Prediction of boiler gas side effective heat transfer coefficients using mixture density networks and historic plant data

Renita Raidoo^{1,*} and Ryno Laubscher^{2,**}

¹University of Cape Town, Rondebosch, Cape Town, 7700

²University of Stellenbosch, Stellenbosch Central, Stellenbosch, 7599

Abstract. Machine learning has received increased recognition for applications in engineering such as the thermal engineering discipline due to its abilities to circumvent thermodynamic simulation approaches and capture complex interdependencies. There have been recent headways to couple deep learning models to process simulations, given the deeper insight they can provide. The present study entails the development of a mixture density network (MDN) capable of predicting effective heat transfer coefficients for the various heat exchanger components of a utility scale boiler. Large boilers are susceptible to dead zones and other anomalous phenomena that influence performance and manifest as multimodalities in the measured data, which system-level 1D process models struggle to capture. The overall water-side heat load calculation, as well as mass and energy balances around the components were done to determine the heat transfer coefficients at each stage of the boiler using historic sensor data. The measured data was then used to train a deep learning model capable of outputting predicted heat transfer coefficients and local model uncertainty. The predictive model can be coupled to a water circuit process model which can be used to study aspects such as metal temperatures and operating philosophies at the different operating loads of the plant.

1 Introduction

Monitoring the thermal performance of boiler heat exchangers is crucial to ensure safe and efficient operation of power plants. Thermal monitoring is used in estimating residual life of boiler tubes and heat exchanger effectiveness. Boiler tubes are subjected to extreme conditions in coal-fired power plants which can lead to mechanical failures. These failures can be caused by a number of problems such as, slagging, fouling, caustic embrittlement, fire-side corrosion and thermal fatigue failure with the latter being one of the main reasons for unwanted plant downtime [1],[2]. Managing boiler tube failures is important as it can help in reducing forced outages and, in turn, improve plant availability and reliability [1]. Heat exchangers can experience rating problems whereby the actual heat absorption is lower when

*e-mail: rdxren002@myuct.ac.za

**e-mail: rlaubscher@sun.ac.za

compared to the intended design heating duty [3]. Heat transfer from the flue gas to the heating surfaces is largely reduced by the effects of ash deposition and non-uniform sweeping of flue gases. Increased levels of deposition and non-uniform flow leads to higher heat exchanger outlet temperatures which results in higher levels of slagging on downstream heat exchangers, lower boiler thermal efficiency and higher fuel demand [4]. Therefore, a need exists to estimate plant heat exchanger effectiveness and tube metal temperatures using actual plant measurement data, seeing as conventional empirical process models struggle to capture the various effects in the boiler. A model capable of capturing actual plant thermal performance would enable studying historic heat exchanger metal temperatures to estimate residual life and quantify effects of boiler input parameters on heat exchanger performance.

The present study proposes a data-driven method of determining effective heat transfer coefficients with minimal plant data using a mixture density network (MDN) approach. The proposed methodology will be applied to a 618 MWe utility-scale pulverised fuel boiler.

Practical machine learning problems can often have significantly non-Gaussian distributions due to the presence of unmeasured phenomena and measurement uncertainty [6]. The MDN model is useful as it has the ability to make better predictions on multimodal data since it is able to predict multiple Gaussian distributions for single output parameters. Bishop [6] proposes a framework for modelling multiple probability distributions such that a range of predicted values can be obtained with an inherent uncertainty. The benefit of this is that the operator can make an informed decision based on the probability and standard deviation given with an associated heat transfer coefficient prediction.

Artificial neural networks (ANNs) have been used in a broad range of applications including: pattern classification, pattern recognition, optimisation, prediction and automatic control [7]. Many recent studies in the field of thermal engineering and plant performance monitoring have incorporated the use of machine learning as an alternative to classic methods. Hu et al [8] make use of a short-term memory-based autoencoder network for early detection of anomalies. The study showed excellent results which indicate that the framework can be used as a legitimate tool for plant performance monitoring. Another study from Herawan et al [9] shows the success of using an ANN to predict the power generation of a waste heat recovery system. Their work highlights the method's ability to perform in highly fluctuating environments with multiple inputs affecting the output, as is common with thermal combustion systems. Other uses of machine learning in power plants include system enhancement and optimisation [10], health monitoring [11] and complex thermal management [5].

