ISIJ International, Vol. 64 (2024), No. 14, pp. 2008–2019
https://doi.org/10.2355/isijinternational.ISIJINT-2024-151

Artificial Neural Network Model for Estimating Mold Flux Melting Temperature

Carlo MAPELLI,[1)*] Davide MOMBELLI,[1)] Gianluca DALL'OSTO,[1)] Jung-Wook CHO,[2)] Nathalie GRUBER,[3)] Irmtraud MARSCHALL,[3)] Maïté CORNILLE,[4)] Marco ALLONI[5)] and Riccardo CARLI[5)]

1) Dipartimento di Meccanica, Politecnico di Milano, Via La Masa 1, 20156 Milano, Italy.
2) Controlling Solidification Process Laboratory (CSP), Pohang University of Science and Technology, Graduate Institute of Ferrous Technology (GIFT), 77 Chengamro, Namgu, Pohang, 37673, South Korea.
3) Chair of Ceramics, Montanuniversitat Leoben, Peter-Tunner-Straße 5, A-8700 Leoben, Austria.
4) ArcelorMittal Global R&D, Voie Romaine, BP30320, 57283 Maizières-lès-Metz Cedex, France.
5) Prosimet S.p.A., Via Rodi 10, 24040 Filago (BG), Italy.

(*Received May 8, 2024; Accepted October 21, 2024; Advance online published October 30, 2024; Published December 15, 2024*)

The accuracy of the current models for the calculation of the melting temperature of the mold flux shows that there is still room for improvement, given that their accuracy could not be satisfactory enough to keep up with the current industrial needs. In this work the use of artificial neural networks for data prediction is explored. The network acts as a "black box" capable to predict the melting temperature determined by complex physical interaction among the involved chemical species composing the flux. The network is trained by learning from real experimental data provided by different research groups through hot stage microscopy. The data was tested first within its respective batches and then tested as a single aggregate data batch. After testing and optimization of the networks' parameters, an acceptable level of accuracy was reached because the estimated melting temperatures point out an average error lower than 30 K if compared to measured data. This opens the possibility for the development of a standalone application that can be used for reference. In order to open the possibility for further improvements of this study the paper shares and makes public the values contained in the matrixes connecting the nodes of neural networks.

KEY WORDS: artificial neural network; mold powder; liquidus temperature; continuous casting.

1. Introduction

The melting behavior of the mold powders used in continuous casting has yet to be completely understood. The importance of the characterization of such behavior lies on the crucial role that mold powders plays on the quality and productivity of the continuous casting process by carrying out several tasks: improving lubrication, controlling mold heat transfer, absorbing non-metallic inclusions and preventing reoxidation.[1,2)] Furthermore, the melting behavior of mold powders is known to rule the refining ability of mold slag and the subsequent infiltration of mold slag film in the meniscus.[3)] Parameters such as the infiltration and consumption of the slag film could be set and controlled by using mold powders featured by a suitable melting behavior. Such a behavior could be achieved by understanding and quantifying a crucial melting behavior indicator: the melting temperature.[4)] In addition, from a technological point of view, the melting temperature is used as most important reference parameter for evaluating the flux behavior. Indeed, industrially, both producers and end users require specific formulation capable to melt the flux in a specific thermal range. However, the phenomena taking place during the melting process of commercial mold powders remains largely unknown as a consequence of their complexity and subsequent interactions among the ionic chemical species involved.[5–9)] Specifically, each of the chemical species present in the mold powders has a specific function to carry out, which implies complex interactions among their components. Moreover, the variability in composition, both in terms of quantity and presence of certain components, is rather vast considering that different chemical compositions of the mold powders are defined as a function of cast steel.[10–13)] Indeed, each mold powder producer has designed and developed its own chemical composition and ways of manufacturing these powders. The chemical complexity and

* Corresponding author: E-mail: carlo.mapelli@polimi.it

high variability of components of the mold powders makes the characterization of their melting temperature a daunting task. Even if the characterization of the mold powders' melting behavior is particularly difficult, a model providing this information could turn out as an advantageous tool. From a practical point of view, manufacturers could benefit from this information. Actually, a reliable model about mold powder could be designed on the basis of required physical properties without the need for a trial-and-error approach.

A reliable prediction tool must be built on the basis of experimental measurement. On the other hand, statistical reliability would only be obtained by means of a considerable amount of measurements, which are time-consuming and costly.[14–18] Other solutions like the use of thermodynamic commercial software approximates the melting temperature of the powders, but the downside of such an approach is associated with computation that assumes an equilibrium state and this approach has pointed out a high level un uncertainty featured by error up to 250 K.[19–21] Similarly, equations derived from linear regression approaches have been developed with the one most used proposed by Mills *et al.* (Eq. (1)).[22]

$$\begin{aligned} T_{liq}(K) = {} & 1191 + 11.4 \cdot SiO_2(mass\%) - 11.0 \cdot CaO(mass\%) \\ & + 4.2 \cdot Al_2O_3(mass\%) + 5.7 \cdot MgO(mass\%) \\ & - 10.1 \cdot Na_2O(mass\%) - 15.8 \cdot K_2O(mass\%) \\ & + 1.9 \cdot F(mass\%) + 8.3 \cdot Fe_2O_3(mass\%) \\ & + 11.6 \cdot MnO(mass\%) + 273 \end{aligned} \quad \ldots\ldots (1)$$

However, these equations are affected by significant uncertainty levels, implying usual errors that range from 30 to 100 K.[21] A higher level of accuracy is still sought after.

In this work, a new approach for foreseeing the melting temperature of mold flux is proposed based on the implementation and development of an artificial neural network (ANN). As the working principle, the neural network should be able to predict the melting temperature by setting the chemical composition of the powder as the input, being a universal calculation mechanism based on pattern recognition. An ANN consists of an input and an output layer connected by one or more hidden layers that have the task to transform the inputs into something that the output layer can use. ANN can process all kinds of data which is coded in numeric form. The data is inserted into the network via the input layer, transformed via the hidden layer(s) and finally scaled to the wanted outcome in the output layer. Even if the neural network approach has a long tradition in material science (it has been used for modelling several properties of steels, *i.e.*, impact toughness, martensite start, hot strength and grain growth)[23–27] there are only few attempts to use it for predicting the mold flux properties.[28–30]

Thus, starting from the optimization (in term of number of neurons in the hidden layer, number of hidden layers, data division and learning algorithm) of a neural network tested on a restricted training data set of mold powders (284 measurements),[31] the current paper extends the use of such a model to a larger data set (422 measurements). Although the number of available experimental measurements is not extremely large, the choice of a neural network-based model was preferred amongst other methods. For instance, Random Forests (RF) could be a feasible instrument to solve this kind of problem. RF are based on several interconnected decision trees (DTs) to achieve higher accuracy than a single one. Decision trees are easily interpretable along with linear models, rule-based models, and attention-based models. However, RF tend to sacrifice the intrinsic interpretability present in decision trees and this aspect is one of the most desirable qualities of DTs. Furthermore, in problems with multiple categorical variables, like the melting temperature of mold flux, random forest may not be able to increase the accuracy of the base learner. In addition, running random forest models on a large dataset has a potentially long training time. In addition, decision trees often show a lack of reliability in their application to new data. One reason behind this is their tendency to perfectly fit all samples in the training data. This results in poor application quality if the data is noisy or contains outliers.[32] When applied to regression problems, RF have the limitation that they cannot exceed the range of values of the target variable used in training. Thus, they perform poorly with data out of the original training range, *i.e.*, they cannot be used for data extrapolation.[33] In addition, RF often have little performance gain when a certain amount of data is reached, on the contrary of ANN, which usually benefit from large amounts of data and continuously improve their accuracy, privileging the choice of this latter. Indeed, one of the purposes of the present work is to build-up an architecture capable to be expanded and adapted as the available experimental information grows by time.

The main objective of the build-up network is to verify its capability of predicting the melting temperature on several commercial mold powder, both taken as single batch or as a combined batch, by stating a relationship between chemical composition and melting temperature even without a fully comprehension of phenomena that happen during the melting process. Such relationship can be established by feeding the network with real experimental data from which the network will "learn", recognize patterns and eventually be able to make its own predictions when facing new, unknown data. To achieve this goal, several parameters of the network's structure will be studied as well as their influence in the network's performance. Finally, similarly from similar papers,[30] the weight and bias inputs as well as the weight and bias of the hidden layer were declared.

