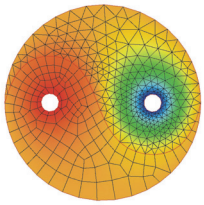
Publishing House
AKAPIT

COMPUTER METHODS IN MATERIALS SCIENCE

Informatyka w Technologii Materiałów

Vol. 10, 2010, No. 1



NEURAL NETWORK-BASED PREDICTION OF ADDITIVES IN THE STEEL REFINEMENT PROCESS

TADEUSZ WIECZOREK[1]*, MIROSŁAW KORDOS[2]

[1] Department of Management and Computer Science, Faculty of Material Engineering and Metallurgy, The Silesian University of Technology, Krasinskiego 8, 40-019 Katowice, Poland,
[2] Department of Mathematics and Computer Science, University of Bielsko-Biala, Willowa 2, 43-309 Bielsko-Biala, Poland
**Corresponding Author: tadeusz.wieczorek@polsl.pl*

Abstract

The paper presents a methodology of improving the efficiency of the steelmaking process based on computational intelligence solutions. To make fine adjustments to the steel composition alloy additions are added to the crude steel to adjust composition for the grade of steel being manufactured. The prediction of metal bath composition is a crucial factor in the economy of ladle furnace operation. Usually it is made by calculations based on the equilibrium of chemical reactions in molten steel. The paper presents the problem solution based on the prediction system build upon the committee of Artificial Neural Networks, the Support Vector Regression and Multivariate Linear Regression Model. A brief state of the art review of the application of computational intelligence (CI) in secondary steelmaking has been made. The prediction system used by authors has been introduced. Problems with data preparation have been presented. Experimental results and the final conclusions and recommendations have been presented. The solutions was implemented in one of the steelworks, where it allowed to improve the economy of the secondary steelmaking process.

Key words: steel refining, neural networks, computational intelligence, SVM, prediction

1. INTRODUCTION

The refining of steel (secondary steelmaking) has become an important part in modern steelworks, which ensures the production of steel of the required properties and the success of continuous casting. Ladle Heating Furnaces (LHF's) are used for steel temperature control, steel deoxidation, reduction of sulfur, alloy additions, inclusion floatation and modification. LHF can also be used as a holding unit if delays occur during production (Engh, 1992). External heat is provided to the ladle to compensate for its loss, by the electric arc. Stirring the steel in the ladle by inert gas increases the reaction between the metal and slag. Stirring also provides the benefit of temperature uniformity throughout the ladle of steel. The steel refining processes performed in the LHF have now become widely adopted, and their various combinations can satisfy specific needs of a given steelmaker.

A typical process has the following stages (Ghosh, 2001):

- Tapping - begins when the heat (batch of steel) is tapped from the Electric Arc Furnace (EAF) into the ladle. The main functions at tapping are:
  - Separation of highly oxidized slag from the steel.
  - Alloys additions to modify steel composition towards that required for the product.
  - Deoxidant addition to control the level of oxidation.

ISSN 1641-8581

  - The main parameters to be controlled at this stage are the amount of C, Mn, P and Si in the steel, Fe contained in slag and steel temperature.
- Flushing - improves homogenization of steel temperature and composition through the stirring effect of the gas (Ar) injected close to the bottom of the ladle via a submerged refractory coated lance or a porous plug in the ladle bottom.
- Other operations may take place, such as addition of alloys or aluminum wire feeding to trim closer to the required composition. Synthetic slag can be added for metallurgical treatments such as desulphurization.
- Reheating at a ladle arc furnace - allows a number of treatments to be carried out:
  - Increase temperature to that required for casting – by means of electrical energy.
  - Composition trimming - by alloy additions or wire feeding.
  - Slag composition adjustment with mineral additions to give the desired metallurgical effect.
  - Deep injection of powdered reagents for desulphurization.
  - Reheating provides a process 'buffer' by maintaining steel temperature in case of production delay.

To make fine adjustments to the steel composition alloy additions are added to the crude steel to adjust composition for the grade of steel being manufactured. The prediction of metal bath composition is a crucial factor in the economy of ladle furnace operation. Usually it is made by calculations based on the equilibrium of chemical reactions in molten steel. The measurement of the compositions require spectrographic techniques which are performed on line. In the present paper a computational intelligence (neural network and SVM) based models have been developed for that purpose. They predict the output variables (quantities of alloy additions) with given a set of up to 33 process (input) variables (the actual chemical composition of the steel and the desired composition). The input variables include mass of the steel bath, temperature and the quantity of different chemical elements in the steel (C, Si, Mn, P, Al, S, etc. thought not always all the input variables are used). The process variables are measured on-line and hence the intelligent system can be used on-line to predict the output parameters.

