

Maps on the Figure 5 display delays in the telephone subsystem (OFFHOOK_DLY), database operations (INSERT_DLY), incident transfer in the emergency centres' technology (DELAY_ECC) and in the responding agencies' technology (DELAY_CAD). Interesting values reported by the EDAS analytical module for the maps in Figure 5 are shown in Table 3.

Table 1. An example of the records related to the highlighted nodes from Figure 5. Time intervals characterising delays are in seconds.

ID	ID_ECC_ REC	RINGING_ TIME	INSERT_ DLY	OFFHOOK_ DLY	DELAY_ECC	DELAY_CAD	ID_ECC_ RESP
1	30	20:03	3	1	1	1	40
2	110	20:05	199	15	8	1	40
3	110	20:05	199	15	8	4	40
4	80	20:11	26	1	128	219	20
5	20	20:12	146	5	500	500	20
6	20	20:12	157	2	500	500	20
7	20	20:13	214	19	500	500	20
8	90	20:13	1	1	228	500	20

Row 1 stands for an example of the normal situation, when emergency centre '30' received an incident for centre '40'. The database response is 3 seconds, which is within acceptable limits. The delay in the telephone subsystem is 1 second, as is the incident transfer time in the emergency centres' network and in the emergency response agencies' technology.

Rows 2 to 8 represent records mined out from the red and blue spots of the four maps in Fig. 5. Rows 2 and 3 point to problems with the database and telephone subsystem for centre '110'. The communication with the target centre '40' and its related emergency agencies is alive, even if the slightly higher values may suggest incoming problems. On the contrary, row 4 shows that centre '80', when receiving for centre '20', encounters long database responses, as well as significant delays in communicating the incident report to the target centre. A minute later, records 5-7 show deadlock at centre '20'. With unacceptable database responses the centre cannot even communicate with the other centres and with its own emergency agencies (here the value 500 stands for unknown information). The last row, row 8, shows that another centre, centre '90', handles incidents properly, but cannot propagate incident reports to target centre '20' and its emergency response agencies, definitely because of the problems of centre '20' reported above.

7. Conclusion

In this work we devised and experimentally proved a method for unsupervised knowledge discovery in the emergency call data. Experiments were focused on revealing clusters of emergency records which would point to abnormal situation in time and place.

Method is based on the Kohonen Self Organizing Map (SOM) algorithm. For the SOM creation and visualization we used commercially available SW Viscovery® SOMine 5.0 and the tool set EDAS, developed by authors at the Technical University of Ostrava.

We proposed transformations of categorical attributes, the discrete values of which are not suitable for the SOM algorithm, into the real numbers domain and showed that after this transformation the SOM is generally able to detect anomalies in the emergency data. The transformations are built on the frequencies of incidents in time and place, expressing the relevance of the incident with respect to the time and place of origin of the incident. The learning proceeds on the values of the transformed attributes, which are nevertheless bound to the original records. In this context, the values of the original understandable attributes could be used for explaining the result, as well as for further processing.

Traditional anomaly detection methods [5], [16] and [21] use SOM for modelling the anomaly-free space from a set of data approved as correct (bearing signs of supervised learning) and then classify a new case as anomalous if it falls outside the modelled non-anomalous space. As hardly any emergency situation can be considered anomalous, we could not use the two-class classifier approach with supervised learning. Therefore we searched for certain patterns in the data. After the patterns were recognized by unsupervised learning, the composition was always presented to a person for them to analyze the situation.

This concept can be enhanced in such a way that, provided that the algorithm is run periodically, the new composition would be further processed only if the composition in the current run is different from the composition in the previous run. SOM quality measures [14], [9] could help to achieve good results here.

To obtain the best from the SOM algorithm, we designed transformations of categorical attributes to get vector space over the real numbers. Further research could deal with a variable time window for selecting records of the training set. It is expected that an optimal interval could exist in which transformations based on the frequency characteristics would work correctly, while in too long or short intervals extreme values could flatten or disappear.

We expect that the monitoring of the performance of the emergency call-taking system and analysis technique presented in the Section (6) of this paper can be effectively combined with a model of resilient incident report transfer between call centres inspired by [6] and [11]. Such a model could be created and the SOM + EDAS analytical tools used for its output monitoring and performance analysis, provided that detailed information regarding the underlying technology of the PSAPs network (the WAN topology, technology and parameters) were available.

