Prediction of Gas Consumption

Jiří Klema, Jan Kout

The Gerstner Laboratory
For Intelligent Decision Making and Control
Faculty of Electrical Engineering
Department of Cybernetics
Technická 2, Prague 6, 166 27
e_mail: {klema, kout}@labe.felk.cvut.cz

Keywords: prediction systems, gas consumption, artificial intelligence, case-based reasoning, neural networks, regression techniques

***Abstract**: This paper deals a universal prediction system based on AI techniques and classical statistical analysis. It describes principles and methodologies used within the system. Further it discusses internal structure of the system and general conditions for its successful use. Despite of system wide applicability, originally it was conceived as a system for prediction of gas consumption. A practically oriented part of the paper reports on the gas consumption problem domain and an application of the system to the domain.*

1. Introduction

Intelligence of agent's behaviour is often characterised by its ability to learn, i.e. to base its decisions on its former experience. That is why study of learning is considered to be one of the basic topics within the field of artificial intelligence. There is a wide range of ways how to utilise former experience - the simplest is a trivial memorisation of past cases, on the other hand past cases can help to create as complex new theories as the Newton's laws when deep reflection is applied to them. A number of various approaches have been suggested and studied to achieve at least part of this diversity. Their comprehensive review is offered in most AI textbooks, e.g. [RuNo95].

The presented task of gas take-off prediction belongs to the type of problems referred to as numerical sequence prediction. Another problem of a similar type is exchange rate or stock prices prediction - both are of vital interest to financial institutions and they have been subjected to many AI studies. An important characteristic of this sort of problems is that the outcome (day take-off e.g.) is influenced by a number of different factors some of them may not be even explicitly mentioned in the problem description. There is no clear functional dependence between the past and present value of the member of the considered sequence. The upper mentioned studies helped to identify those AI learning methods which seem to be best suited to this type of problems - case-based-reasoning (CBR) and neural nets (NN) definitely belong to them. The present study will apply both these approaches as well as a classical statistical analysis. At the same time it shows that a combination of classical and AI methods can acquit itself well in industrial use. The results achieved by all these methods will be compared and evaluated.

The following text summarises all the methods used for prediction of gas consumption altogether with a notion of the system as an ensemble. Emphasis is put on description of aggregate system behaviour, i.e. correctness of prediction and its evaluation, system robustness, time model persistence and conjunct necessity of model adaptation. The final recapitulation generally defines requirements

that assure optimum prediction quality over approachable data. Last but not least, the paper alludes versatility of the system discussed considering possible variance between individual distributors as well as with respect to eventual system utilisation in other problem domains.

2. Problem description

Prediction method of gas consumption has been developed for local gas distribution companies. These companies take gas from a (supra)national gas provider and distribute it withinside the region they cover. Usually their contract with a gas provider is designed in the form of the day consumption limits. The local distributor should keep its take-off in between the contracted limits every day. Whenever the take-off exceeds the limits the company has to pay an indispensable penalty. In order to minimise the penalty charge the local distributor has to shift its engagement to its following.

Three types of gas consumers can be distinguished (with respect to the local distributor):

- Big industrial consumers - their contracts are designed similarly to the contract between the local distributor and gas provider.
- Other industrial consumers (workshops, garden-tillages, ...) - their gas charge is reduced as far as they are able to regulate their gas consumption (to switch to alternative energy source whenever they are asked by their distributor).
- Households - they take gas for a constant price without any possibility to regulate their consumption.

It follows that timely prediction of day consumption can remarkably help to the local distributor to keep in between the consumption limits. Firstly, it can evoke switch-over of some consumers to alternative energy sources. Secondly, it gives a chance to load or empty a local distributor's gas-holder that is very often used to smooth consumption irregularities.