2. Experimental Procedure

2.1. Data Acquisition

For the setup of the neural network, a set of data is needed for "training" the network. This data comes from actual experimental data obtained through experimental measurements on commercial mold fluxes. In the case of no apparent or obvious correlation between inputs and outputs, a large training dataset is suggested. Yet, the exact amount of data needed is unknown as every case has its peculiarities. As a rule of thumb, a large training dataset would be required in order to obtain more accurate results in its predictions by the model.[34] This aspect must be emphasized since the availability of data for this case could affect detrimentally the performance of the network in the long run even when

all the other parameters of the network had been optimized. However, for the neural network construction presented in this paper, based on only one hidden layer featured by a limited neurons number, the set of measurements adopted is considered enough for a good estimation and capable in avoid overfitting. For this experiment, three different batches of data coming from four different sources were employed. The batches of experimentally obtained data were referred to as

- Batch 1: 120 measurements, performed at Prosimet in Italy.
- Batch 2: 169 measurements, performed at the Pohang University of Science and Technology in South Korea and at the Montanuniversität in Leoben, Austria.
- Batch 3: 133 measurements, provided by Arcelor Mittal in France.
- Combined batch: all previous data batches were merged in the hope of obtaining a greater data sample that, in theory, would help to improve the network's accuracy.

The statistical analysis as well as the development of the networks were done both for each of the data batches as well as for the batch containing the combined data. For all the four datasets 70% of measurements was used for training while 30% for validation. The neurons number in the hidden layer was increased and optimized on the number of available measurements, but still keeping it at a low value to avoid any undesired effect.

2.2. Melting Temperature Measurements

To determine the mold flux melting temperature, all the four laboratories have used the same measurement technique that is heating microscopy, which is the preferred tool of analysis used in the metallurgical sector when the melting behavior of a certain material must be studied.[35] Direct observation could be considered as the most adequate approach since the complexity of these materials does not allow a theoretical approach to be established for the prediction of the melting behavior. In example, the replacement of some fluxes with others leads to melting temperature fluctuation from few degrees up to few tens of degree.[6] Commonly, most of the investigated mold powders are featured by a melting temperature higher than 1 350 K.[6]

In example, the first batch of measurements were performed by a Heating Microscope MISURA by Expert System Solutions MISURA 3.32HSM software. Part of the second batch was measured by a HESSE EM301 Heating Microscope through its proprietary EMI 2 software. Similar equipments were used at the other involved laboratories.

For the measurements, samples of different commercial powders have been used and underwent the same procedure according to DIN 51730 standard, *i.e.*, the powder was grounded in an agate mortar, then the powder has been pressed in a cylindrical piece of 2 mm × 1 mm and finally, the cylinder resting on an alumina plate has been inserted in the furnace of the heating microscope and heated at a fixed rate of 10 K/min under inert atmosphere.

The height of the cylinder was observed during the heating process and expressed as a percentage of the starting value. The software recorded the height of the samples at the sintering, softening and melting temperatures. Such temperatures are determined by the dimensional parameters (height, width and contact angle) of the sample projected by its silhouette as shown in **Fig. 1**.[36] As a function of the shape assumed by the silhouette of the samples several characteristic temperature can be distinguished (where H is the sample height and W the sample width):[35]

- sintering temperature: identified when there is a dimensional variation of 5% with respect to the original image obtained from the sample;
- softening temperature: the corners appear more rounded and there is a smoothing of the upper part of the walls of the sample;
- sphere temperature: the sample is formed almost entirely of liquid phases, and the shape of the sample is controlled by the surface tension. The edges of the sample become completely round, with H = W;
- hemisphere temperature: identified when the contact angle becomes approximately 90° (or even more). The sample is approximately hemispherical, with W = 2H;
- melting temperature: the sample has spread out such that its height is one-third that of the hemisphere temperature. It is assumed that the sample has completely liquified and has reached its melting point.

In addition, for random samples, cross-confirmation of the melting behavior was determined by comparing the HSM measured values with ones obtained by TG-DSC analysis. The difference between the two technique was averagely +/− 25 K.

2.3. Chemical Composition Measurements

The chemical composition of the mold powders depends on the properties required for the industrial process they are used in. Flux chemical composition has been already defined as the most important factors of influence on the melting temperature.[22] Sixteen (16) of the most representative and influential chemical species were taken into account (**Table 1**). The chemical composition of all the studied powders has been measured by X-ray fluorescence spectrometer.

However, special considerations were taken in order to further simplify the input data. A neural network would greatly benefit from a smaller number of inputs, enhancing its predicting capabilities. For instance, all volatile

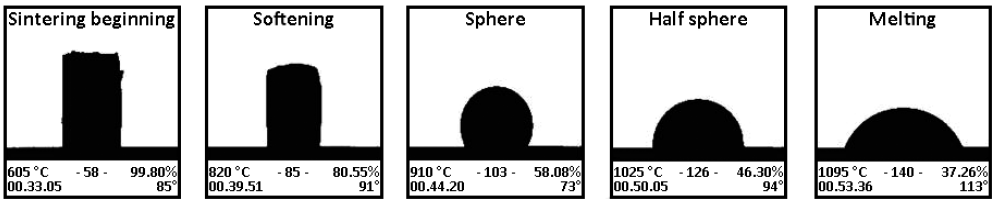

Fig. 1. Examples of softening and melting steps identified by the heating microscope. The images can be taken either at a specific time interval or at a specific temperature range. Adapted from.[36]

components were not considered for further calculations. Furthermore, the carbon was not considered either in its free form or as total amount. The purpose of carbon in the mold powders is that of regulating the melting rate, having no effect whatsoever on the final temperature of the powder.[6,37] Thus, it can be neglected. Also, Na_2O and K_2O were jointly considered as single component due to their similar behavior and role in the powder as fluxes and network breakers. More specifically, although it is well known that alkalis tend to volatilize in CaF–Na_2O–K_2O–SiO_2–CaO system and their ratio influences the melting and the viscosity of the flux, in this paper their contribution was unified due to their low concentration (mostly below the 10 wt.% and halves of the samples below 5 wt.%) that minimize to a negligible effect the variation of melting temperature.[38,39] Consequently, the final number of working inputs went from 16 to 13 (Table 1).

2.4. Neural Network Construction

The use and implementation of a neural network can be justified if no clear relationship is observed between inputs (chemical composition) and output (melting temperature). In the case of no apparent correlation and in the absence of any major statistical anomaly, the experiment carries on with the implementation of the neural network.[40]

A single hidden layer ANN was created using MATLAB version R2022b. Since this is a function approximation problem, the principles of feed forward nets are selected in order to build the model. A scheme of the designed network is illustrated in **Fig. 2**.[41]

Set variables were the transfer function, the training algorithm, the data division without a test dataset while the number of neurons in the hidden layer was optimized per each of the tested batch. All other settings (like number of epochs and learning rate) were left as default or optimized as necessary. The output node is a linear transfer function. Details for each aspect are described below.

A normalization of data has been performed because it allows a visualization of the distribution of data, helping to determine if the data can be said to have a normal distribution or if there are only a handful of inputs or data range to which most of the outputs can be attributed to. Normalization on input data will be computed as reported in Eq. (2),

$$x_{i,n} = \frac{x_i\,(mass\%) - x_{min}\,(mass\%)}{x_{max}\,(mass\%) - x_{min}\,(mass\%)} \quad \text{..... (2)}$$

where $x_{i,n}$ is the normalized value of one of the chemical component (*i.e.* SiO_2) for the *i*-sample, x_i is the actual value of the same chemical component of the *i*-sample, x_{min} and x_{max} are the minimum and maximum value of such chemical within the entire data batch.

Additionally, the normalized data will prevent an incorrect assignation of the weights and biases to the inputs over their apparent influence over the output. The melting temperature of the mold flux was instead log-normalize as reported in Eq. (3),

$$y_{i,n} = \log(1 + y_i) \quad \text{..... (3)}$$

where $y_{i,n}$ was the normalized value of the experimental melting temperature and y_i the actual experimental melting temperature.

The activation, or transfer function, that has been applied is the logistic sigmoid due to its good performance to describe system featured by significant non-linearity and because it transforms the values in the range 0 to 1, thus the signs of all output values will be same (*i.e.*, positive values as the melting temperature cannot be negative as well as the chemical composition) (Eq. (4)).[42]

$$f(x) = \frac{1}{1 + e^{-x}} \quad \text{..... (4)}$$

Table 1. A typical chemical composition of a mold powder sample used during the experimental measurements and relative interval of normalization. The visualized order is the same used for input nodes in the neural network building.