In the present work, heat transfer coefficients are predicted using plant data from an actual subcritical boiler located in Southern Africa over a period of 336 days. The heat absorbed by the heat exchangers is obtained by calculating the increase in enthalpy of the water/steam in each circuit using measured pressures and temperatures. The data in this study utilises 70 inputs for the calculations used to estimate heat coefficients. These heat exchanger calculations entail the use of combustion calculations and mass and energy balances. The required heat transfer rates are determined, and subsequently the flame temperature, mass flow rate of fuel and heat transfer coefficients at the various heat exchangers of the boiler system.

Following this, an ANN is developed and coupled with a probabilistic model to create a MDN which is capable of outputting multiple conditional probability distributions. It should be noted that the input to the model has only 13 features which are direct inputs to the boilers operation. The inputs include steam flow rate demand, ambient temperature, excess air ratio and final steam thermal conditions among other integral pressure and temperature readings. These inputs are user controlled and are not products of the boiler's operation. The output

parameters of this model are the standard deviations, mixing coefficients and predicted means (the predicted heat coefficient value). The model is validated using the mean average error and statistical inference methods, namely prediction intervals and confidence intervals.

2 Materials and methods

2.1 Natural convection boilers

The boiler in the present study is of the radiant water-tube type, operating on a natural circulation principle. The boiler consists of 7 heat exchangers, namely the furnace, platen superheater, final superheater, secondary reheater, primary superheater, primary reheater and economiser. The layout of the flue gas and water circuits can be seen in Figure 1a and 1b.

Figure 1: Schematic of boiler

2.2 Calculation of heat transfer coefficients from measurement data

Heat transfer coefficients represent the proportionality between the heat flux and the temperature gradient between two fluid control volumes. In order to determine the heat transfer coefficient θ_i for the i^{th} heat exchanger, measured data is used for mass and energy balance calculations. These need to be performed to determine the average control volume temperatures as well as the heat transfer rate from the hot to the cold fluid. For the below heat transfer calculations a quasi-steady flow assumption was made. The heat transfer rate per heat exchanger is calculated using the difference between the water inlet and outlet enthalpies as seen in equation 1.

$$\dot{Q}_{HX,i} = \dot{m}_{s,i}(h_{s,out,i} - h_{s,in,i}) \quad (1)$$

The enthalpies are evaluated at the inlet and outlet positions of the heat exchanger using the measured temperatures and pressures in the water circuit. The mass flow of steam is evaluated at each stage of the boiler since there are added flows between the superheaters and

reheaters from the attemperation. The furnace heat transfer calculation differs slightly for the subcritical boiler. The furnace heat load is determined using the feedwater flow into the steam drum and the latent heat of evaporation of water evaluated at the steam drum pressure. The total water heat load per measured observation is calculated by summing the heat transfer rates of each boiler heat exchanger, $\dot{Q}_{steam} = \sum \dot{Q}_{HX,i}$.