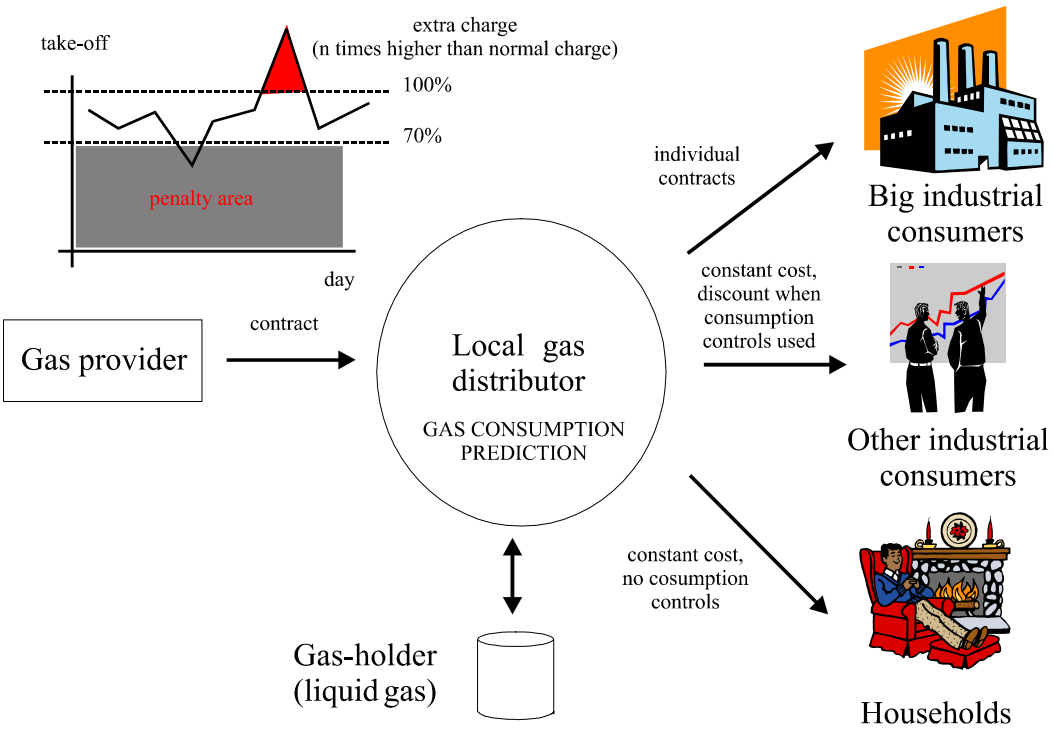


Figure 1 Gas distribution diagram

3. System structure and methodologies used

Logically the system can be divided into three independent parts (see Figure 2). The first part is instrumental towards holding past values of consumption and appropriate influencing factors. We call it *archive file*. Content of the archive file must evidently correspond to the observed distribution net and it cannot be transferred to other nets. The second part is as a matter of fact a mathematical algorithm. It consists of statistical and self-learning methods of time series prediction. This *prediction algorithm* is universal in its application. Provided preservation of the same archive file structure it can be directly put to be used in arbitrary domain. A domain model completes the system. This model consists of model parameters calculated during such a called training phase of the system development. The parameters are distance and correction weights. Each influencing factor is adherent to one distance and one correction weight. Their values provided by stochastic optimisation by genetic algorithms (GAs). The model can assure sufficient predictive power if and only if it is derived of sufficiently extensive data, i.e. the archive file has to contain satisfactory number of history records. On the other hand, validity of the appropriately derived weights survives during long time period. In other words, these parameters do not have to be recalculated often. Sufficient time self-adaptation of the model is brought to effect by archive file updating (new records are automatically added to archive file as they appear on the system input).

To summarize it up, the system behaviour is in every point determined by the algorithm, which is invariant, and by the data, which change during time. The content of the archive file and the adjusting of model parameters set down the data. Consequently, the system behaviour is concurrently defined by its static component (algorithm, parameters) and its dynamic component (the archive file content). This combination provides time robustness of the system (sporadic re-training) as well as its time adaptability (it can still respond to topical trends of predicted value).