Component	Concentration [mass%]	Interval of normalization	Input order
$C_{tot.}$	8.34	–	–
C_{free}	6.27	–	–
SiO_2	37.43	0–54	1
Fe_2O_3	1.73	0–12	2
Al_2O_3	5.97	0–36	3
CaO	19.75	0–62	4
CaF_2	12.43	0–23	5
MgO	2.17	0–19	6
Na_2O+K_2O	6.24	0–17	7
MnO	0.00	0–11	8
S	0.09	0–5	–.
B_2O_3	0.00	0–7	9
Li_2O	0.00	0–5	10
ZrO_2	0.08	0–7	11
TiO_2	0.12	0–1	12
P_2O_5	0.062	0–1	13
T_{liq} [K]	1 458	1 000–1 750	

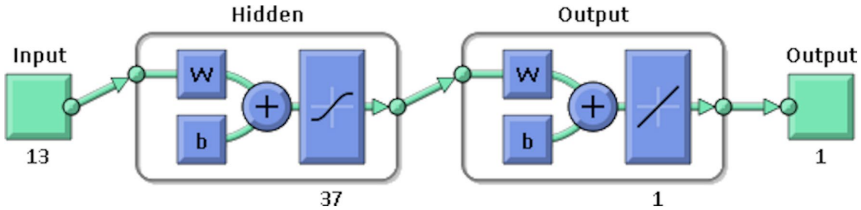

Fig. 2. The structure of the network consists of an input layer with 13 inputs (chemical composition), a hidden layer consisting of a variable number of neurons (*i.e.*, 37) with adjustable weights and biases and an output layer consisting of a single output (liquidus temperature). (Online version in color.)

Several parameters that directly affect the network's performance have been tested and optimized, exploiting the knowledge gained during the first exercise published in a previous paper.[31] Current ANN parameters include:

- training algorithm (Lavenberg-Martquardt);
- data division (70/30 training/validation);
- number of epochs (default = 1 000);
- learning rate (adapted on the base of dataset dimension);
- number of neurons in the hidden layer (optimized on the base of dataset dimension);
- error goal (measured by the Root Mean Square Error).

Finally, the performance of each of the iterations of the network will be measured according to the Root Mean Square Error (RMSE) obtained when comparing the network's predictions with the experimental data and this procedure has been fulfilled separately from the training trials in order to perform the validation of the model. RMSE is typically used for evaluating an ANN performances and it was chosen because it is more conservative than standard deviation.[43,44] It is typically used to measure the differences between experimental and predicted-by-a-model values. It is strongly and disproportionately influenced by the size of the squared error and it is sensitive to outliers. Thus, it is a responsive parameter to assess the goodness of the fitting. Minimization of RMSE was used as method for the optimization of number of neurons in the hidden layer, as this parameter is one of the most influential on the network's performance, and learning rate.

3. Results

3.1. Melting Temperature Data Analysis

Histograms were employed for the visualization of the data. The histograms representing the distribution of the melting temperatures for all the data batches are presented in **Fig. 3**.

As expected, the different origins of the mold powders resulted in a different distribution of the melting temperatures. All the batches showed an approximately normal distribution of the data, even if Batch 3 has pointed out a gap right in the middle of the bell curve. All the three initial data batches were either negatively or positively skewed, mainly due to the presence of outliers, which may be left out of the exercise as a way to improve the network performance. To better visualize the identified outliers, a boxplot was reported in **Fig. 4**. It is worthy to mention that all the batches have approximately the same mean and median value for the melting temperature and the symmetrical distribution of the box and the whiskers around the average demonstrates the normality of data distribution.

However, the unknown nature of the natural phenomena behind the melting process of the mold powders does not allow discarding these outliers. Thus, all the available data were used in the present work, leaving for the future the attempt to improve the ANN performances through outliers discharging.

Special attention must be given to the distribution of the combined batch. As shown in Fig. 3(d), even if the data shows a normal distribution, it is possible to observe the influence of each data batch. For example, the gap belonging

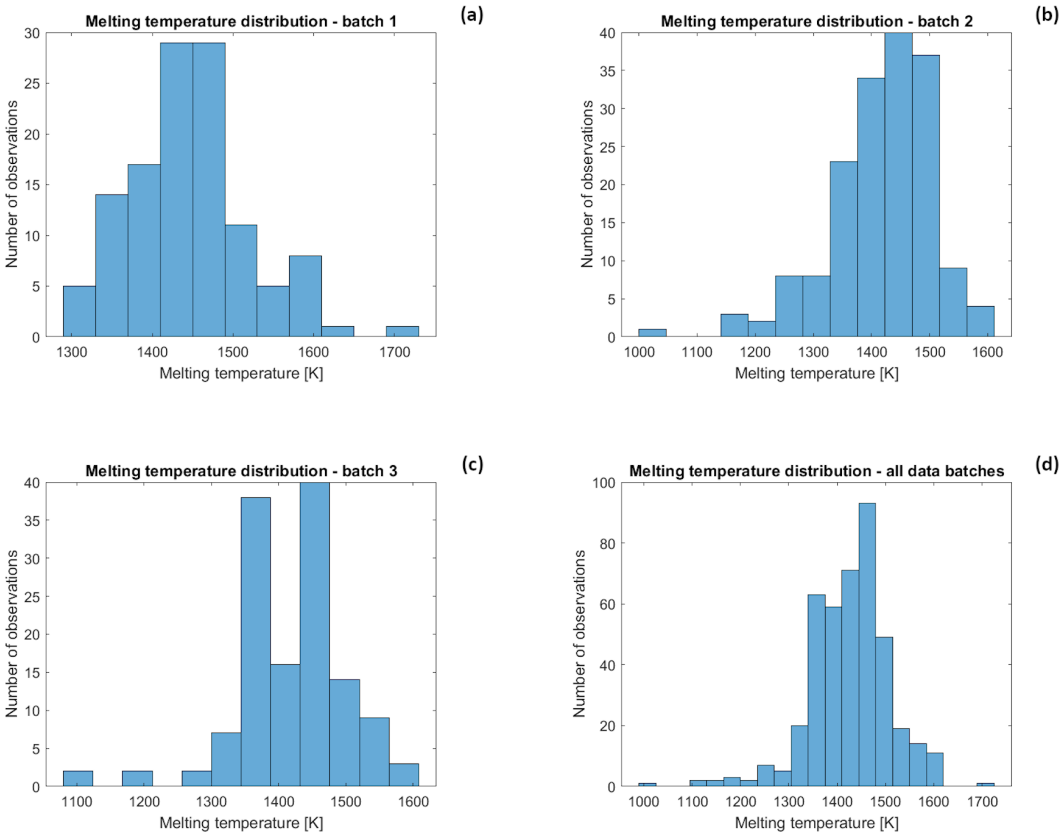

Fig. 3. Melting temperature distribution of Batch 1 (a); Batch 2 (b); Batch 3 (c) and combined batch (d). (Online version in color.)

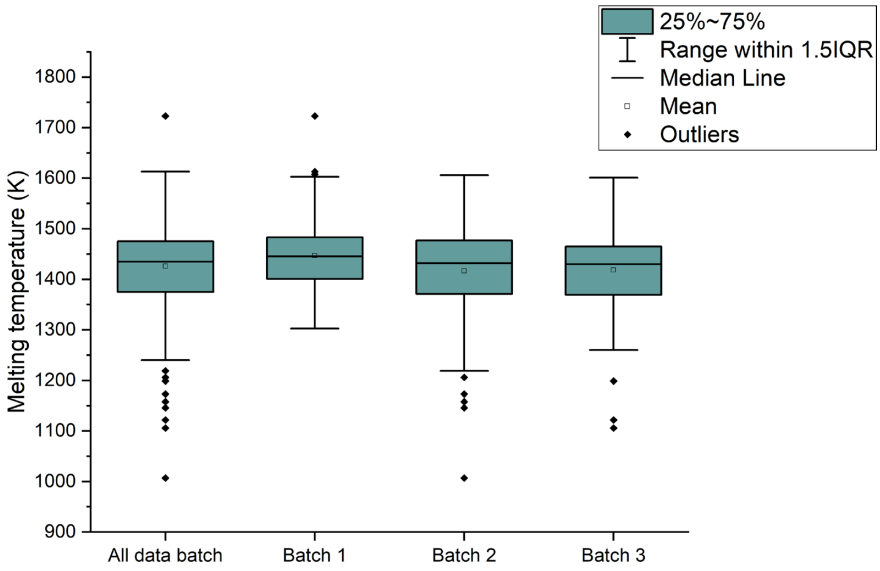

Fig. 4. Boxplot of melting temperature for the three batches plus the combined one. (Online version in color.)

