Investigation of amorphous Al-alloys with neural network

Bodo Schmidt, Friedrich-Schiller-Universitaet Jena, Germany, e-mail: p5scbo@rz.uni-jena.de

Keywords: structure factor, partial distribution function, neural network

Abstract

This work is intended to demonstrate the application of neural networks in material science. A specific example is the problem of the structure factor. The network is used in order to predict the partial distribution function form given intensity measurements. Afterwards it is possible to make statements about the interatomic distances and the coordination numbers.

1 Introduction

In recent years neural networks developed very fast. In computer science the theoretical background was investigated. Various applications were found in many fields: in engineering or industrial production as well as in science. The ability to adapt from a known problem to a similar one distinguish a neural network to other algorithms. In the first part an introduction to the theory of the application is given. In the second part the used network is discussed.

2 The atomic structure

The question of the formability of metallic glasses has been discussed in many papers recently. There is a variety of influencing parameters on the creation process (e.g. combination of elements, concentration, procedure of production etc.) so that metallic glasses as a result are more or less determined by chance. In the past there were a lot of works to make up criteria of the metallic glass forming that have to be met. But the experimentally produced glasses still showed deviations to these criteria [1]. Every approach however leads to the description on the atomic level. Every atomic model is intended to connect experimentally obtained data to structural properties. In order to check the evidence of the models one has to compare experimental and theoretical results. The experimental techniques often used to investigate the local arrangements of the atoms are X-ray and neutron scattering. The concept of the structure functions is a widely used method to describe information of the arrangement and distances of the atoms. However, experimental data are limited, so it is important to retrieve as much information as possible from single measurements.

Measuring the coherently scattered intensity $I_{\mathrm{coh}}(Q)$ one can deduct to the total structure fac $S^{\mathrm{FZ}}(Q)$ after [2]:

$$S^{\mathrm{FZ}}(Q) = \frac{I_{\mathrm{coh}}(Q) - c_i c_j (b_i - b_j)^2}{\langle b \rangle^2}$$

using c_i as the concentration of component i, b_i as the scattering length of component i a $\langle b \rangle = c_1 b_1 + c_2 b_2$ (j follows respectively of course). For the parameter Q that describes t momentum transfer the following relation is held

$$Q = 4\pi \frac{\sin \Theta}{\lambda}$$

The total structure factor $S^{\mathrm{FZ}}(Q)$ can be expressed as the sum of three weighted partial structu factors as follows:

$$S^{\mathrm{FZ}}(Q) = \underbrace{\frac{c_i^2 b_i^2}{\langle b \rangle^2}}_{W_{ii}} S_{ii}(Q) + \underbrace{\frac{2 c_i c_j b_i b_j}{\langle b \rangle^2}}_{W_{ij}} S_{ij}(Q) + \underbrace{\frac{c_j^2 b_j^2}{\langle b \rangle^2}}_{W_{jj}} S_{jj}(Q)$$

The factors W_{ii}, W_{ij} and W_{jj} contain the concentrations and scattering lengths because of t above definitions.

Carrying out the Fourier transformation of $S^{\mathrm{FZ}}(Q)$ leads to:

$$\varrho^{\mathrm{FZ}}(R) = \underbrace{\frac{c_i b_i^2}{\langle b \rangle^2}}_{W_{ii}^*} \varrho_{ii}(R) + \underbrace{\frac{2 c_i b_i b_j}{\langle b \rangle^2}}_{W_{ij}^*} \varrho_{ij}(R) + \underbrace{\frac{c_j b_j^2}{\langle b \rangle^2}}_{W_{jj}^*} \varrho_{jj}(R)$$

The partial distribution functions $\varrho_{ii}(R)$, $\varrho_{ij}(R)$ and $\varrho_{jj}(R)$ contain the interatomic distan as well as the coordination numbers. Often the partial reduced pair distribution functio plotted. The following equation join it with the partial distribution function:

$$G_{ij} = 4\pi R \frac{\varrho_{ij}(R)}{c_j} - \varrho_0$$

with ϱ_0 being the mean atomic number density.

For completeness it should be mentioned that there are other representations according Bhatia and Thornton [3].

Normally one need three linearly independent measurements of $S^{\mathrm{FZ}}(Q)$ respectively $\varrho^{\mathrm{FZ}}(R)$ order to determine $S_{ii}(Q)$, $S_{ij}(Q)$ and $S_{jj}(Q)$ respectively $\varrho_{ii}(R)$, $\varrho_{ij}(R)$ and $\varrho_{jj}(R)$. S method that gives hint to the 3 solutions with only one measurement of $S^{\mathrm{FZ}}(Q)$ respectiv $\varrho^{\mathrm{FZ}}(R)$ would be an advantage. It is not possible to trick mathematics so the word hint shou be stressed. Of course the solving of an equation system with less equations than variables impossible. But the physical theory behind the above equations define the solution too.

There are two possibilities for the training patterns of the net. For the first pattern we sta from 3 given function $\varrho_{ii}(R)$, $\varrho_{ij}(R)$ and $\varrho_{jj}(R)$. We can conclude to $\varrho^{\mathrm{FZ}}(R)$ and further to $I_{\mathrm{coh}}(Q)$. The second way starts with using known pairs of $I_{\mathrm{coh}}(Q)$ and the three par distribution functions. These patterns ($I_{\mathrm{coh}}(Q)$ as the input and $\varrho_{ii}(R)$, $\varrho_{ij}(R)$ and $\varrho_{jj}(R$ the output/training-pattern) are used in order to train the net.