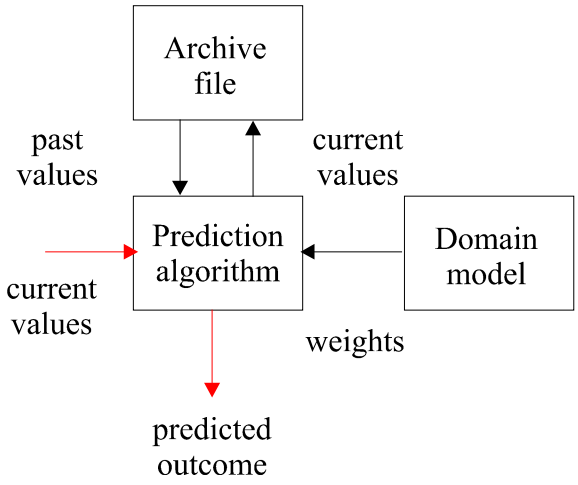


Figure 2 Logical system structure

3.1. Data pre-processing and analysis, archive file

The archive file is divided into basic units, training examples (records). Structure of these examples corresponds to the predicted value, influencing factors and length of the predicted period. The system generates 24 hours predictions of gas consumption, the training example corresponds to the same period length. Data pre-processing and data analysis altogether with archive file generation play a key

role in a predictive model building. The most important seems to be especially selection of appropriate influencing factors (attributes) and selection of proper training examples (or rejection of corrupted training examples). Unfortunately, this task seems to be hard to automate and it usually requires background knowledge. This knowledge can be combined with or even replaced by statistical tests revealing correlation among attributes and the predicted variable. Quality of the archive file strongly limits accuracy of the final prediction. Experiments show that this quality is much more important than the algorithmic method selected. It means, it is possible to select a better or worse algorithmic method but the prediction accuracy limit given by the archive file quality can never be exceeded.

3.2. Case-based reasoning, k-nearest neighbours method

Case-Based Reasoning (CBR) is a major paradigm in automated reasoning and machine learning. In case-base reasoning, a reasoner solves a new problem by noticing its similarity to one or several previously solved problems and by adapting their known solutions instead of working out a solution from the scratch [Man97]. CBR can mean different things depending on the intended use of the reasoning: adapt and combine old solutions to solve a new problem, explain new situations, critique of a new solution based on old classes, reason from precedents to understand a new situation, or build a consequent solution based on previous cases. However, these different aspects can be classified into two major types: interpretative CBR (trying to explain) and problem solving CBR (suggesting new solutions) [Kol96].

CBR suggests a model of reasoning that incorporates problem solving, understanding and learning and integrates all with memory processes. The quality of a case-based reasoner's reasoning depends on five things [Kol96]:

a) The experiences it has had.
b) Its ability to understand new situations in terms of those old experiences.
c) Its adeptness at adaptation.
d) Its adeptness at evaluation and repair.
e) Its ability to integrate new experiences into its memory appropriately.

In short, given a case to solve, case-based reasoning involves the following steps [Man97]:

a) Retrieving relevant cases from the case memory.
b) Selecting a set of the best (closest) cases.
c) Deriving a solution to the considered case.
d) Evaluating the solution (in order to make sure that poor solutions are not repeated).
e) Storing the newly solved case in the case memory.

The crucial part of the CBR system development is the retrieval and selection of cases since the remaining operations of adaptation and evaluation will succeed only if the past cases are the relevant (representative enough) ones. The retrieval of relevant cases depends mainly on selection of appropriate set of indices. The first task when memory of cases is created is to find relevant case features for indexing. Then, in learning feature importance, each feature is associated with a weight that is adjusted after each prediction attempt during the training process by comparing the current case with its most similar stored cases. Selection of appropriate distance metric shows important role, too.