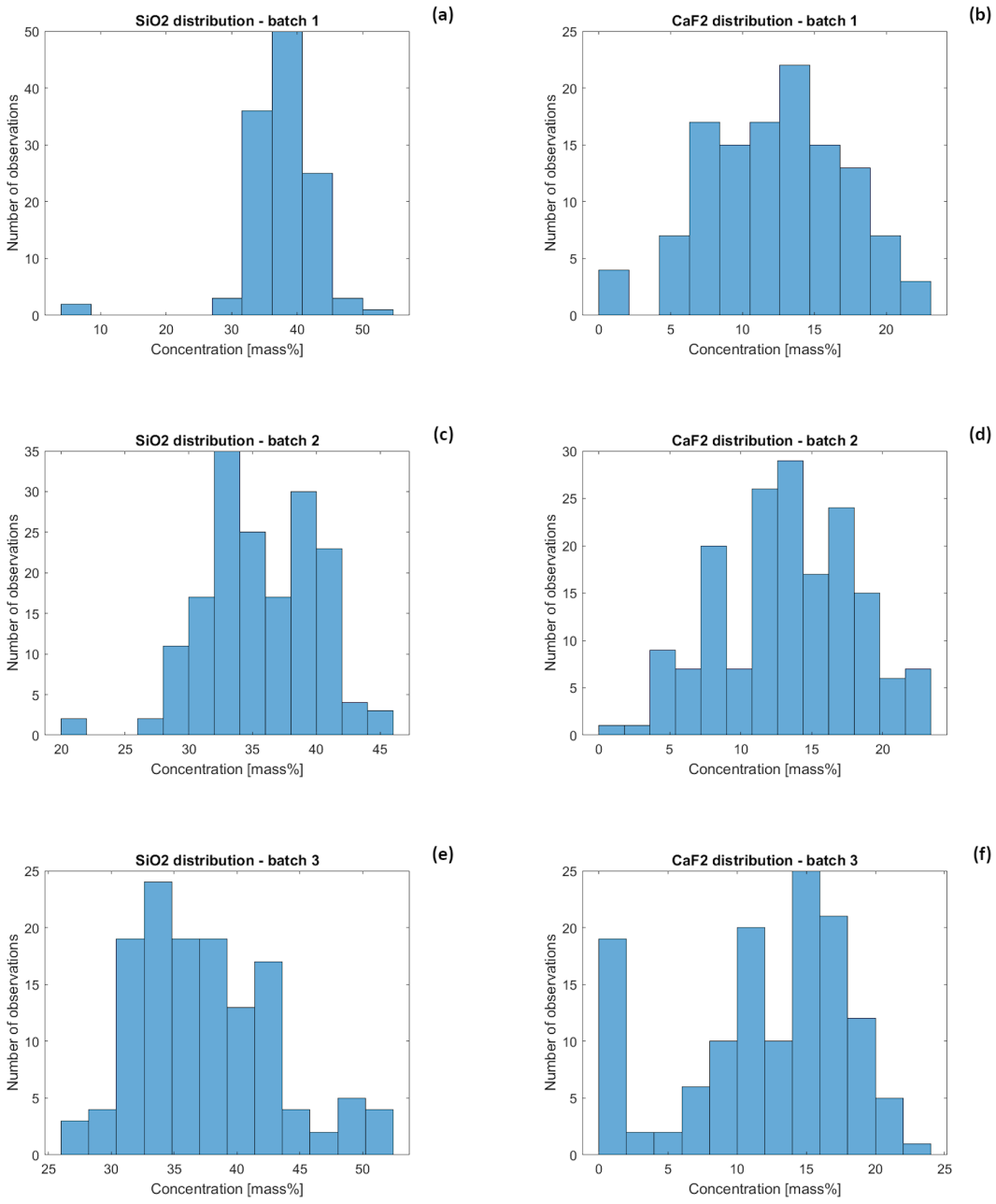

Fig. 5. SiO_2 and CaF_2 distribution of (a–b) Batch 1, (c–d) Batch 2 and (e–f) Batch 3. (Online version in color.)

to Batch 3 is still rather noticeable and so are the outliers belonging to batches 2 and 3, influencing on how skewed the data distribution resulted to be. From this visualization, it is already foreseeable that at the beginning of data processing the separation of each data batch from others would be the best course of action.

3.2. Chemical Species Distribution and Data Correlation

The several chemical species contained in the mold flux play a major role in defining the melting temperatures and their distribution. In particular, for the CaF_2–Na_2O–K_2O–SiO_2–CaO system the correct balance between SiO_2 and CaF_2 is fundamental to achieve the desired viscosity. Indeed, CaF_2 can easily replace the divalent oxygen ion resulting in breakdown of the Si–O network consequently decreasing the viscosity of the slag. This balancing is also effective in lowering the surface tension and melting range, while it causes an increase in solidification point (for concentrations over 5 wt.%) due to precipitation of complexes like cuspidine ($Ca_4Si_2O_7F_2$).[45] For this reason, the varied distribution of SiO_2 and CaF_2 for each batch have been pointed out because they are recognized as very significant chemical species to determine the melting point (**Fig. 5**). As for the melting temperature, their distributions are closer to a gaussian distribution. However, among the three batches, differences according to the data provider are markedly evident. The distributions, as well as the scatter plots (**Fig. 6**), depict an absence of a clear relationship between the two main chemical species and the melting temperature. This is demonstrated by the close-to-zero values of the R^2 when linear regression is applied to the experimental data (0.0002–0.0887). The only exception when the data shows a possible correlation is between CaF_2 and melting temperature for Batch 2 (0.256). However, also for this case, the R^2 coefficient is low enough to exclude even a weak linear correlation.

One of the main justifications for the use of a neural network for data prediction is that there should be no clear correlation between input(s) and output(s). If some sort of visible correlation existed, other regression models would be better suited for building the prediction model. As for CaF_2 and SiO_2, also all the other chemical components showed no visible correlation with the melting temperature. In particular, as the alkaline components are well known to have a direct influence on the powder physical properties and melting, the sum of alkaline species ($CaO+Na_2O+K_2O+MgO+Li_2O$) versus the melting temperature is shown in **Fig. 7**.[46,47] Since the sum of all the alkaline species can obscure their tendency to lower the melting temperature of the flux, the sum of the two major species (Na_2O+K_2O) is over imposed on the same graphs. Na_2O+K_2O versus melting temperature shows a moderate inverse linear correlation (R^2 = 0.38–0.57). However, this is not strong enough to neither explain the melting behavior of the flux nor explaining the complex relationships toward the other chemical species featuring the flux. Concluding, all the scatterplots (Figs. 6 and 7) unequivocally evidenced the absence of a clear correlation hence justifying the use of an ANN to solve the problem.

3.3. Neural Network Configuration and Optimization

In the designed shallow neural network model, the single most important parameter to be considered for the network is the number of neurons in the hidden layer. A network

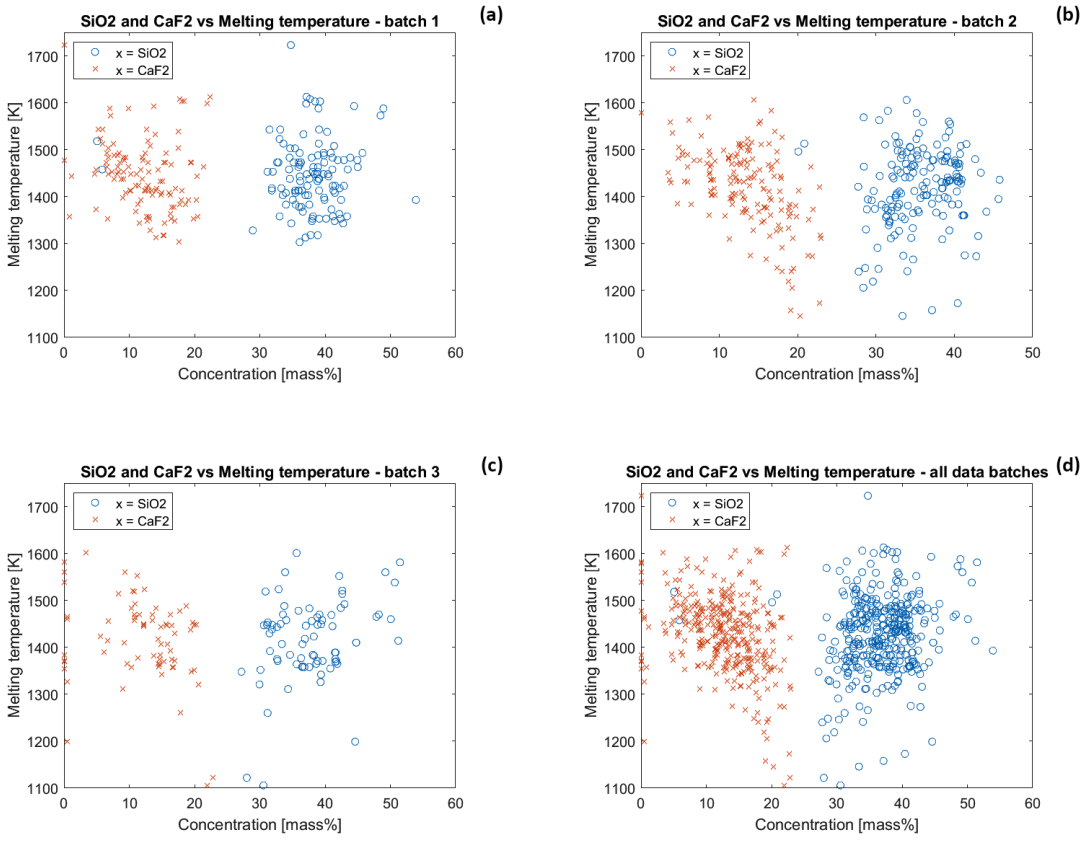

Fig. 6. Relations between SiO_2, CaF_2 and the melting temperature. (Online version in color.)