The SOM algorithm consumed the training set of records consisting of transformed attributes of the original incident records, identified subsets of records belonging to the real emergency case, hurricane Emma, or characterising the anomalies in the PSAPs technology in the performance analysis experiment, and built an isolated cluster over these records automatically. The method devised here proved its ability to discover anomalies hidden in data, visualizing them in an effective user-friendly manner and indicating a way towards the further development of the intelligent management of the emergency call information system.

References

[1] Baca, R., Kratky, M., and Snasel, V.: Pover Outage Data Analysis via SOM Neural Networks. In Proceedings of ELNET 2008, Ostrava, 2008, pp. 87-98.

[2] Brockett, P.L., Xia, X., Derrig, R.A..: Using Kohonen's Self-organizing Feature Map to Uncover Automobile Bodily Injury Claims Fraud, Journal of Risk and Insurance 65(2), 1998, pp. 245-274.

[3] Fire and Rescue Service statistical yearbook, http://www.hzscr.cz/clanek/statistical-yearbook-2007-312484.aspx (cited 22nd March 2009).

[4] Gan, G., Ma, Ch., and Wu, J.: Data Clustering: Theory, Algorithms and Applications, SIAM, Philadelphia, 2007.

[5] Gonzalez, F., and Dasgupta, D.: Neuro-Immune and Self-Organizing Map Approaches to Anomaly Detection: A

Comparison, Proceedings of the 1st International Conference on Artificial Immune Systems (ICARIS), Canterbury, UK, Sept. 2002, pp. 203-211.

[6] Grosan, C., Abraham, A., and Hassanien, A.: Designing Resilient Networks Using Multicriteria Metaheuristics, Telecommunication Systems: Modelling, Analysis, Design and Management, Springer Verlag, Germany. Aug, 2008.

[7] Haykin, S.: Neural Networks: A Comprehensive Foundation, 2nd edition. Upper Saddle River, NJ: Prentice-Hall, 1999.

[8] ISCRAM, Information Systems for Crisis Response and Management: http://www.efficient-response.com/iscram2008/ISCRAM2008_CFP130.pdf (cited 22nd March 2009).

[9] Kaski, S. and Lagus, K.: Comparing Self-Organizing Maps. In Proceedings of ICANN96, International Conference on Artificial Neural Networks, Springer, Berlin 1996, pp. 809-814.

[10] Kohonen, T.: Self-Organizing Maps. Springer-Verlag, Berlin, 1995.

[11] Konak, A., and Bartolacci, M. R. (2007). Designing survivable resilient networks: a stochastic hybrid genetic algorithm approach. Omega, Special Issue on Telecommunications Applications, Volume 35, Issue 6, December 2007, pp. 645-658.

[12] Lichodzijewski, P., Zincir-Heywood, A.N., and Heywood, M.I.: Dynamic Intrusion Detection Using Self-Organizing Maps. In The 14th Annual Canadian Information Technology Security Symposium (CITSS), 2002.

[13] Lloyd, S., P.: Least Squares Quantization in PCM. IEEE Transactions on Information Theory, vol. 28, no. 2, 1982, pp. 129-137.

[14] Mitra, S., Pal, S.K.: Self-organizing neural network as a fuzzy classifier. IEEE Trans. Systems,Man, Cybernetics 24 (3), 1994, pp. 385–399.

[15] NENA, National Emergency Number Association: http://www.nena.org/911-statistics (cited 20th December 2008).

[16] Ramadas, M., Ostermann, S., and Tjaden, B.: Detecting Anomalous Network Traffic with Self-Organizing Maps. Proc. of Recent Advances in Intrusion Detection, 2003, pp. 36-54.

[17] Salton, G. & Buckley, C.: Term-weighting approaches in automatic text retrieval. Information Processing and Management 24(5), 1988, pp. 513–523.

[18] Viscovery®. SOMine 5.0. Copyright © 1998-2007 by Viscovery Software GmbH, http://www.viscovery.net (cited 7th November 2008).

[19] Vokorokos, L., Balaz, A., and Chovanec, M.: Intrusion Detection System Using Self-Organizing Maps. Acta Electrotechnica et Informatica, No.1, Vol.6, TU Kosice, 2006.

[20] Yang, M. Y.: Extending the Kohonen self-organizing map networks for clustering analysis. In Computational Statistics & Data Analysis 38, 2001, pp. 161–180.

[21] Ypma, A., and Duin, R.P.W.: Novelty detection using self-organizing maps. Progress in Connectionist Based Information Systems, Vol. 2, Springer, 1998, pp. 1322-1325.