In our system the problem of prediction of alloy additions has been approached using a committee of Multilayer Perceptron Neural Network (MLP), Support Vector Regression (SVR) and Multivariate Linear Regression (MLR) methods. The paper is organized as follows: section 2 describes a brief state of the art review of the application of computational intelligence (CI) in secondary steelmaking, section 3 introduces the system used by us. In section 4 we present the data preprocessing issues. Section 5 introduces the detail of the MLP prediction. Sections 6 and 7 introduce the other approaches to the prediction problem, i.e. SVR and MLR. Section 8 presents experimental results. Section 9 contains the final conclusions and recommendations.

2. APPLICATION OF COMPUTATIONAL INTELLIGENCE IN SECONDARY STEELMAKING

A survey made by Wong et al., (2000) presented a comprehensive bibliography of neural network application research in business (there are some other comprehensive reviews presented in the literature, (Bhadeshia, 1999; Jams, 2001; Meireles et al., 2003), but not such detailed), in the area of steel plant's real-time process control had identified 250 research articles. Among them papers considering the LHF process are not too many. The main problems discussed in the literature are: experimental modeling of the steelmaking process, electrode lift control system, stirring by inert gas control system, ladle control system and supervisory control for assisting the operator in making a decision about how successfully a batch of steel has been made.

Electrode lift control system is the important component of the ladle furnace system, and its working efficiency has direct effect on the yield, quality of the steel and energy consumption. The position of the electrode, the current and the voltage of the electric arc have strong coupling relationship. It is difficult to build exact mathematic models by using the conventional methods of electrical equivalent circuits (Wieczorek, 2006). Almost all the conventional ladle electrode lift control system adopt the PID control strategy, and often utilize the field experience to set the parameter of the controller (Siemens, 2005). Therefore, the current of the electric arc fluctuate greatly, thus has bad effects on the quality of the molten steel refining (Li et al., 2004). A lot of research have been done aiming at the electrode lift control problem. Intelligent complex control basing on fuzzy control and the adaptive electrode control

algorithm by using the feedforward control have been proposed by Li et al., (2004). W.E. Staib proposed to utilize the neural network to control the electrode, this can learn online during the smelt process (Staib, 1993). In China, as well, an intelligent ladle control system had been developed. Combined AI technologies were used for ladle furnace heat balance calculation and steel temperature prediction, dynamic energy input optimization and intelligent electrode control. Satisfactory application results had been obtained (Sun et al., 2000).

Stirring the steel bath in the ladle by the flow of an inert gas (usually Ar) improves the steel refining process. Important is the flow rate of Ar, dimension of gas bubbles, stirring time and temperature. Li et al. (1997) have presented a method of control the optimal flow rate of Ar. Using fuzzy controller integrated with the feedforward ANN a cascade control system is realized. The controller can keep the flow of Ar at the optimal point during the whole refining period. Peter et. al. (2005) have investigated the mass transfer rate caused by Ar stirring during ladle refining. Process was quantified by taking sequential steel and slag samples during the treatment of 20 heats. Each heat was stirred with a different argon flow rate. Heats were treated at two different plants. Mass transfer rate constants were determined for each heat by using process simulation and thermodynamic models. Relationships between mass transfer rate constants and stirring powers as well as ladle geometries were compared between the two plants. It was found that the reaction kinetics during ladle refining depend on the bulk transport of the steel to the slag/steel interface and on the thermodynamic equilibrium at the slag/steel interface.

In order to assist the operator in making a decision about how successfully a batch of steel has been made, some systems basing on fuzzy logic and neural networks have been developed (Roy et al., 1999; Van Gorp, 1999). They are aiming to assess the quality of the steel production using key processing parameters as inputs and performance parameters as outputs. Roy et al. (1999) have described the secondary steelmaking process and the methodology followed to develop the fuzzy model for process automation. Most steel industry processes are already controlled using proportional integral and derivative controllers. These systems are able to cope with many of the possible production situations but there are some extreme cases where many phenomena, which may be only partially or fully understood, are likely to disturb the production conditions. In these instances, fuzzy logic controllers can give a faster reaction to disturbance. An example is development of a shape control system combining neural networks and fuzzy logic. A back propagation neural network performs pattern recognition and a fuzzy control model calculates required changes to control parameters to retrieve a perfect shape. Roy et al. (1999) have developed an approach to utilize both data analysis and expert knowledge to define the membership functions and the rules.