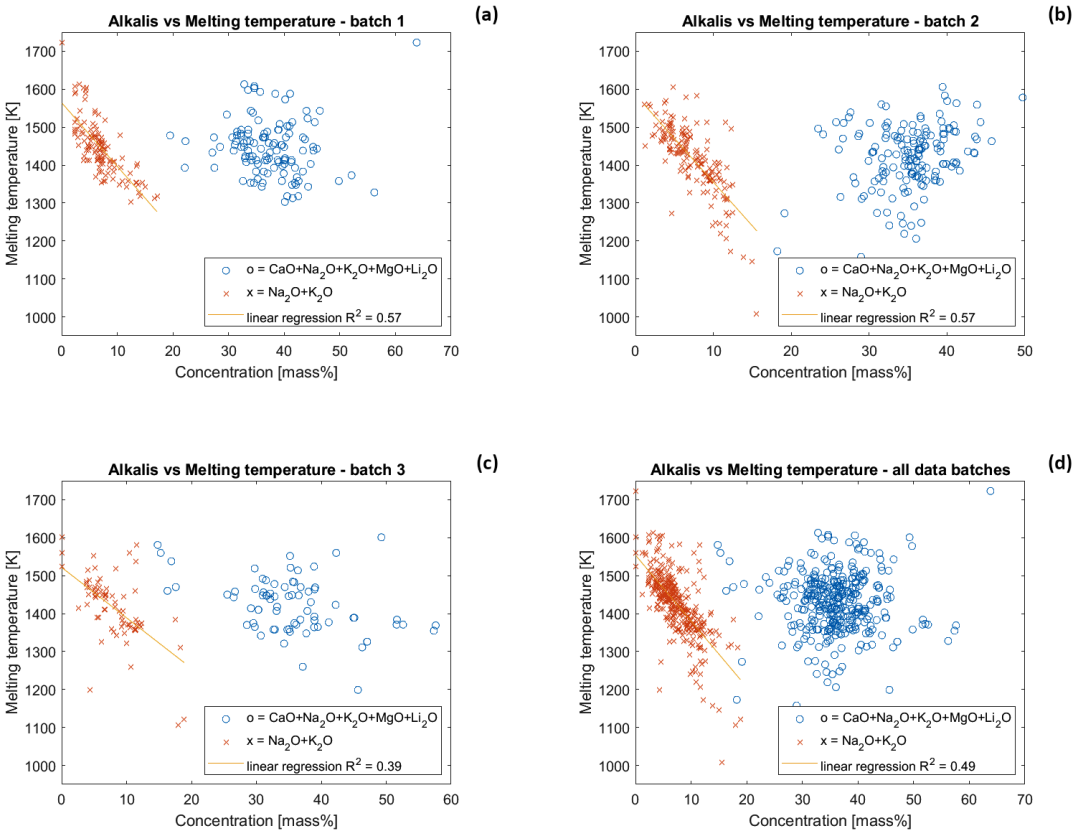

Fig. 7. Relations between the sum of all the alkaline species ($CaO+Na_2O+K_2O+MgO+Li_2O$) and (Na_2O+K_2O) only towards the melting temperature. (Online version in color.)

with a variable number of neurons was built for achieving an accurate visualization of the effect of this parameter.

For measuring the effect of the number of neurons, the model has been trained by a variable number of neurons going from 1 to a 100, with the performance being measured by means of the RMSE between the predictions of the network and the actual data. For the optimization of the networks, it was decided to divide the data set as 70% for training and 30% for validation.

The RMSE for both sets were plotted and compared (**Fig. 8**). As the size of the hidden layer increases the RMSE of training decreases until a specific number of neurons. After that point, it tends to stabilize. The more neurons in the hidden layer, the better the network will be "memorizing" the data used for training and, naturally, the smaller the error will be. This is less evident for the validation set, probably because only the 30% of measurement were used to verify the capability of the ANN to foresee the experimental melting temperature, leading to a higher RMSE variation along with the neurons increase. This can be explained by the fact that the more neurons the hidden layer has, the more complex the model becomes, meaning that the network has become overfitted. For the performance of the network, the main indicator of how well it is performing is the accuracy of the validation data set.

Based on the optimization exercise the number of neurons of hidden layer has been optimized and a configuration featured by the number of neurons that produced the lowest RMSE for the validation set has been selected. On this basis, the other parameters have been also optimized, keep the activation function, the learning algorithm and the data division the same already proved to minimize the RMSE,[31] resulting in the configurations shown in **Table 2** for the three separated batches.

To be specific, also the number of epochs has been tested. One epoch is when an entire dataset is passed forward and backward through the neural network only once. Since it is not possible to pass the entire dataset into the neural net at once, it must be divided into a certain number of batches. Thus, an epoch is made up of one or more batches and iterations is the number of batches needed to complete one epoch. However, in all trials and batches the number of epochs hardly overcomes 20–25 for minimizing the RMSE of validation set. In addition, even by forcing a high number of epochs, the improvement of the ANN model performances was negligible and stochastic. Thus, the routine was left free to auto set the epochs (*i.e.*, by leave this parameter to default - 1 000) without imposing a constrain. The same considerations can be stated for the learning rate (LR). In Batch 1 and 2 by diminishing the learning rate led to a marginal, non-significant and stochastic improvement of the ANN performances, while this was not true for Batch 3 and combined batch. Indeed, the best ANN performances (evaluated on the validation dataset) were obtained by setting a learning rate of 0.01. By the way, a further decreasing the learning rate of a factor of 10, the RMSE related to the training dataset reached practically zero while worsening the prediction of the validation dataset. Thus, LR was fixed to 0.01 privileging the minimization of error in the validation dataset only for those batches where it was significant.

In **Fig. 9** the graphic representations of the data correlation between the predicted data and the experimental data from both the validation and training have been pointed out and the revealed performance is certainly better than the

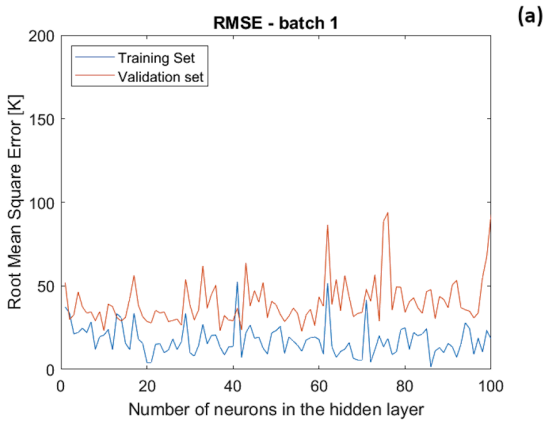

Fig. 8. RMSE of the training and validation sets according to the number of neurons in the hidden layer for (a) Batch 1, (b) Batch 2 and (c) Batch 3. (Online version in color.)

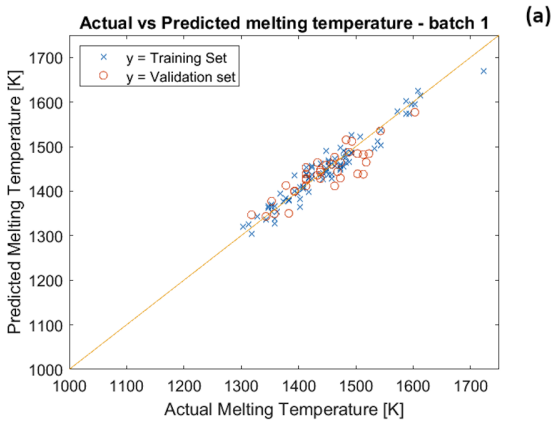

Fig. 9. Data correlation between inputs and outputs for (a) Batch 1, (b) Batch 2 and (c) Batch 3. (Online version in color.)

Table 2. Final set of parameters used for the optimization of the neural network model for Batch 1, Batch 2 and Batch 3.

Parameter	Batch 1	Batch 2	Batch 3
Number of hidden layers	1	1	1
Number of neurons per layer	6	3	14
Activation function	Logistic sigmoid	Logistic sigmoid	Logistic sigmoid
Learning algorithm	Lavenberg-Martquardt	Lavenberg-Martquardt	Lavenberg-Martquardt
Epochs	1 000 (default)	1 000 (default)	1 000 (default)
Learning rate	0.1 (default)	0.1 (default)	0.01
Training set size	70%	70%	70%
Validation set size	30%	30%	30%

approach based on exploitation of thermodynamic computation at equilibrium, whose average error is more than 100 K and the approach based on linear equation proposed by Shu *et al.* whose error ranges from 30 K up to 100 K.[19–21] The performance in terms of minimum RMSE of the optimized neural networks built separately for each of the data batches are reported in **Table 3**.