In the application measured intensity curves are used as the input. The output pattern ca propagated by the network. Several examples will be demonstrated in the lecture.

3 Backpropagation algorithm

There exist a huge variety of neural networks and learning algorithms. The first question is which model is the right choice? Since we need a network that recognise patterns, conclude to a result and is able to develop a good choice is the backpropagation algorithm. In order to simplify the structure of the network we consider a feed-forward network (without feedbacks). This is a good assumption, since the input is restricted to the first layer. Furthermore the number of weights is reduced to the minimum possible.
All variables are used as in [4]. The brain consist of about 10^{11} nervous cells. These cells operate as little switch elements. They are connected via dentrids and axons in which dentrids lead to a cell and an axon leads away. The so called synapses work as a connecting component between the axons and the dentrids. The transmission is carried out using small electrical currents. Taken these facts into account we can model an artificial connection as drawn in Fig. 1. The calculation consist of the following steps, from the input a_i to the last output a_j. The

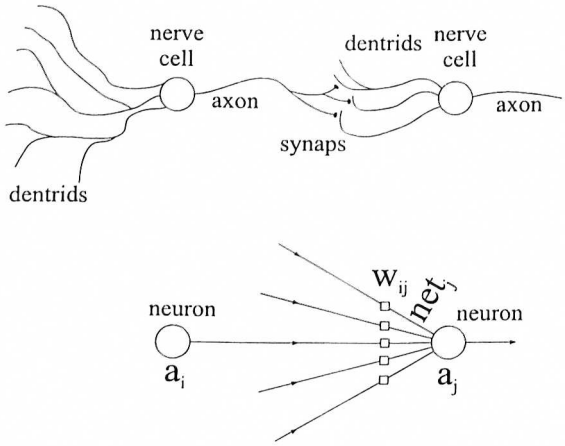

Figure 1: The adaption of a neuron from a nerv cell

nerve cells or in the artificial way the neurons contain the activation a_i. Every activation a_i is transferred to a weight w_{ij}. The first index represents the origin and the last index the target. All modified activations are summed up and used as the input net_j. The equation is defined as

$$net_j = \sum_i a_i w_{ij} \tag{6}$$

The net_j-values are *transferred* into the neuron-activation a_j using a so called activation function f_{act} as follows

$$a_j = f_{act}(net_j) \tag{7}$$

What kind of activation function is possible? Since the backpropagation algorithm needs the first derivation of f_{act} it should be a steady function. In order to simulate saturation a good choice is the sigmoidal Boltzmann function. A general definition is

$$f_{act}(x) = \frac{y_{min} - y_{max}}{1 + e^{\frac{x - x_0}{W}}} + y_{max} \tag{8}$$

The following variables were used: W as the width, x_0 as the turning point and $y_{\min}$ respectively $y_{\max}$ as the boundaries of the function. Taking these facts into account we are able to propagate a neural network completely.
In order to train the network onto a specific pattern t_j at the output layer we have to consider the learning algorithm. The weights as the only changeable parameters are modified as follows:

$$\Delta w_{ij} = \eta a_i \delta_j \tag{9}$$

with the following definition for δ_j:

$$\delta_j = \begin{cases} f_{\text{act}}^{-1}(net_j)(t_j - a_j) & \text{if } j \text{ is located in the output layer} \\ f_{\text{act}}^{-1}(net_j) \sum_k (\delta_k w_{jk}) & \text{if } j \text{ is located in the hidden layer} \end{cases} \tag{10}$$

The δ_j for a hidden layer uses the δ_j of the succeeding layer. η acts as a learning rate. A sketch of the structure of the whole network is given in Fig. 2.

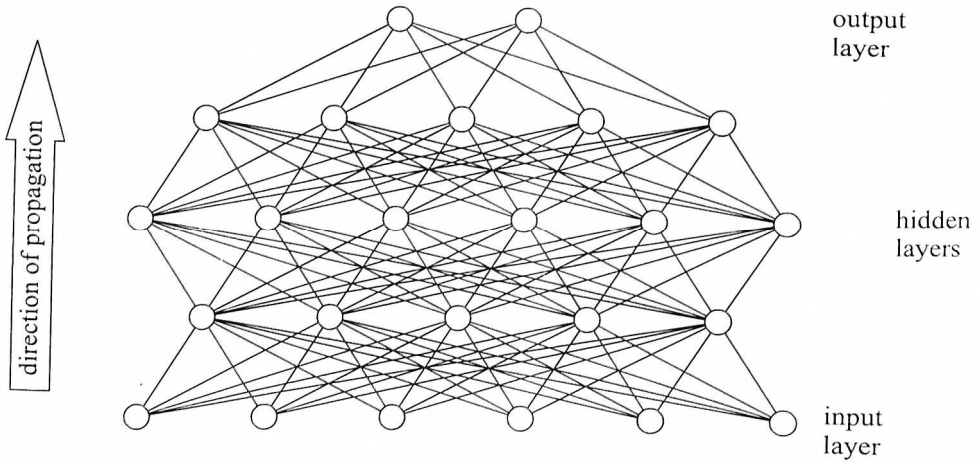

Figure 2: The structure of a multilayer network

References

[1] Egami, T: Journal of Non-Crystalline Solids 205-207 (1996) 575.

[2] Faber, T. E.; Ziman, J. M.: Philosophical Magazine 11 (1965) 153.

[3] Bhatia, A.; Thorton, D. E.: Physical Review B2 (1970) 3004.

[4] Zell, A.: Simulation Neuronaler Netze: Addison-Wesley, 1st ed. 1994.