An application of decision tree method for constructing rules for managing the secondary steelmaking process has been proposed. The C4.5 algorithm has been used (Wieczorek et al., 2007) and the CART algorithm has been used (Wieczorek & Świtala, 2008). Basing on the decision trees some set of rules describing the electric-arc process has been constructed. The proposed algorithms can be considered as a general method for knowledge extraction. Recently, in the world, some intelligent expert systems for control and monitoring of the process are proposed. One of them has been implemented and applied in real industrial process by the research team from the Silesian University of Technology, managed by the author (Wieczorek et al., 2008; Wieczorek & Pilarczyk, 2008).

3. THE PREDICTION SYSTEM

The purpose of the prediction system is based on the measured data of the LHF process (chemical compositions, temperature, etc.) to predict how much of particular additions should be added to the process. That is done in several steps. First the system predicts the quantity of elements to be added and then based on the economic factors and availability of particular additions that contains the required elements, the quantities of particular additions are calculated.

After the feature selection and data preprocessing (which is described in the next section), the data is provided to the core of the prediction system, which consists of MLP, SVR and MLR methods (figure 1). A separate system is built to predict each element in each grade of steel. However, sometimes it happened that there is not enough data (there were too few LHF processes in the past) for a given grade of steel to build the prediction model. Thus, the rarely produced sorts of steel are joined together and then using k-means weighted clustering (weights proportional to the input-output correlation on the whole data Euclidean distance is used) they are

grouped into bigger datasets comprising the data of the production of similar steel grades, for which the regression models are built. If it happens that the prediction results for a given cluster are worse than for a whole dataset, then the model built for the whole dataset (for all rarely produced grades of steel) is used instead of a given cluster model.

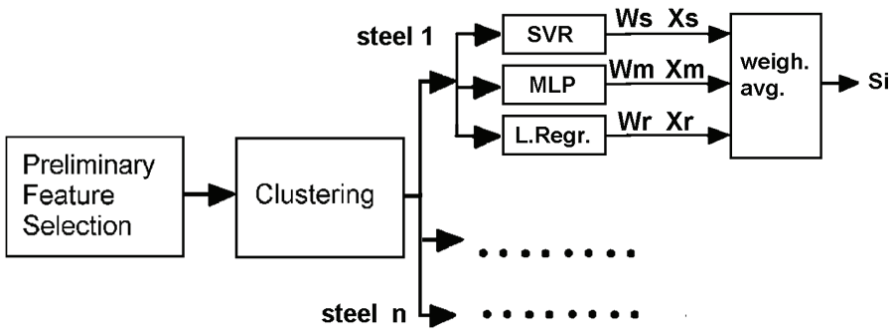


Fig. 1. *The system for predictions of the quantity of alloys used in the LHF process.*

As the experiments showed, a committee of three different methods (MLP, SVR, MLR) usually produced more accurate results than any single method. For that reason the three methods are used and a final predicted value is calculated as a weighted average from the three methods, with higher weights given to this method which performed better in the crossvalidation test. The measure of the method quality we used was the correlation on the test part of the set in the crossvalidation between the predicted value and the observed value.

$$W_M = corr_M^3 \qquad W_S = corr_S^3 \qquad W_R = corr_R^3 \quad (1)$$

$$W = 1 - \max(W_M, W_S, W_R) + W_M + W_S + W_R \quad (2)$$

$$X = (X_{avg}(1 - \max(W_M, W_S, W_R)) + W_M X_M + W_S X_S + W_R X_R) / W \quad (3)$$

where $corr_M$, $corr_S$, $corr_R$ are the correlations between the actual value and the values predicted by MLP, SVR and MLR method respectively, X is the final value predicted by the model, X_M, X_S, X_R are the values predicted by the MLP, SVR and MLR model respectively, X_{avg} is the average value of that element added to the LHF process across all the previous cycles.

After the quantities of particular elements are predicted, the quantities of the additives have to be calculated by solving a system of linear equations (each additive contains usually two or three elements). If the equation system can be solved in several ways (what is frequently the case), the most economic solution (with the lowest total costs of the additives) is selected.

The system has been implemented at one of the Polish steelwork as a web application in .NET technology and it uses the optimal parameters obtained while performing the experiments. The experiments with SVR were performed in *Statistica*, then the optimal SVR parameters were determined. The experiments with MLP were performed with our own software written in *Delphi* and the optimal MLP weights and structure were found. The linear regression parameters also were determined with *Statistica*.