In order to state a more general model, the analysis of the combined batch (that has been built collecting in one batch all the data) has been performed to verify the influence of a larger data batch on the network attitude to learn. As fulfilled with the separated data batches, the model has been trained by a variable number of neurons that went from 1 to 100. The performance of the network has been measured by means of the RMSE of both the training data set and the validation data set. In **Fig. 10**(a) the comparison between RMSE for training and validation sets has been shown. Based on the optimization exercise about the number of neurons, a configuration with the 37 neurons in the hidden layer has been built because it has reached the lowest RMSE. On this basis, the rest of the parameters have been also optimized resulting in the configurations shown in **Table 4**. The comparison between the predicted data and the experimental data has been shown in Fig. 10(b). The performance of the neural network built for the combined batch shows as the best result obtained, goes as RMSE for the training set equal to 25.18 K and RMSE for the validation set at 28.66 K. The value of validation set in the combined batch is more reliable than the average error observed by one of most consolidated linear equation involving the effect of all the chemical species. To better visualize this last assumption, a linear regression model (without predictors interaction) (Fig. 10(c)) and the Mills equation were applied to the combined dataset (Fig. 10(d)). It is evident that both the linear regression and the Mills' equation overestimate the melting temperature of mold flux, leading also to a larger scattering of the predicted data if compared with

Table 3. RMSE for the training am validation sets for Batch 1, Batch 2 and Batch 3.

Parameter	Batch 1	Batch 2	Batch 3
RMSE for the training set	15.69 K	29.73 K	5.93 K
RMSE for the validation set	24.44 K	26.80 K	13.98 K

Table 4. Final set of parameters used for the optimization of the neural network model for the combined batch and related RMSE.

Parameter	Combined batch
Number of hidden layers	1
Number of neurons per layer	37
Activation function	Logistic sigmoid
Learning algorithm	Lavenberg-Martquardt
Epochs	1 000 (default)
Learning rate	0.01
Training set size	70%
Validation set size	30%
RMSE for the training set	25.18 K
RMSE for the validation set	28.66 K

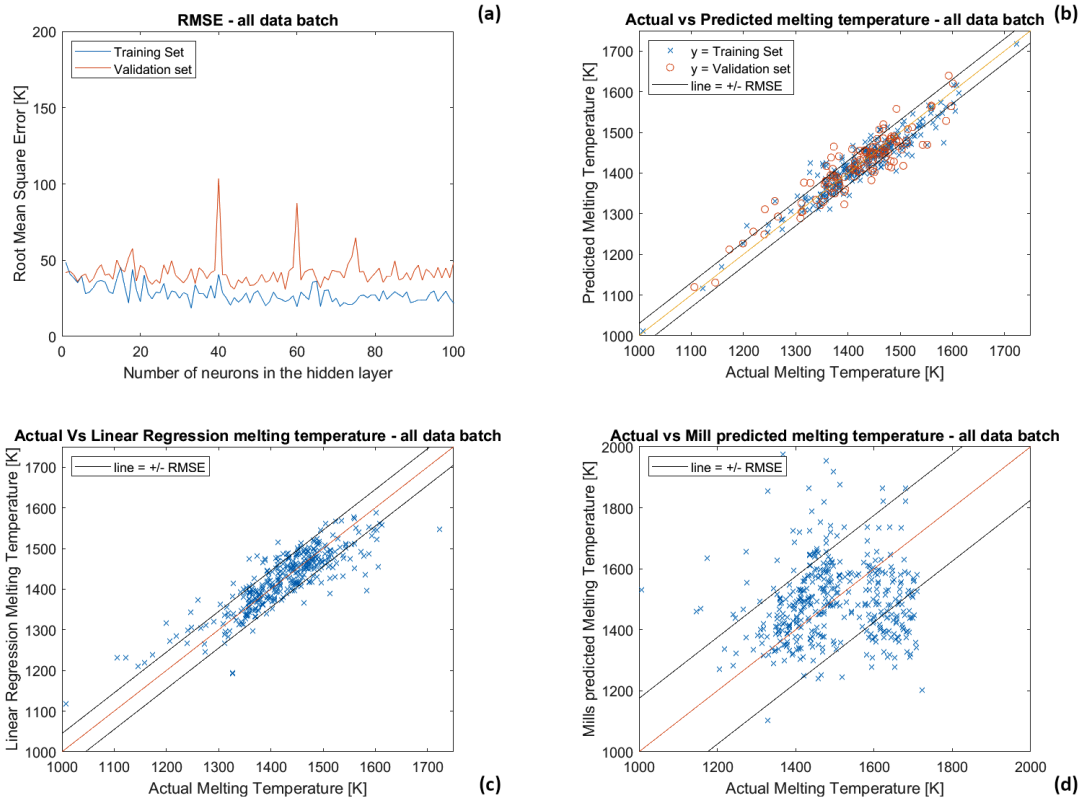

Fig. 10. RMSE of the training and validation sets according to the number of neurons in the hidden layer (a) and data correlation between inputs and outputs (b) for the combined batch. Comparison between actual and linear regression model (c) or Mills equation predicted values (d). (Online version in color.)

Weight Matrix Hidden Layer

0.2360	−1.5585	−0.1211	0.0372	0.7087	−1.5389	0.2307	−1.5070	1.1975	1.4485	1.0677	0.1522	−1.4902
1.7667	−1.1252	−0.2585	−1.0350	1.5375	1.3860	−0.3728	2.1234	−1.0117	−0.4918	−0.9533	−0.6784	0.3493
1.2590	−0.6553	−1.6127	−0.1260	−0.7074	1.9582	0.7114	0.1992	0.5019	1.1370	−1.6914	0.9189	−0.4584
−0.3507	1.1879	−0.6279	−0.6836	1.1427	−1.4719	1.5645	−1.4027	−0.7810	0.4898	0.4283	−0.9100	1.2981
0.8716	0.0427	−0.2853	1.3381	1.0599	−1.8703	−1.0212	0.8923	1.1066	0.9622	0.5875	1.0715	−1.2165
−0.6384	−1.4531	−1.0112	1.1652	0.3177	−0.8283	−0.0571	−0.0665	−1.1764	1.5360	−0.7528	−1.4408	1.2006
−0.0586	−1.1652	−1.4746	0.3422	−0.7010	−0.9778	−1.3565	0.7163	−1.3243	0.7064	−0.8192	−1.1535	−1.4234
−0.0212	0.7783	−0.9407	0.3011	1.7257	−0.8975	0.9538	−1.4865	−1.3464	−0.5506	0.9485	−1.0052	1.4458
0.8241	−1.0735	−1.3109	0.1996	1.7167	−0.2745	−0.3342	1.7178	0.5134	−0.6485	0.3142	1.1639	1.3565
0.1879	−1.5925	−0.6465	1.5395	−0.6000	−1.5811	−0.7157	0.9824	−1.1447	−0.4438	−1.0917	−1.4177	−0.0739
0.5942	1.6295	1.0461	1.0977	0.2591	0.4553	0.1288	0.6135	−0.3849	−1.3302	−0.1289	−2.0103	1.3444
−1.7117	−1.5757	−0.1392	−0.2012	−1.1176	−0.4747	−0.7842	−0.0524	0.2010	−2.0412	0.7229	−0.9871	0.8254
0.3660	1.1037	1.4442	−1.0168	1.4042	0.5472	−1.3901	−0.7863	1.3952	−0.0610	−0.4677	0.3498	1.4680
1.3123	1.4068	0.1916	1.5343	−0.5182	−0.2536	−1.3941	0.3681	−0.1043	−1.4104	1.1562	−1.2149	−0.5126
−1.0746	−0.9055	0.0837	−0.9091	−0.2249	1.5692	1.5268	1.2209	1.1428	−1.1655	1.1856	0.4536	0.9984
−0.9924	−1.2337	1.4486	0.8459	0.5539	1.5995	1.3404	−0.9639	0.7831	0.8240	−0.0977	−1.3236	0.4927
0.4545	0.6364	1.1033	0.2074	−1.0069	−0.4925	1.4652	1.5902	1.7912	0.1291	−0.4014	−1.2377	−0.4282
0.3960	−0.5714	−1.5817	0.2520	−1.7157	−2.1777	0.3057	1.2341	0.9196	−0.2188	−0.6817	−1.2985	1.1411
0.9436	1.0657	−0.8548	−1.4619	1.4275	0.4703	1.1332	−0.9125	−1.5127	0.0037	0.0989	0.1283	−1.4427
0.6048	−0.9178	1.1480	−0.9211	−0.8994	0.7212	1.0909	0.6185	0.8690	−2.4306	−0.2173	0.4958	0.8382
−0.9916	−1.0276	−0.1431	0.5229	−0.6706	1.5370	−0.1005	−2.0737	0.7806	1.5700	−0.0243	−1.1770	−1.0387
−1.6916	1.1777	−0.8377	0.8127	0.2793	1.9843	0.7442	0.1440	−1.1491	0.2327	0.1474	1.7590	0.5947
−0.6747	1.3739	−1.4704	0.6738	0.6931	1.3964	1.4443	−0.2179	1.2041	0.7915	0.1493	−1.2642	1.2460
1.5762	0.0541	−0.0782	−0.4465	0.5545	1.1552	0.6024	−0.9344	−0.7440	1.4838	0.1596	−1.4351	−1.4876
0.9653	1.0919	−1.1698	−0.3708	−0.8662	0.7097	1.5364	−0.9639	1.3452	−2.3365	−0.5414	−0.0115	0.2074
0.4676	0.6595	−0.5158	−1.5090	−1.1431	1.0440	1.1129	−1.4537	0.8729	−0.9323	−1.0959	−0.6251	−1.4209
−0.6896	−1.4677	0.1171	0.7055	0.6287	1.1035	0.6875	−1.2037	−1.4516	1.6511	1.2906	0.6036	−1.0145
−1.0323	−0.5859	−0.1995	−1.7882	0.6156	1.3342	−0.6974	−1.2369	0.6930	1.1743	−1.3661	1.2334	−0.2971
−0.2877	1.7165	1.2517	1.0325	1.4535	0.5707	−1.1294	−0.0697	0.5843	0.2248	1.2484	1.4398	−0.6767
−0.1891	−0.4361	1.5151	1.3194	−0.2062	0.4567	0.2862	−1.2933	0.6184	0.7330	−0.7427	1.5082	1.9033
0.8532	0.7754	−0.4552	−0.8211	−0.5468	−2.0381	−0.0985	1.1940	−1.6931	−0.0363	1.4121	1.7575	−0.0448
0.0773	0.5955	−1.4112	−1.9560	−0.5863	−0.9350	1.2626	−1.2377	1.6591	−1.3947	−0.3062	1.5666	−0.2754
−1.1007	0.9909	0.8945	−1.0204	−1.0722	0.3662	0.5357	1.4440	−1.3985	0.5430	−1.6444	0.8627	−1.1262
1.0743	−0.1100	1.3508	0.6093	0.9057	−1.1659	−0.1714	0.8412	−1.6439	0.3414	1.3085	1.1283	1.6807
−1.1849	1.5960	0.3836	0.3546	−0.6083	0.6286	1.5623	−1.3188	−0.3524	1.3641	0.7996	1.3616	−0.0491
1.3436	−1.2610	1.3476	0.6623	0.3231	−1.3200	−0.5505	0.2366	1.3638	−2.1971	1.6632	1.3101	1.5803
1.5658	−1.5896	1.2193	0.8118	1.5207	0.2596	−0.0068	0.8101	−1.5385	0.1503	0.7265	−0.0466	−0.6965