The given task represents building of a consequent solution (prediction) based on previous cases. The task asks for a specific system that omits some of basic features of CBR and puts emphasis on its other features again. For instance, new solutions are never stored to the case memory, there are no big demands on memory organisation and indexing as the memory consists only of hundreds of

examples. On the contrary, it emphasises requirements on case similarity metric, finding optimum attribute weights and adaptation of old solutions (take-off values) to a new day. The gas consumption prediction problem deals only with numeric attributes. For this type of attributes important issues of CBR system design seems to be k-nearest neighbours method (kNN). Consequently, this method can be considered either as a kNN method enhanced for memory management or a CBR method putting emphasis on similarity metrics and weight parameters optimisation.

3.3. Non-linear local regression

Most statistically based data mining modules use linear regression analysis by using the popular least squares method. The least squares method calculates a straight line through a set of observed data. This method is successful and useful in many cases. The problem may occur when the data exhibit singular features. Singular data or data close to singular are potentially dangerous for computer processing because of the computer round-off arithmetic. By using the standard least square method, results might lead to quite non-stable solutions which might provide us with the inaccurate solutions. Since these problems may occur, the employment of SVD (Singular Value Decomposition) method is very effective. The SVD method is mainly useful and helpful for the cases where the ratio between the number of patterns and the number of attributes is low, e.g. potential appearance of singularity. The SVD method offers the optimal solution like the least square approach does. Even more its solution is as stable as possible. More detailed and complex information about regression methods can be found in [Rya97].

The results using the SVD method are bounded to linear combinations of input attributes. To improve the results, the non-linearity is added into the system model. The necessity of added non-linearity derives from the detailed analysis of the data. From the analysis concludes what attributes are the most suitable to enrich them with the non-linearity. The enriched system reaches the better and more precise solutions than the standard linear regression using the least square method. The quality and preciseness of the data analysis are the crucial part of the described method.

The locality of the regression approach is devoted to the pre-selection of training examples to build the regression model for the given pattern. It is similar to the least square method where the neighborhood area is defined. The area always covers the N most similar cases with respect to the given attribute (day). To sum up, there is no one common model for all attributes (days). Each attribute has its own unique model given by its characteristic to reflect its features and conditions.

3.4. Prediction based on neural networks

A neural network can be described as a processing device, either an algorithm, or actual hardware, whose design was motivated by the design and functioning of a human brain. The neural networks can be employed on the practical tasks from the field of artificial intelligence (AI). Most neural networks have some sort of training rule whereby the weights of connections are adjusted on the basis of presented patterns. In other words, neural networks learn from examples as children do and that is the reason why neural networks are interesting for quite a lot of very dissimilar people working on different problems and subjects.

The neural networks can be described by a mathematical model, which simplifies the functionality of neurons in the human brain. A formal neuron is the fundamental part of neural networks. It consists of N inputs that model dendrites. Inputs are weighted by real numbers that model their permeability. The weights can be even negative to model their inhibitory character. A formal neuron has one output that is calculated as the weighted sum of inputs transformed by the activation function. The activation function gives the neural networks their nonlinear character, which is very important as it is described later.

The *Topology* and *learning algorithm* are two basic features of neural networks. Only the basic parts used by the prediction system are mentioned. More detailed information about the neural networks can be found for example in [Sou91].

The neural networks consist of set of formal neurons that can be connected in many ways. The number of neurons and their connection defines the *topology* (architecture) of the network. The neural networks can be also seen as a black box with their inputs and outputs. That is why the input, output and hidden neurons are defined. The only topology that is considered later is a special case of feed-forward network called layered feed-forward network. It consists of layers of nodes, which are interconnected with one layer above (apart from output layer) and with one layer bellow (apart from input layer). The layered feed-forward neural networks have usually one input and output layer and one or two hidden layers (can be seen on the figure 3).