The gas composition of the flue gas flowing through the boiler is determined using the combustion products from a global step complete combustion calculation which takes the fuel ultimate analysis and the excess air ratio α as inputs. The ultimate analysis of the fuel is taken from the design data, C: 0.4156, H: 0.0222, N: 0.0097, O: 0.079, S: 0.0094, H_2O : 0.055 and Ash: 0.409 [12]. The quasi-steady fuel flow rate per measurement is found by balancing the total boiler energy inputs and energy outflows, as seen in equation 2. For the energy inputs; $h_{fuel,sensible}$ denotes the sensible enthalpy of the fuel and HHV denotes the higher heating value of the fuel. Furthermore, $\dot{m}_{fuel}$ is the mass flow rate of fuel, $\dot{m}_{dry,air}$ is the dry air supplied per kilogram of fuel, $\dot{m}_{moist,air}$ is the moisture entering the boiler along with the air per kilogram fuel and $h_{air,ambient}$ is the ambient enthalpy of the air. For the energy outflows, the heat transfer plus all the losses in the system need to be considered. These losses include the total heat loss due to the unburnt carbon, radiation, bottom ash, unaccounted losses and losses caused by ash at the exit. $Y_{ar,C,loss}$ is the as-received fraction of unburnt carbon per kilogram of fuel, $f_{rad,loss}$ is the percentage of chemical energy lost to the environment, $Y_{ar,ash}$ is the as-received ash mass fraction, $h_{bottomash} = c_{p,ash} (1073K - 298K)$ is the bottom ash heat loss, $C_{unaccounted}$ is the unaccounted heat loss, $\dot{m}_{fg} = \dot{m}_{fuel} \cdot (1 - Y_{ar,ash} + \dot{m}_{dry,air} + \dot{m}_{moist,air})$ is the flue gas mass flow rate and $h_{fg,out}$ is the boiler exit flue gas enthalpy calculated using a real gas mixture property database.

$$\dot{Q}_{sensible,fuel} + \dot{Q}_{combustion,fuel} + \dot{Q}_{combustion,air} = \dot{Q}_{steam} + \dot{Q}_{carbonloss} + \dot{Q}_{radiationloss} + \dot{Q}_{bottomashloss} + \dot{Q}_{unaccountedloss} + \dot{m}_{fg,out} h_{fg,out} + \dot{Q}_{sensibleloss} \quad (2)$$

$$\begin{aligned} \dot{Q}_{sensible,fuel} &= \dot{m}_{fuel} h_{fuel,sensible} \\ \dot{Q}_{combustion,fuel} &= \dot{m}_{fuel} HHV_{fuel} \\ \dot{Q}_{combustion,air} &= \dot{m}_{fuel} (\dot{m}_{dry,air} + \dot{m}_{moist,air}) h_{air,ambient} \\ \dot{Q}_{carbonloss} &= \dot{m}_{fuel} Y_{ar,C,loss} HHV_{carbon} \\ \dot{Q}_{radiationloss} &= \dot{m}_{fuel} HHV_{fuel} f_{rad,loss} \\ \dot{Q}_{bottomashloss} &= \dot{m}_{fuel} Y_{ar,ash} (1 - Y_{flyash}) h_{bottomash} \\ \dot{Q}_{unaccountedloss} &= \dot{m}_{fuel} HHV_{fuel} C_{unaccountedloss} \\ \dot{Q}_{sensibleloss} &= \dot{m}_{fuel} Y_{ar,ash} Y_{flyash} h_{flyash,exit} \end{aligned}$$

Once the fuel flow has been determined, the flame temperature (furnace inlet gas temperature) can be found using an adiabatic combustion control volume calculation. The flame temperature and furnace heat transfer rate, calculated using the measured data, is then used to determine the furnace exit gas temperature. Radiation leaving the furnace exit plane is ignored. Using the inlet gas temperatures and calculated heat transfer rates per heat exchanger, as for the furnace, the exit gas temperatures for each heat exchanger can be found using a simple energy balance calculation.

To account for the air flow that contributes to the flue gas composition, two modelling factors are included in the air flow calculation namely the excess air ratio α and the air leakage factor

α_{leak} . The percentage of O_2 is measured at the air-heater outlet and used to find the leak air flow at a 100% load. Using CFD simulation data [12] of the current boiler at 100% load case, α was found and in turn the α_{leak} using an iterative root-finding solver. For the remainder of the data observations α_{leak} was fixed and α recalculated to find the input required to match the measured O_2 .