Biases Hidden Layer

−3.5341
−3.3327
−3.2651
2.8769
−2.8003
2.6910
2.5236
2.1897
−2.0194
−1.8162
−1.6720
1.4577
−1.2073
−1.0027
0.4587
0.3319
−0.3972
−0.2497
0.0227
0.4669
−0.4155
−0.7024
−0.8991
1.1269
1.3585
1.3401
−1.6006
−1.9760
−2.1248
−2.2246
2.6642
2.3226
−2.6650
3.3700
−3.2785
3.1415
3.6322

Weight Matrix Output Layer

−0.8010
0.3479
0.8331
0.6486
0.6287
−0.0605
−0.9411
0.6923
0.1334
0.3356
0.5010
−1.0084
−0.4307
−0.1572
−0.7710
0.6478
0.4401
−0.5044
−0.0359
−0.7710
0.6105
−0.9179
−0.4487
−0.1447
0.7363
0.5266
−0.6454
0.5609
0.5918
−0.4879
1.0267
−1.7189
1.1221
−1.1594
0.7631
1.7445
0.4301

Bias Outuput Layer

−0.2442

Fig. 11. Weights and biases for the optimized artificial neural network of the combined batch.

the ANN results. Furthermore, the Mills' equation resulted in a larger RMSE (+/− 173.58 K) than the linear regression model (+/− 43.88 K). This is due to the fact that the linear regression model is obtained from the current combined data set while the Mills equation was derived from a totally different data set. This strongly highlight once again the limitation of a classical statistical approach to solve a complex problem like the foreseen of flux melting temperature, *i.e.*, the incapacity to extend the empirical model for a different data set and its intrinsic inability to learn and adapt from new input data. Furthermore, the linear regression increases the number of bad predicted values, especially at the extremity of the experimental melting temperature range, by overestimating and underestimating the prediction at low and high temperatures, respectively (Fig. 10(c)). On the contrary, the ANN model keep the predictions withing the +/− RMSE along the whole measured melting temperature range, by also minimizing the number of predictions with a value higher than the mean error (Fig. 10(b)). Finally, it is worth mentioning that during the first iteration of the linear regression modelling, the only two predictors with a p-value higher than 0.05, and therefore not statistically significant, were "Na_2O+K_2O" and ZrO_2, despite the obvious contribution of the former to the melting temperature of the flux (Fig. 7). Consequently, following the backward-elimination rule, the least significative predictor (ZrO_2) was removed, which resulted in the significance of all the remaining chemical species. By reducing the number of regressors from 13 to 12, the necessity to simplify the problem by excluding a chemical species that may affect the melting temperature is highlighted, further to lead to a model that has a mean error (+/− 44 K) higher than the experimental evaluation (+/− 25 K), *i.e.*, not useful to avoid long and energy consuming measurements.

On the basis of optimization procedure, the structure of the network has been pointed out (**Fig. 11**) and the weights and biases that corresponding to the combined batch network have been reported to allow even other researchers to improve the model starting from such a basis. In particular, the first matrix (37 rows x 13 columns) depicts the weight assigned to each of the 37 neurons in the hidden layer per each of the 13 input parameters (chemical species). Those numbers are the interconnections between neurons and inputs and therefore do not have a real physical significance. The second matrix represents the biases assigned to each of the 37 neurons in the hidden layer. It serves to correct what the hidden layer cannot calculate against the input. The third and fourth matrixes, respectively, represent the connection between the hidden layer and the output layer and its bias. In other words, they represent the argument of the activation function, in this case logistic sigmoid (4), and its intercept. From the complexity of the matrixes reported in Fig. 11, the non-linearity of the problem is well highlighted. This means that a linear regression model, like the approach proposed by Mills (1), can be a limited instruments if extended to any unknown database, since it works well on the training dataset but fails in extrapolation if the new data are too far from the training itself. Similarly, when a linear regression model is applied to a specific dataset, it lacks in correctly estimate the extreme values of the experimental range and it suffer of lack of flexibility. On the contrary, an ANN can be more effective to extrapolate the melting temperature from a new dataset even far from the training and validation datasets.

4. Conclusions

The use of artificial networks for the prediction of the melting temperature by using only the chemical composition of such powder has proved to be promising:

- the designed shallow neural network has been pointed out as an efficient tool to associate the chemical species composing the flux with its melting temperature;

• the use of logistic sigmoid function as the activation function points out good performances;

• when 13 chemical species are taken into account a hidden layer composed by 37 neurons is an optimal choice and this statement has been proved by the trials performed for the combined batch that is the most significant group of collected data;

• the Root Mean Square Error of neural networks trained on combined batch is 28.66 K that is lower than the error performed by approaches based on linear equations or on thermodynamic computation at equilibrium;

• the matrix of weight ruling the neural network for the combined batch has been made public.

Conflict of Interest Declaration

The authors declare that they have no conflict of interest.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Authors Statement

Jung-Wook Cho, Nathalie Gruber, Irmtraud Marschall, Maïté Cornille, Marco Alloni, Riccardo Carli: measurements, data curation; Carlo Mapelli: conceptualization, supervision, methodology, writing – original draft; Davide Mombelli, Gianluca Dall'Osto: formal analysis, investigation; visualization, writing - original draft, review and editing.