4. DATA PREPARATION

In the real system we use first forward feature selection with beam search and feature rankings. However, for the sake of the simplicity of this paper we do not discuss this issues here, instead we show the results on the 26 preselected features, which are the same for each steel grade. This simplification only slightly affects the prediction results, but make it significantly easier to present and compare the results. Additionally, the MLP network by its nature performs a kind of feature selection by setting the values of the weights that connect inputs from the unimportant variables to very small values.

However, it is worth mentioning here, that the LHF operators at the steelwork must understand why given feature combination influences the result, otherwise, they are not willing to apply the prediction of the system. So even if the prediction is correct but based on the assumptions which are unclear to the LHF operators, the input features for that prediction must be changed and the system re-trained to make the prediction results understandable and thus acceptable to the operators.

Principal Component Analysis, although frequently used in other problems of feature space reduction did not prove to be a suitable tool for our task. First, because the consecutive eigenvalues decrease very slowly for this data and second it makes the results very hard to interpret by an expert or LHF operator who must approve the model.

A good practice is to standardize the data before the training, e.g. according to the following formulae:

$$x_{std} = \frac{x - \bar{x}}{\delta} \qquad \delta = \sqrt{\frac{1}{k}\sum_{i=1}^{k}(x - \bar{x})^2} \tag{4}$$

to make particular inputs influence independent of their physical range. It may be also be beneficial to remove the outliers from the training set. Moreover, a model with higher sensibility in the intervals with more dense data distributed may be preferred. To address the problem, for example, the data can be transferred according to hyperbolic tangent (figure 2, equation 5). The other advantage of such a transformation is the automatic reduction of the outliers' influence on the model. We do not consider the outliers as erroneous values and thus do not reject them, but rather reduce their influence on the final model.

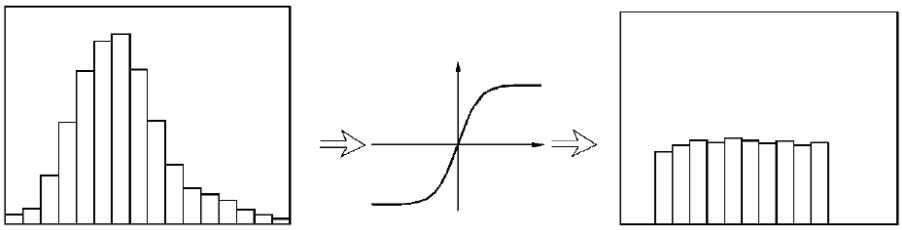


Fig. 2. *The idea of transforming data from a Gaussian-like distribution to uniform distribution.*

As the experiments showed that transformation of both input and output variables improves the prediction results. In the case of neural network, the slope of activation functions of input and output neurons are optimized as a part of network training. In the case of SVR and linear regression, the data is transformed before the training.

5. THE NEURAL NETWORK MODULE

The standard approach to data regression using MLP networks is to apply a 3-layer MLP network with a linear input, logistic sigmoid or hyperbolic tangent (*tanh*) hidden and linear output units.

The neural network we use is a three-layer perceptron (MLP), as shown in figure 3. Each neuron first sums the incoming signals x_i multiplied by corresponding weights w_i and then transforms the weighted sum x by the activation function $y(x)$ (figure 2, equation 5), where y is the output of the neuron (Bishop, 1996):

$$x = \sum_i w_i x_i \qquad y = \frac{1 - \exp(-\beta x)}{1 + \exp(-\beta x)} \tag{5}$$

During the training process the initially random weights are being adjusted so that the network error (the difference between the desired and the actual output for each training vector) is minimized. We used the variable step search (VSS) training algorithms (Kordos & Duch, 2008), because of the effectiveness of incorporating in the algorithm the neuron biases β as additional parameters adjusted in the training process. However other algorithms also could be used after implementing this extension.

The MLP network has hyperbolic tangent activation functions in all layers: input, hidden and output. The adaptable parameters are the weights and biases of the hidden and output layer neurons and the slope of *tanh* functions in input and output layer. The purpose of using *tanh* activation functions in the input layer neurons is to transform the data distribution, as discussed in the introduction to section 4. The purpose of using *tanh* activation function in the output layer neurons is to reduce the influence of outliers on the model outcome. Thus, the outliers can be retained in the training set; they can carry some useful information on rare data points. On the contrary, with standard MLP network, the best approach would be rather to eliminate the outliers.