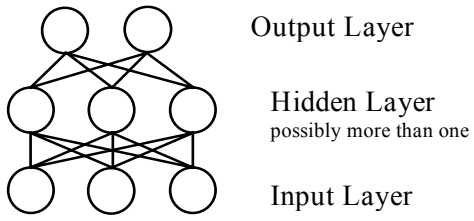


Figure 3 – Layered Feed-Forward Neural Network Topology

There are also many *learning methods* developed for neural networks by now. They can be divided into two main parts supervised learning and unsupervised learning. In supervised learning, there is a teacher who tells the net how well it performs and the network correspondingly adjust its parameters according the given algorithm. In this case, the neural network is taught from well-known examples (teacher). In case of unsupervised learning the net is autonomous. The network perceives the presented data and tries to find out some of the properties of the data set and learns to reflect these properties in its output. All the knowledge used during the learning phase in encoded in the network model and the learning method.

The data described in section 3.1. have been treated by a layered feed-forward network using the most successful and popular supervised learning algorithm - backpropagation algorithm. There are many modifications of the backpropagation algorithm and the used one is named delta rule backpropagation algorithm. In general the algorithm consists of two stages: forward and backward propagation. During the first stage (forward propagation), the network uses the current parameters (at the beginning settled randomly) of the network and computes the actual output vector. Backpropagation stage computes the differences between the desired and actual (computed) vectors. The differences are propagated back through all the layers down and all the parameters of all the nodes and weights are correspondingly adjusted. The described method is repeated for all the patterns until the requested correctness is achieved. More detailed information and description of the practical implementation of layered feed-forward networks using the backpropagation algorithm can be found in [Kou96].

The backpropagation algorithm suffers from a major drawback. The convergence of the algorithm is very sensitive to the initial set of weights. Very often, it is found that the algorithm fails to converge. This phenomenon is usually explained as the algorithm's "getting stuck at a local minimum" of the error surface. Even when it does converge, it is training examples very slowly and the mean square error decreases only after a great number of steps. Another problem of the backpropagation algorithm is the inconsistency of training. The mean square error might remain the same for many steps and then suddenly plunge to a very low value (for example solving of the parity problems suffers from

this feature).

The other problem is to find the optimal topology of the employed network and the most suitable learning parameters which would lead to the solution as fast as possible. One or two hidden layers are typically used. The number of input attributes gives the number of input neurons. The number of desired outputs defines the number of output neurons. It is necessary to set the number of neurons in the hidden layer(s). This problem is connected to the problem of generalization of neural network. The architecture of the network should correspond to the complexity of the problem otherwise the over-training problem can occur.

For the purpose of neural network module, the data should be scaled. The employed topology is closely connected to the training data. One hidden layer seems to be enough. The number of input neurons depends on the number of input attributes given by data analysis. There is one output neuron representing the gas take-off. The number of neurons in the hidden layer is set up experimentally. There are several heuristics, but none of them is general enough to fit in every case. That is why it is necessary to experiment with the data to achieve as convenient result as possible.

3.5. Combination of the methods

The problem of numerical prediction is the kind of problem for which the combination of several independent methods is effective to calculate one final result. If we compare the numerical prediction problem to the classification system, where there are several subjunctive classes (A,B,C,…), then it is essential that it is possible to employ more easily mathematical tools on numerical output values to combine results of several methods. Let us suppose that there are three independent prediction modules (CBR, LNR-SVD, NN), then the easiest output result can be computed as an average value from all results provided by the individual modules. In case of classification tasks, it is much more complicated to combine these methods. If one module classifies the given example into class A and the second one in class B, then it is not possible to combine the results without any apriory information. Apriory information can be for example the measure of believe of individual modules, an existence of meta-level module and so on. Combination of more numerical outputs has one more advantage. The advantage is in the fact that computing an average value can eliminate the errors of different signs from different modules. To sum up, the combination of more numerical methods brings two effects. In case of different error signs, the error is minimized (eliminated). In case of same signs, the combination eliminates the higher errors of a particular method.