As mentioned, the average gas- and water-side inlet and outlet temperatures are required to determine the temperature gradient between the gas and water control volumes, as shown in equation 3. The temperature gradient is required to calculate the heat transfer coefficients. The water temperatures are measured and gas temperatures calculated as explained previously. The inlet gas temperature to a given heat exchanger is assumed to be the outlet gas temperature of the previous heat exchanger. Using the known heat transfer rate and inlet gas temperature the outlet gas enthalpy can be found as: $h_{fg,out}(T_{fg,out}) = h_{fg,in}(T_{fg,in}) - \dot{Q}_i/\dot{m}_{fg}$. The outlet enthalpy is then used to determine the outlet temperature of the heat exchanger.

$$\Delta T_i = \bar{T}_{fg,i} - \bar{T}_{s,i} \quad (3)$$

Once the average temperatures are found, the heat transfer coefficient can be determined using the effective heat transfer area of the heat exchanger as seen in equation 5.

$$\bar{T}_{s,i} = \frac{T_{s,in,i} + T_{s,out,i}}{2} \quad \bar{T}_{fg,i} = \frac{T_{fg,in,i} + T_{fg,out,i}}{2} \quad (4)$$

$$\theta_i = \frac{\dot{Q}_i}{A_i \Delta T} \quad (5)$$

Figure 2 shows the heat transfer coefficients of the boiler heat exchangers at various boiler loads, represented by $\dot{Q}_{steam}$. It can be seen that the heat transfer decreases along the flue gas path, the highest rates are found at the furnace and the lowest at the economiser. The clustering around each of the bands is indicative of the multi-modality of the data which suggests external forces have significant effects on the distribution of the coefficients. The heat transfer to the furnace walls, platen superheater and final superheater for the case study boiler is mainly due to radiation from the hot combustion gases [12]. The unsteadiness of the combustion processes leads to a large variation in estimated heat transfer coefficients per given load for these radiant heat exchangers. It is seen as the heat exchanger location moves further away from the furnace the band of data scatter reduces.

3 Deep Learning Method

3.1 Data

The input data to the machine learning model is taken from pressure, temperature and mass flow sensors on the plant and are listed in table 1. Reheater components have been abbreviated to RH. It should be noted that the inputs to the MLP are all direct inputs to the boiler operation and not results of the boilers operation; so essentially these parameters are operator controlled settings. The main steam temperature, which is the inlet steam temperature to the high pressure turbine, as well as the reheater 2 outlet temperature are in effect controlled parameters since they are directly affected by the amount of attemperation per boiler load.

Figure 2: Heat Exchanger θ at various boiler loads

Table 1: Measured inputs to the ANN

	Feedwater	Steam drum	Main steam	RH 1 inlet	RH 2 outlet	Attemperator 1	Attemperator 2	RH attemperator
Temperature	x		x	x	x			
Pressure	x	x			x			
Mass flow rate	x			x		x	x	x

The output data matrix has 7 columns which contain the calculated heat transfer coefficient values for the various heat exchangers. The rows of the matrix correspond to the different observations.

The data is split into 85% training set and 15% validation set. The purpose of the validation set is to test the inference ability of the model. It is used to determine the model accuracy in a 'real-world' deployment. The model is trained using the training set, so the accuracy is not a true representation of the model's performance due to built in training bias. The model has technically never seen the validation set before, therefore its predictions are objective and impervious to prior information.

3.2 Multilayer perceptron neural networks

A multilayer perceptron (MLP) neural network is a fully connected layered feedforward network [13]. This means that every node in a given layer is connected to every node in the adjacent layer, however there are no connections between nodes of the same layer. The network is parameterised by weight W and bias b matrices. Different layers extract different representations of features from the input data, each layer learns to extract more complex and deeper features from its previous layer [14].

The function of a neural network can be seen as two phases, firstly training and secondly generalising i.e. making predictions.