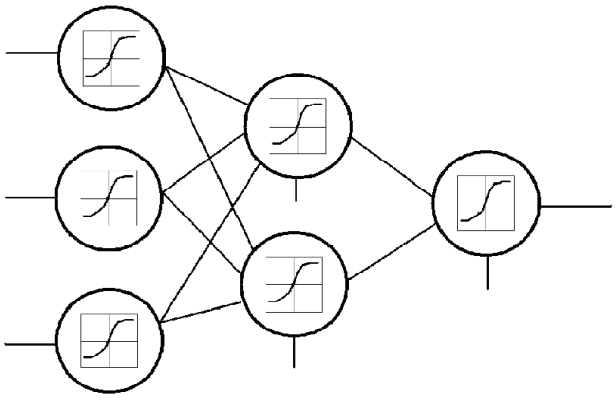


Fig. 3. *Architecture of the MLP neural network.*

The following network parameters are tuned in the system:

- Input layer: β (slope of *tanh* function),
- Hidden layer: weight and biases,
- Output layer: weight, biases and β (slope of *tanh* function).

The neural network learns the optimal parameters of the activation function during the training. The slopes of the activation functions of the hidden layer neurons are fixed ($\beta = 1$) to limit the number of parameters and keep the model simple. However, we are going to conduct some experiments to find out if making them tunable is justified for our purposes.

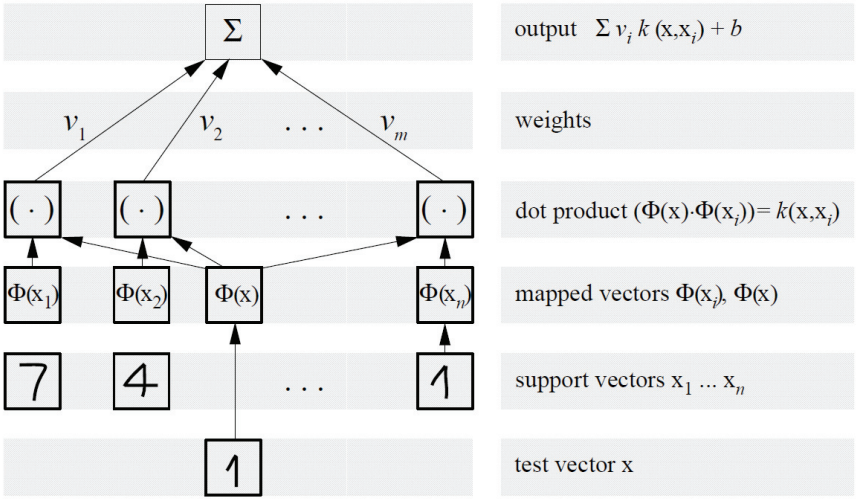


Fig. 4. *Architecture of the Support Vector Regression (source: (Smola & Scholkopf, 2001))*

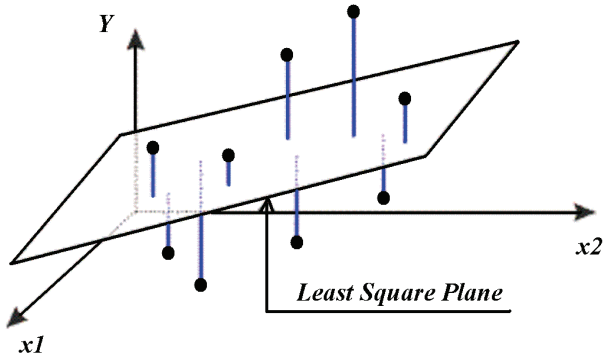


Fig. 5. *The Linear Regression*

5. THE SUPPORT VECTOR REGRESSION MODULE

The goal of Support Vector Machines (SVM) is to find the optimal hyperplane that separates the set of vectors in such a way that vectors, which belong to one category (class) of the target variable are on one side of the hyperplane and vectors of the other class are on the opposite side of the hyperplane. The vectors located close to the hyperplane are called support vectors. Because the original data cannot be frequently separated into two classes by a hyperplane, the SVM model uses so called kernel function to transform the data to a higher dimensional space, where the data can be linearly separated by a hyperplane. SVMs are to some degree similar to neural networks. An SVM with a sigmoid kernel function is equivalent to a two-layer, MLP network (Smola & Scholkopf, 2001).

The predictive model generated by SVM classification (as described above) is obtained using only a small subset of the training data (called support vectors), for which the cost function takes the highest values. That is because the training points that lie far from the margin (separating hyperplane) do not contribute to the error value and thus do not influence the shape and location of the decision boundaries. Analogously, the model produced by SVR (Support Vector Regression) depends only on a subset of the training data, because the cost function for building the model ignores any training data close to the model prediction. SVM and SVR models are solved using quadratic programming. The detailed discussion of the method is out of scope of this paper and can be found in Smola & Scholkopf, (2001).