From the statistical point of view, the errors can be divided into two categories: data inaccuracy and method inaccuracy. The first type of inaccuracy can be caused by input values that are not characteristic for the given situation, for example steep increase or steep decrees of one value caused by unexpected conditions (circumstances). Wrong inputs can be also caused by inserting incorrect input values (kind of human/input system mistake). These kinds of inaccuracy have always an impact on the result no matter what method is employed. The latter type of inaccuracy is caused by the used method and the aim of every system is to minimize it. The possible approach that minimizes the inaccuracy is the combination of the methods. It is possible to use the comparative local advantage of an individual method. From the statistical analysis of achieved results on the training data, an individual method might be more successful under the special and well defined conditions (for example concrete months, extreme temperatures, afternoon hours etc.). If it is so, the successful method has higher priority when computing the result under the special conditions. The described method is quite challenging and offers large space for experiments. On the other hand it is extremely demanding on the amount of training data. In the future, the effort will be mainly focused on optimisation of combination of particular methods.

4. Achieved results and their quality

The criteria that describe the quality of the prediction are a fundamental part of the final solution from the beginning. The criteria are important information during the learning phase, data analysis and model construction. They are used as feedback that measures the current quality of the solution and shows the way of model optimization. The final model is tuned to meet the criteria (reinforcement learning) as successfully as possible.

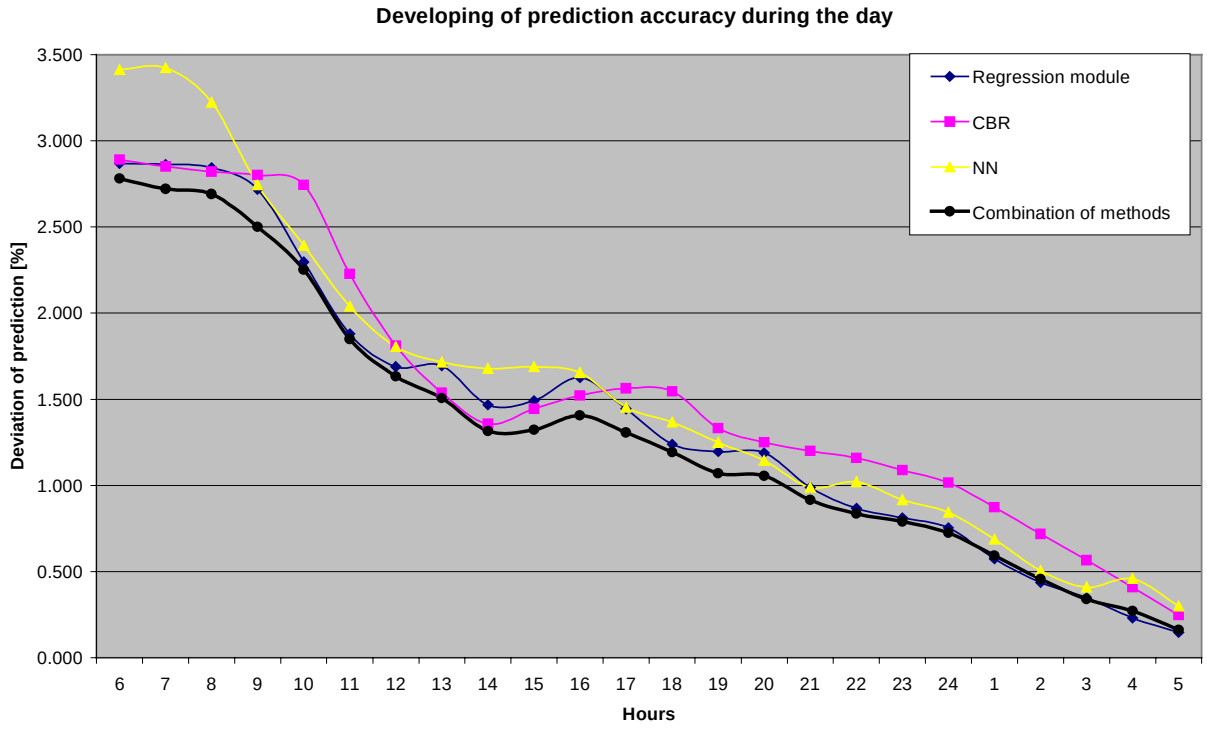
The gas take-off system is developed with respect to three main criteria:

- mean absolute percentage deviation of gas take-off predictions computed at the concrete hour in all given days,
- standard deviation of gas take-off predictions computed at the concrete hour in all given days and
- percentage correctness of the prediction with respect to the given level (for example 5%).