REFERENCES

1) R.B. Mahapatra, J.K. Brimacombe, I.V. Samarasekera, N. Walker, E.A. Paterson and J.D. Young: *Metall. Trans. B.*, **22** (1991), 861. https://doi.org/10.1007/BF02651163
2) A. Kamaraj, A. Dash, P. Murugaiyan and S. Misra: *Metall. Mater. Trans. B Process Metall. Mater. Process. Sci.*, **51** (2020), 2159. https://doi.org/10.1007/s11663-020-01906-9
3) H. Jin, X. Wang, Y. Wang, S. Wang, X. Zhang and S. He: *J. Iron Steel Res. Int.*, **30** (2022), 465. https://doi.org/10.1007/s42243-022-00874-5
4) Y. Li, Y. Han, Y. Sun, P. Gao, Y. Li and G. Gong: *Minerals*, **8** (2018), 177. https://doi.org/10.3390/min8050177
5) I. Marschall, N. Kölbl, H. Harmuth and C. Atzenhofer: *J. Iron Steel Res. Int.*, **26** (2019), 403. https://doi.org/10.1007/s42243-019-00254-6
6) K.C. Mills and C.Å. Däcker: The casting powders book, Springer International Publishing, Cham, (2017), 4. https://doi.org/10.1007/978-3-319-53616-3
7) K.C. Mills and A.B. Fox: *ISIJ Int.*, **43** (2003), 1479. https://doi.org/10.2355/isijinternational.43.1479
8) S. Sridhar, K.C. Mills, O.D.C. Afrange, H.P. Lörz and R. Carli: *Ironmak. Steelmak.*, **27** (2000), 238. https://doi.org/10.1179/030192300677534
9) T. Kishi, H. Takeuchi, M. Yamamiya, H. Tsuboi, T. Nakano and T. Ando: *Nippon Steel Tech. Rep.*, **34** (1987), 11.
10) P.R. Scheller: *Steel Res. Int.*, **81** (2010), 886. https://doi.org/10.1002/srin.201000185
11) Z. Hao, W. Chen and C. Lippold: *Metall. Mater. Trans. B Process Metall. Mater. Process. Sci.*, **41** (2010), 805. https://doi.org/10.1007/s11663-010-9372-6
12) Z.Q. Hao, W.Q. Chen, C. Lippold and H.X. Mao: *Guocheng Gongcheng Xuebao/The Chinese J. Process Eng.*, **9** (2009), 514.
13) J.Y. Hu, X.W. Liao, L.H. Feng, W. Kang and Y. Liu: *Metalurgija*, **61** (2022), 163.
14) T. Mukongo, P.C. Pistorius and A.M. Garbers-Craig: *Ironmak. Steelmak.*, **31** (2004), 135. https://doi.org/10.1179/030192304225011115
15) H. Todoroki, T. Ishii, K. Mizuno and A. Hongo: *Mater. Sci. Eng. A.*, **413**–**414** (2005), 121. https://doi.org/10.1016/j.msea.2005.08.181
16) J.A. Bothma and P.C. Pistorius: *Ironmak. Steelmak.*, **34** (2007), 513. https://doi.org/10.1179/174328107X203912
17) L. Sun, H. Wang, M. Jiang, Q. Lin, C. Liu and Y. Zou: *Adv. Mater. Res.*, **287**–**290** (2011), 227. https://doi.org/10.4028/www.scientific.net/AMR.287-290.227
18) W.J. Boettinger, U.R. Kattner, K.-W. Moon and J.H. Perepezko: Dta and Heat-Flux Dsc Measurements of Alloy Melting and Freezing, in: Methods for Phase Diagram Determination, Elsevier, Amsterdam, (2007), 151. https://doi.org/10.1016/b978-008044629-5/50005-7
19) C.W. Bale, E. Bélisle, P. Chartrand, S.A. Decterov, G. Eriksson, K. Hack, I.H. Jung, Y.B. Kang, J. Melançon, A.D. Pelton, C. Robelin and S. Petersen: *Calphad Comput. Coupling Phase Diagrams Thermochem.*, **33** (2009), 295. https://doi.org/10.1016/j.calphad.2008.09.009
20) B. Sundman, B. Jansson and J.O. Andersson: *Calphad*, **9** (1985), 153. https://doi.org/10.1016/0364-5916(85)90021-5
21) Q. Shu, Z. Wang and K. Chou: *High Temp. Mater. Process.*, **32** (2013), 281. https://doi.org/10.1515/htmp-2012-0063
22) K.C. Mills, A.B. Fox, Z. Li and R.P. Thackray: *Ironmak. Steelmak.*, **32** (2005), 26. https://doi.org/10.1179/174328105X15788
23) H.K.D.H. Bhadeshia: *ISIJ Int.*, **39** (1999), 966. https://doi.org/10.2355/isijinternational.39.966
24) D. Dunne, H. Tsuei and Z. Sterjovski: *ISIJ Int.*, **44** (2004), 1599. https://doi.org/10.2355/isijinternational.44.1599
25) C. Capdevila, F.G. Caballero and C. García De Andrés: *ISIJ Int.*, **42** (2002), 894. https://doi.org/10.2355/isijinternational.42.894
26) L.X. Kong and P.D. Hodgson: *ISIJ Int.*, **39** (1999), 991. https://doi.org/10.2355/isijinternational.39.991
27) Y.Y. Yang, D.A. Linkens, M. Mahfouf and A.J. Rose: *ISIJ Int.*, **43** (2003), 1040. https://doi.org/10.2355/isijinternational.43.1040
28) T. Watanabe, K. Omura, M. Konishi, S. Watanabe and K. Furukawa: *ISIJ Int.*, **39** (1999), 1053. https://doi.org/10.2355/isijinternational.39.1053
29) M. Hanao, M. Kawamoto, T. Tanaka and M. Nakamoto: *ISIJ Int.*, **46** (2006), 346. https://doi.org/10.2355/isijinternational.46.346
30) M.A. Duchesne, A.M. Bronsch, R.W. Hughes and P.J. Masset: *Fuel*, **114** (2013), 38. https://doi.org/10.1016/j.fuel.2012.03.010
31) M. Vargas Hernandez, C. Mapelli, J. Cho, N. Kölbl, I. Marschall, M. Alloni and R. Carli: *Metall. Ital.*, **114** (2022), 23.
32) P. Roßbach: Ger. Frankf. Sch. Financ. Manag., (2018), https://blog.frankfurt-school.de/wp-content/uploads/2018/10/Neural-Networks-vs-Random-Forests.pdf, (accessed 2024-04-30).
33) P. Ellis: Extrapolation is tough for trees!, http://freerangestats.info/blog/2016/12/10/extrapolation, (accessed 2024-04-30).
34) M. Bernico: The Data Question, https://towardsdatascience.com/the-data-question-b6a8b60dc934, (accessed 2024-04-30).
35) C. Venturelli: *Micros. Today.*, **19** (2011), 20. https://doi.org/10.1017/s1551929510001185
36) Expert Lab Service: Heating Microscope, https://www.expertlabservice.it/en/#heating-microscope, (accessed 2024-04-30).
37) F. Chen, G. Wen, P. Tang, L. Yu and F. Han: *J. Therm. Anal. Calorim.*, **147** (2022), 10965. https://doi.org/10.1007/s10973-022-11361-0
38) M.S. Seo and I. Sohn: *J. Am. Ceram. Soc.*, **102** (2019), 6275. https://doi.org/10.1111/jace.16456
39) M.S. Seo and I. Sohn: *J. Am. Ceram. Soc.*, **105** (2022), 6320. https://doi.org/10.1111/jace.18557
40) L. Hardesty: Explained: Neural networks, https://news.mit.edu/2017/explained-neural-networks-deep-learning-0414, (accessed 2024-04-30).
41) MathWorks: Fitnet (function fitting neural network), https://it.mathworks.com/help/releases/R2019b/deeplearning/ref/fitnet.html, (accessed 2024-04-30).
42) S. Sharma, S. Sharma and A. Athaiya: *Int. J. Eng. Appl. Sci. Technol.*, **04** (2020), 310. https://doi.org/10.33564/ijeast.2020.v04i12.054
43) S. Shahinfar, H. Mehrabani-Yeganeh, C. Lucas, A. Kalhor, M. Kazemian and K.A. Weigel: *Comput. Math. Methods Med.*, **2012** (2012), 1. https://doi.org/10.1155/2012/127130
44) K. Messaoud, I. Yahiaoui, A. Verroust-Blondet and F. Nashashibi: Procceding of IEEE Intelligent Transportation Systems Conference, ITSC 2019, IEEE, Auckland, (2019), 1813. https://doi.org/10.1109/ITSC.2019.8916887
45) R. Carli, C. Righi and A. Giacobbe: Proceeding of Steelmaking Conference, AIST, (2002), 157.
46) G. Wen, S. Sridhar, P. Tang, X. Qi and Y. Liu: *ISIJ Int.*, **47** (2007), 1117. https://doi.org/10.2355/isijinternational.47.1117
47) W. Wang, D. Cai and L. Zhang: *ISIJ Int.*, **58** (2018), 1957. https://doi.org/10.2355/isijinternational.ISIJINT-2018-232