The experiments with Support Vector Regression were performed in the *Statistica* software. *Statistica* provides an option to generate the source code for the model in C++. Thus we used the generated code and then adjusted it to C# language to be able to use it directly in our system.

6. MULTIVARIATE LINEAR REGRESSION

Least square fitting using the multivariate linear regression we were seeking for simple linear dependences between the linear combination of different elements x_i (inputs) and the quantity of addition if the addition y still has to be added.

$$y = w_o + \sum_i w_i x_i \qquad (6)$$

One may think that it should be only a one-dimensional case, where only the quantity of a single addition in the chemical analysis should determined. However, some elements interact with each other and depending on the quantity of other elements and on the time of the process they may be partially removed from the steel going to the slag instead. Thus in reality, the dependence on the quantity of the same element in the chemical analysis is the strong-

est, but taking into consideration more elements tends to improve the prediction results.

7. EXPERIMENTAL RESULTS

The dataset on which the experiments presented in this section were performed is available from *http://www.kordos.com/datasets/steel26.zip*. Because the data comes from a real steelworks, it is confidential and could not be released as it is. Thus, the 26 input variable names[1] were replaced by x1, … x26, all other inputs were removed, only four most common elements (C, Si, Mn, P) were left and the data was standardized with zero mean and unit standard deviation (it would be standardized anyway before learning the prediction model). We tried different number of hidden units in the neural network. The best results were obtained for the number between 25 and 35. Thus, we finally used 28 hidden units for predicting the quantities of each element. The experiments with MLP neural network were performed with software created by us in Delphi. (*http://www.kordos.com/mknn.html*) The experiments with linear regression and with SVR were performed in the *Statistica* software, using RBF kernels with γ=0.056 and Regression Type 2 (that were the parameters giving the best results). Optimal *C* and *nu* parameters were determined during the training individually for each addition. SVR was chosen for because it is an effective method, which even frequently performs better than MLP. However, on this data it did not, except for predicting the quantity of silicon. Nevertheless, including SVR and sometimes even including MLR as a weighted component of the system (figure 1) improved the results.

Ten runs of 10-fold crossvalidation were performed and the average results on the test part of the dataset are reported in table 1 and table 2. The MSE errors are always reported on the original output data (not on the data transformed by the *tanh* function). The standard deviations were usually of the order of 0.002-0.004 for between-crossvalidation and 0.01-0.03 for within-crossvalidation runs and the S.D. ratio was between 0.08 and 0.13. The training time of the network was of the order of tens of seconds up to single minutes, depending on the dataset size. However, the time of building a full model (for one grade of steel or for one cluster), including feature selection and selection of optimal neural network architecture (which can be different for each feature subset) and SVR parameters can even be of the order of hours on a single processor. That is acceptable in our application, since the model must be built only once. The system must only predict the data on the fly and the prediction is immediate.

Table 1. *Mean Squared Error of predicting the quantity of additives to be added in the LHF process. Prediction performed on the whole dataset. ("prep." means that the input and output data was preprocessed prior to the training by a constant tanh function, selected manually to give the better transformation to a uniform data distribution.)*

	MLP+ATF 26-28-1	MLP, *β=1* 26-28-1	SVR (RBF, γ=0.056)	MLP 26-28-1, *β=1*, prep.	SVM (RBF, γ=0.056), prep.
C	0.035	0.041	0.059	0.036	0.050
Si	0.073	0.095	0.065	0.076	0.065
Mn	0.106	0.119	0.163	0.111	0.154
P	0.134	0.148	0.185	0.140	0.142
average	0.087	0.101	0.118	0.091	0.103

Table 2. *Mean Squared Error of predicting the quantity of additives to be added in the LHF process. Prediction performed on 5 clusters, as described in chapter 3.*

	MLP+ATF 26-28-1	MLP, *β=1* 26-28-1	SVM (RBF, γ=0.056)	MLP 26-28-1, *β=1*, prep.	SVM (RBF, γ=0.056), prep.	Multivariate Linear Regression	Full Model
C	0.032	0.038	0.054	0.034	0.048	0.051	0.027
Si	0.069	0.080	0.064	0.072	0.076	0.080	0.057
Mn	0.098	0.111	0.151	0.105	0.147	0.159	0.094
P	0.126	0.140	0.172	0.135	0.136	0.144	0.115
average	0.081	0.092	0.110	0.086	0.102	0.108	0.073