Forward propagation can be seen as the forward pass of information from layer to layer. Each layer accepts inputs, applies some activation function and then passes it to the successive layer until the final layer outputs the prediction $\hat{Y}$. The activation function used for this model is a ReLu function which outputs a value directly proportional to the input. If the input is positive,

or it outputs zero if the input is negative. In the first line of equation 6, the weights are applied to the output of the previous layer $A^{[l-1]}$. It should be noted that the first layer of the network is a matrix of inputs such that $A^{[0]} = X$. Once the activation function is applied, the pattern continues until the final activation $g(z)$ is applied to produce the predicted value at the output layer L .

$$\begin{aligned} Z^{[l]} &= W^{[l]}A^{[l-1]} + b^{[l]} \\ A^{[l]} &= g^{[l]}(Z^{[l]}) \\ A^{[L]} &= \hat{Y} \end{aligned} \quad (6)$$

Training is performed by the process of forward- and backward-propagation which aims to adjust the trainable parameters, W, b . The purpose of this is to minimise the cost function shown in equation 7. Once a complete forward propagation cycle has been completed, the model uses the predicted output to estimate the cost function value which indicates how close the prediction is to the actual value.

$$J_{MSE}(\bar{y}^n, \hat{y}^n) = \frac{1}{d_{target}} \sum_{j=1}^{d_{target}} \frac{1}{2} (\bar{y}^n - \hat{y}^n)^2 \quad (7)$$

Backward propagation is the automatic differentiation algorithm used to calculate the gradients of the weights and biases in a neural network graph structure with respect to the cost function value. These parameter gradients are used to update the trainable parameters using a gradient descent algorithm. Gradient descent is an optimisation algorithm which minimises the error of a predictive model with regard to a training dataset and the cost function. In the current work, the model is optimised using the ADAM algorithm shown in equations 8 and 9 [15]. It is known that the default configuration parameters do well on most problems and were therefore used in the present work.

$$\begin{aligned} V_{dW} &= \beta_1 V_{dW} + (1 - \beta_1) dW & V_{db} &= \beta_1 V_{db} + (1 - \beta_1) db \\ S_{dW} &= \beta_2 S_{dW} + (1 - \beta_2) dW^2 & S_{db} &= \beta_2 S_{db} + (1 - \beta_2) db^2 \end{aligned} \quad (8)$$

$$W = W - \alpha \frac{V_{dW}}{\sqrt{S_{dW}} + \epsilon} \quad b = b - \alpha \frac{V_{db}}{\sqrt{S_{db}} + \epsilon} \quad (9)$$

Where V_{dW} , V_{db} , S_{dW} and S_{db} are initialised at 0. The process of forward and back propagation and gradient descent continue until optimal weights and biases have been found.

3.3 Mixture Density Networks

MDNs developed in the present study are standard feedforward artificial neural networks with a probabilistic extension module which outputs the mixture parameters, μ, π, σ . These parameters define κ probability distributions within the multi-modal data. The ANN, which incorporates stacked densely-connected neural layers has 7 neurons for the corresponding output parameters. The resulting MDN output for the current work, therefore, has shape $K \times 7 \times 3$. For any given input value, the mixture model provides the formal mathematical description for modelling an arbitrary conditional density function as seen in equation 10.

$$P(\bar{Y}|\bar{X}) = \sum_{k=1}^k (\bar{X}) \mathbb{N}(\bar{Y}|\bar{\mu}_k(\bar{X}), \bar{\sigma}_k^2(\bar{X})) \quad (10)$$

The schematic below in figure 3 shows the general architecture of the MDN model. An arbitrary number of input nodes and hidden layers are shown.