The criteria can be very critical at one given hour during a day in order to control the gas consumption (for example changing to alternative source of power). An another criterion can be an absolute value of maximal deviation.

These criteria are important not only during the developing of the prediction system, but they are also important to compare individual methods or to compare the competing systems. It is necessary to define the same criteria for every comparison. It is even necessary to provide the systems (methods) with the same training and testing data sets, otherwise the comparison is not accurate as well as meaningful.

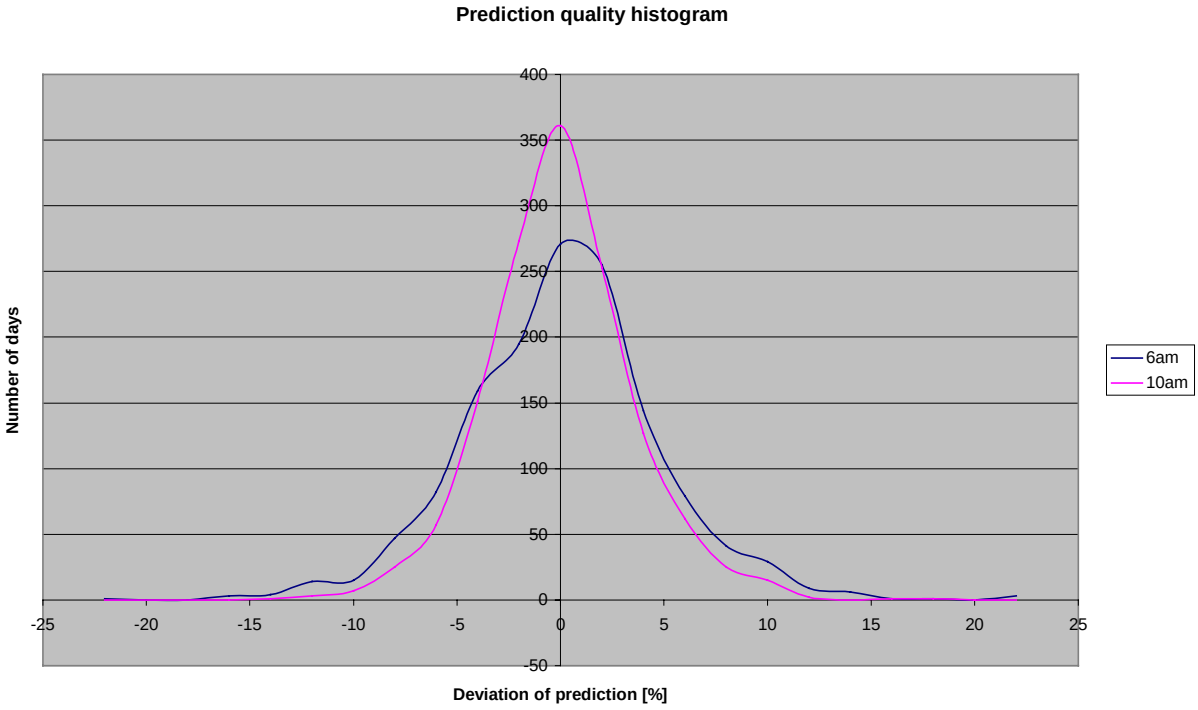
The two possible ways of graphical representation of results are show on Graph 1 and Graph 2. The



Graph 1

Graph 1 demonstrates the developing of the prediction during the gas day hour by hour. From the graph, the advantage of the combination of methods can be seen as well. The result computed as the combination of particular methods is better than results achieved by all individual methods. From the customer point of view, as the most important hours seem to be the hours between 9-11 a.m. when the customer can evoke actions in order to keep the gas take-off between the consumption limits.