8. CONCLUSIONS

A system for predicting the quantities of additives in the steel production process was presented. The paper concentrates on the computational intelligence based prediction block of the system. We have proposed that the input and output variable trans-

[1] The input variables are: chemical analysis at the input stage of the LHF process, chemical analysis at the output stage (demanded), time and temperature of the process stage, energy and gas (Ar) used during this stage, mass of the steel bath.

formation from a Gaussian-like to uniform-like distribution to be performed during the neural network training, incorporated in the training method. That gives better results than preprocessing the data before the training, since it allows adjusting the distribution individually for each feature, without a strong a priori assumption that the optimal distribution is always uniform. As it can be seen from the tables 1 and 2, MLP network with adaptable activation function provides the best results, however not so good as the full system (consisting of ANN, SVR and MLR modules – figure 1). That results can probably be further improved if the selection of the activation functions in the hidden layer is performed during the training (Eskander, 2008).

The goal of the system is to precisely calculate quantities of additives demanded during the LHF process, thus reducing the cost of a single refinement process and consequently to allow producing more steel monthly. To make the system easier for understanding by experts, what is crucial in our application we consider incorporating rule extraction from neural network using an adaptation of one of Setiono's methods (Setiono & Thong, 2004).

ACKNOWLEDGEMENTS

The project was partially sponsored by the grant No 6ZR9 2006 C/06742 from the Polish Ministry of Education and Science (MNiSZW).

REFERENCES

Bhadeshia, H., 1999, Neural Networks in Materials Science, *ISIJ Int*., 10, 966-979.

Bishop, C., 1996, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford.

Engh, T.A., 1992, *Principles of metal refining*, Oxford Univ. Press, Oxford.

Eskander G. S., 2008, *Round Trip Time Prediction Using the Symbolic Function Network Approach*, Available from: http://arxiv.org/ftp/arxiv/papers/0806/0806.3646.pdf (last accessed 1.12.2009).

Ghosh, A, 2001, *Secondary steelmaking*, CRC Press Boca Raton, New York.

Jams, S.-L., 2001, Current status and future trends in the automation of mineral and metal processing, *Control Engineering Practice,* 9, 1021-1035.

Kordos, M., Duch, W., 2008, Variable Step Search Algorithm for Feedforward Networks, *Neurocomputing*, 71, 13-15, 2470-2480.

Li, M., Zhou, Z., Shi, L., 2004, The Neural Network Decoupling Control Based on The Internal Model Control, *Proceedings of the 5th World Congress on Intelligent Control and Automation*, Hangzhou, P.R. China.

Li, S., Gao, X., Chai, T., 1997, Fuzzy control and its application in Ar stirring system, *Proc. of 2nd Asian Control Conf.* Seul., 3, 711-714.

Meireles, M., Almeida, P., Simões, M., 2003, A Comprehensive Review for Industrial Applicability of Artificial Neural Networks, *IEEE Transactions on Industrial Electronics,* 50, 3, 585-601.

Peter, J., Peaslee, K., Robertson, D., Thomas, B., 2005, Experimental Study of Kinetic Processes during the Steel Treatment at Two LMF's, *Conference Proceedings of AISTech*, 1 & 2, 959-973.

Roy, R., Collantes, L., Berdou, J., Whittaker H., Madill, J., 1999, Fuzzy Process Modeling for Secondary Steelmaking, *Proc. of 18th International Conference of the North American Fuzzy Information Processing Society,* New York, 1, 864-868.

Setiono, R., Thong, J., 2004, An approach to generate rules from neural networks for regression problems, *European Journal of Operational Research*, 155, 1, 239-250.

Siemens A.G., 2005, SIMELT NEC – neural network based energy control efficiency, Industrial solution and services, Siemens Press, 1-8.

Smola A, J., Scholkopf B., 1998, *A Tutorial on Support Vector Regression*, Available from: http://www.svms.org/regression/SmSc98.pdf (last accessed 1.12.2009).

Staib, W.E., 1993, The Intelligent Arc Furnace, *Steel Technology Int.*, 2, 103-107.

Van Gorp, J., 1999, Steel Plant Modeling and Analysis Using a Neural Network, *ANNIE, Intelligent Engineering Systems Through Artificial Neural Networks*, 9, 739-744.

Wieczorek, T., Mączka, K., Świtała, P., 2007, Automatyczne tworzenie baz wiedzy z wykorzystaniem drzew decyzyjnych, *Bazy danych – nowe technologie. Bezpieczeństwo wybrane technologie i zastosowania,* eds, Kozielski S., Małysiak B., Kasprowski P., Mrozek D., WKiŁ Warszawa, 385-392.