Figure 3: ANN-MDN Schematic

The mixing coefficients, denoted by π , must satisfy the constraints: $\sum_{k=1}^k \pi_k(x) = 1$ and $0 \leq \pi_k(x) \leq 1$ [6]. This is achieved by passing the activation from the fully connected layer through a softmax function as seen in equation 11.

$$\pi_k(x) = \frac{\exp(a_k^{\pi,n})}{\sum_{l=1}^k \exp(a_l^{\pi,n})} \quad (11)$$

The variance must satisfy $\sigma^2 \geq 0$, it is therefore represented in terms of the exponential of the corresponding activation such that $\sigma_k(x) = \exp(a_k^{\sigma})$ [6]. The error function shown in equation 12 is defined as the negative logarithm of likelihood [6]. The derivatives with respect to w (vector of weights and biases) of the error function are determined in order to minimise the error.

$$J(\bar{Y}, \bar{\pi}, \bar{\sigma}, \bar{\mu}) = - \sum_{n=1}^{d_{target}} \ln \left\{ \sum_{k=1}^k \pi_k(\bar{x}^n, \bar{W}) \cdot N(\bar{y}^n | \bar{\mu}_k(\bar{x}^n, \bar{W}), \bar{\sigma}_k(\bar{x}^n, \bar{W})) \right\} \quad (12)$$

3.4 Model Development

A course grid search was performed to determine the best architecture for the network. The present study tested models with one and two layers, with varying numbers of neurons. The batch size was kept constant at 128 and a low learning rate of 0.0001 was used to ensure optimal convergence. The performance of the tested architectures were compared using the error metrics, MAE and MAPE, shown in equation 13.

Once the best-performing architecture has been identified using the standard feedforward ANN, the MDN module was added. Similar to the ANN various MDN hyperparameters were tested and the best-performing values used.

Once the final model had been developed and completely trained, its accuracy was tested using the validation dataset.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y - \hat{y}| \quad MAPE = \frac{100\%}{n} \sum_{i=1}^n \left| \frac{y - \hat{y}}{y} \right| \tag{13}$$

4 Results and Discussion

As can be seen in table 2, the model with 2 hidden layers and 15 neurons performs the best since it had the lowest MAE. The model was tested using 2, 3 and 5 distribution components and it was decided that 2 components captured the modes in the data sufficiently well as there was no significant increase in prediction accuracy with more components.

Table 2: Hyperparameter Search Result: MAE on Development Dataset

Layers	25	15	10
1	21.96	9.83	10.155
2	10.86	3.22	20.77

The main requirement of the model was to be able to predict the heat transfer coefficients of all 7 heat exchangers in the boiler. The output signals are shown in figure 4. Specific points are indistinguishable however it can be seen that the model does capture the general trend of the data since the observations and predictions predominantly overlap. Another interesting feature of the model is its capacity to interpret noisy or disrupted signals. This is seen in the sections of straight lines. The model is able to fit a strongly correlated approximation which mimics this sensor failure. A detailed sample which includes the confidence and predictive interval bands is shown below in figure 6a, 6b and 6c. The final model’s results are shown in table 3.

Figure 4: Heat transfer coefficient predictions for all heat exchangers in the boiler system

Table 3: Model Performance Results

	Training	Validation
MAE [$\frac{W}{m^2K}$]	3.35	3.11
MAPE	7.78%	7%

Figure 5 shows the error distribution graphs for the training and validation set. It can be seen that 57% of the predictions have less than a 10% error. The validation set has approximately 60% of the predictions within the 10% error margin.

Figure 5: Error Distributions

Figure 6 shows the observed data, the predicted mean, the confidence interval band and the prediction interval band. Figure 6a shows a sample of the validation dataset of the furnace. The furnace, as was mentioned, is the stage of the boiler that is most susceptible to fluctuations in the boiler’s operating parameters.

Figure 6c shows a sample of the development dataset of the economiser which is the final stage in the flue gas path of the boiler. A constant offset is noticed for the economiser heat transfer coefficient predictions, but the magnitude of the error is relatively low. Furthermore, the fluctuations in the observed data is significantly lower when compared to the furnace values.

The boiler tubes which typically fail due to localised overheating are the superheater elements, more specifically the final superheater seeing as it has the highest steam temperatures. It is, therefore, encouraging to see that the model is capable of predicting the actual heat transfer coefficients with relative accuracy. Although there are some outliers, the confidence interval captures the vast majority of these peaks.