The Graph 2 is the frequency graph of the errors categorized by their values. The deviations have Gauss distribution and the progress between 6 a.m. and 10 a.m. can be seen as well. From this type of graph, it is possible to see the number of days with relatively high error. The number of days with relatively high error is very important for the quality of the prediction.



Graph 2

5. Conclusion

The described system is able to successfully predict the gas take-off in local gas distribution networks. It was developed for German local distribution companies in cooperation with TeleDataElectronics company. Its functionality was proven on several different nets, the system always fulfil the requirements of the customer and was used to control the gas consumption. Currently it is used by two local distributors and it is installing on three other gas nets. The system combines the techniques of machine learning (Cased Based Reasoning and Neural Networks) and regress extrapolation. The system has its own static and dynamic part to guarantee the robustness and time stability, e.g. the system does not require the new training when character of the gas take-off is more less stable. Model changes are processed continuously with the changes of dynamic part of the system – achieve file. The robustness of the gas take-off prediction system is also supported by the combination of three different methods.

Mean absolute percentage error of prediction is about 4%. The preciseness of the prediction depends on the data quality, data amount (history) and on deterministic characteristic of the net behavior.

Generally speaking, the prediction system can be considered as an abstraction and a simplified model of the distribution net. The system can be included into more complex system that monitors and controls the distribution net. The prediction system can be part of decision support module that help the local distributor to keep the gas take-off between the consumption limits. As it was already mentioned, the customer can evoke switch-over of some consumers to alternative energy sources. If it is possible, it gives a chance to load or empty a local distributor's gas-holder that is very often used to smooth consumption irregularities. These special conditions are different customer by customer and should be taken into account in order to use all possibilities how to successfully meet the customers' requirements.

The presented task of gas consumption prediction belongs to the type of problems referred to as numerical sequence prediction. The important characteristic of the described system is its universality. After slight modification of the model structure and the data structure, the system can to predict arbitrary numerical magnitude (electricity power take-off, water take-off, traffic, exchange rates, stock prices and so on). These problems are of vital interest to financial institutions and they have been subjected to many AI (Artificial Intelligence) studies. Due to the easy modification of the prediction system to predict different numerical magnitude, the system can be seen as open. Generally it can be said that prediction correctness does not depend on type of predicted magnitude but on amount, variability, credibility and relevance of the data used for model generation. In other words, it depends on quality and length of period of the predicted system behaviour monitoring. From another point of view, the outcome (e.g. day gas consumption) is influenced by a number of different factors and some of them may not be even mentioned in the problem description.

The largest space for improvements seems to be in the combination and smarter integration of particular methods. This phase of studies and experiments might lead to their more sophisticated combination. The most promising seems to be application of local accuracy estimates [Woo97].

6. References

[Kol96] Kolodner, J.: Case-Based Reasoning, Morgan Kaufmann Publishers, 1996

[Kou96] Kout, J.: Neural Network Agent. Technical Report of The Gersner Lab CTU Prague & FAW, GL-28/97, Linz - Hagenberg - Praha - Wien, 1997

[Man97] Mántaras, R., L.: Case-Based Reasoning: an overview, AI communications vol. 10, March 1997, p. 21-30

[RuNo95] Russell, S., Norvig, P.: Artificial Intelligence. A Modern Approach. Prentice-Hall, Inc., London 1995, ISBN 0-13-103805-2

[Rya97] Ryan, T. P.: Modern Regression Methods, A Wiley-Interscience Publication, New York - Chichester - Brisbane - Toronto - Singapore, 1997

[Sou91] Souček, B., and The IRIS Group, Neural and Intelligent System Integration, A Wiley-Interscience Publication, New York - Chichester - Brisbane - Toronto - Singapore, 1991

[Woo97] Woods, K., Kegelmeyer, W., P., Bowyer, K.: Combination of Multiple Classifiers Using Local Accuracy Estimates, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, April 1997, p. 405-410, 1997