Wieczorek, T., 2006, Intelligent control of the electric-arc steelmaking process using artificial neural networks, *Computer Methods in Material Science,* 6, 1, 9-14.

Wieczorek, T., Blachnik, M., Mączka, K., 2008, Building a model for time reduction of steel scrap meltdown in the electric arc furnace (EAF). General strategy with a comparison of feature selection methods, *Lecture Notes in Artificial Intelligence*, 5097, 1149-1159.

Wieczorek, T., Pilarczyk, M., 2008, Classification of steel scrap in the EAF process using image analysis methods, *Archives of Metallurgy and Materials*, 53, 2, 613-618.

Wieczorek, T., Świtała, P., 2008, Wykorzystanie algorytmu CART do automatycznego tworzenia bazy wiedzy systemu ekspertowego, *Bazy danych, rozwój metod i technologii. Architektura, metody formalne i zaawansowana analiza danych*, eds, Kozielski S., Małysiak B., Kasprowski P., Mrozek D., WKiŁ Warszawa, 439-450.

Wong, B. K., Lai, V. S., Lam, J., 2000, A bibliography of neural network business applications research, *Computers & Operations Research,* 27, 1045-1076.

Yanguang S., Daixian W., Yan T., Shi T., Fang Y., Wang S., 2000, An Intelligent Ladle Furnace Control System, *Proceedings of the 3th World Congress on Intelligent Control and Automation,* Hefei P.R. China, 1, 330-334.

APPENDIX 1. DATASET PROPERTIES

The dataset used in the experiments (available at *http://www.kordos.com/datasets/steel26.zip*) contains 26 continues inputs and four outputs (C, Si, Mn, P). There are 2384 vectors in the data set.

Table 3. *Correlations between input and output variables in the steel26 in the whole dataset (absolute values)*

out\in	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13
C	0.02	0.05	0.01	0.41	0.72	0.09	0.12	0.08	0.32	0.09	0.12	0.09	0.63
Si	0.05	0.05	0.02	0.00	0.05	0.06	0.26	0.07	0.06	0.02	0.05	0.03	0.08
Mn	0.19	0.21	0.18	0.09	0.16	0.75	0.11	0.24	0.06	0.29	0.34	0.13	0.16
P	0.20	0.24	0.18	0.34	0.33	0.46	0.01	0.11	0.36	0.11	0.40	0.16	0.54

out\in	x14	x15	x16	x17	x18	x19	x20	x21	x22	x23	x24	x25	x26
C	0.07	0.01	0.10	0.06	0.89	0.17	0.68	0.26	0.08	0.32	0.29	0.12	0.37
Si	0.01	0.07	0.01	0.06	0.15	0.25	0.12	0.02	0.05	0.06	0.01	0.24	0.14
Mn	0.28	0.49	0.15	0.41	0.18	0.01	0.12	0.05	0.14	0.06	0.71	0.02	0.63
P	0.10	0.30	0.19	0.42	0.09	0.04	0.17	0.01	0.24	0.36	0.96	0.04	0.12

Table 4. *Number of records and S.D. ratio in each cluster*

Cluster No	whole dataset	1	2	3	4	5
Number of vectors	2348	138	889	251	678	428

PREDYKCJA DODATKÓW STOPOWYCH W PROCESIE RAFINACJI STALI Z WYKORZYSTANIEM SIECI NEURONOWYCH

Streszczenie

Dla uzyskania żądanego składu chemicznego stali wprowadzane są do niej w procesie produkcji dodatki stopowe. Precyzyjne przewidywanie końcowego składu chemicznego produkowanej stali jest niezwykle ważne dla ekonomicznego prowadzenia procesu w piecu kadziowym. Zwykle obliczenia są prowadzone w oparciu teorię procesów chemicznych zachodzących w ciekłej stali w warunkach równowagi. W pracy problem przewidywania masy dodatków stopowych koniecznych do uzyskania żądanego składu został rozwiązany przy użyciu sztucznych sieci neuronowych i algorytmu maszyny wektorów podpierających (SVM). Opisano opracowany model inteligentny obliczania dodatków. Przedstawiono zbudowane moduły tego systemu: sieci neuronowe, SVM, regresję liniową wielu zmiennych. Omówiono zagadnienia preprocesingu danych do uczenia modułów inteligentnych. Przedstawiono uzyskane wyniki, wnioski wyciągnięte z badań oraz efekty uzyskane w rzeczywistych warunkach przemysłowych.

Received: December 2, 2009
Received in a revised form: March 15, 2010
Accepted: March 30, 2010