The model’s accuracy and precision were analysed using statistical inference methodologies. The main focus of this study is to determine the efficacy of this model on making predictions, therefor prediction intervals and confidence intervals are used as error metrics. Although prediction intervals and confidence intervals have underlying similarities, they represent different properties of the data. The confidence interval is the range which has a 95% chance of containing the true population of the data. It is based off the statistical parameters, standard deviation and mean of the estimated predictions generated by the model. It can be interpreted as the likelihood of the observed data falling within 2 standard deviations of the predicted mean. The predictive interval stipulates what range the predicted mean can fall between

Figure 6: A sample of the actual and predicted outputs using the MDN on the development set

based on the range of the true data. This can be interpreted as the estimated range in which a future prediction will fall given the true observations that have already been observed within some level of confidence, also 95% in this case.

Table 4: Statistical Results

	% In-band	Band Width
Statistical Prediction Interval	91.98	22.33
Prediction Confidence Interval	76.83	12.49

It is interesting to note the performance of the model on each heat exchanger along the path of the flue gas. The economiser prediction, including the predicted standard deviation is less precise than the furnace. It is also noteworthy that the statistical prediction interval is substantially wider than the model’s predicted confidence interval.

5 Conclusion

In conclusion, a methodology to use an MDN to predict the heat transfer coefficients of various components in a natural convection boiler was presented. The model had an overall accuracy of 93% , with 91.98% of the measured data falling within the confidence interval and 76.83% of the mean prediction falling within the prediction interval. Given that the confidence interval is wider than the prediction interval, it can be concluded that the model requires longer training and finer tuning to improve the precision of the standard deviation parameter. Overall the model was able to capture plant behaviour and showed distinct trends in the data.

References

[1] Akash Singh, Vivek Sharma, Siddhant Mittal, Gopesh Pandey, Deepa Mudgal, Pallav Gupta, International Journal of Industrial Chemistry **9**, (2018), 1 – 15

[2] Wu Haibo, Zhang Man, Lu Qinggang, Sun Yunkai, Journal of Thermal Science **21**, No.4 (2012), 368 - 376

[3] Serth, Robert W., Lestina, Thomas, *Process heat transfer : principles and applications* (Oxford, England ; Waltham, Massachusetts : Academic Press, 2014) 67

- [4] Zhang Yanguo, Li Qinghai and Zhou Hui, *Theory and Calculation of Heat Transfer in Furnaces* (San Diego: Elsevier Science Technology, 2016) 186
- [5] Yang Liu, Nam Dinh, Yohei Sato and Bojan Niceno, *Applied Thermal Engineering* **144**, 308-320 (2018)
- [6] Christopher M. Bishop, *Pattern Recognition and Machine Learning* (Springer Science+Business Media, LLC, 2006) Ch. 5.6
- [7] Preeti Manke, Sharad Tembhurne, *International Journal of Computer Science Issues* **9**, 520 - 526 (2012)
- [8] Di Hu, Chen Zhang, Tao Yang and Gang Chen, *Sensors* **20**, (2020)
- [9] Safarudin Gazali Herawan et al, *Soft Computing* **22**, 5955–5964 (2018)
- [10] Hosham Salim, Khalid Faisal and Raheel Jawad, *Applied Computational Intelligence and Soft Computing* **2018**, (2018)
- [11] Prabhas Hundi, Rouzbeh Shahsavari, *Applied Energy* **256**, (2020)
- [12] Ryno Laubscher, Pieter Rousseau, *Applied Thermal Engineering* 114057 **160**, (2019)
- [13] Ke-Lin Du, M. N. S. Swamy, *Neural Networks and Statistical Learning* (Springer, 2019)
- [14] Ling Liu, Yanzhao Wu, Wenqi Wei, Wenqi Cao, Semih Sahin, and Qi Zhang, *IEEE 38th International Conference on Distributed Computing Systems* **38** (2018)
- [15] Ian Goodfellow, *Deep Learning* (The MIT Press, Massachusetts, 2015) 308