LIP-STUDENTS-23-6

AI for thin film characterization using RBS

Bernardo Soares[1,a]

[1] *Faculdade de Ciências da Universidade de Lisboa, Lisboa, Portugal*

Project supervisor: Tomás Sousa, Francisco G. Barba e Ricardo Pires

October 10, 2023

Abstract. One of the methods employed for thin film characterization is Rutherford Backscattering Spectrometry (RBS), a potent analytical tool that lies on the detection of back-scattered ions from a sample surface, enabling researchers to investigate material composition and properties. However, analyzing RBS data can be a laborious and time-consuming process. The use of AI models can automate this process and make it useful for experimental scenarios. In this work, we show that Artificial Neural Networks are able to predict with good accuracy target's thickness and material based on simulated RBS spectra as training data. After processing the simulation data, we proved that the model can accurately predict gold targets developed directly in the laboratory. Additionally, we displayed how to automatically generate simulated data in the SIMNRA software.

Keywords: Artificial Intelligence (AI), Machine Learning (ML), Neural Networks, Rutherford Backscattering Spectrometry (RBS), SIMNRA

1 Introduction

During the past decades, Rutherford Backscattering Spectrometry (RBS) has become a crucial nuclear technique of Ion Beam Analysis (IBA). RBS success is due to being fully quantitative and depth-sensitive in a range of a few nm up to tens of μm, therefore being widely used for thin-film characterization with high-quality analysis [1–3].

Nonetheless, analyzing RBS data is a laborious and time-consuming process, even for an expert. It can also be challenging to maintain a high-quality standard analysis for big datasets due to fatigue. To bypass this, implementing a fitting procedure in batch mode to automate the analysis of large amounts of data is a solution. However, this method (or similar) are computationally expensive and can still prove to be inadequate to analyze great quantities of data [2, 4, 5].

Another reason that contributes to the poorness of this procedure is the fact that automatic fitting does not learn/take advantage of previous runs, therefore increasing its running time. Manual processing RBS spectra can be slow, but an individual might be able to learn some information from earlier spectra analysis, which can be exploited to process the subsequent samples. That being said, the implementation of machine learning algorithms to process RBS spectra can be an appropriate solution, especially Artificial Neural Networks (ANNs). [2].

Neural networks have cemented themselves as one of the strongest supervised learning techniques during the last decade. The basic unit of neural networks is the neuron, which takes a vector of N input features and produces a scalar output. A neural network comprises numerous neurons organized into sequential layers, where the information produced by one layer is utilized as input for the subsequent layer. The initial layer within the neural network is referred to as the input layer, while the middle layers are commonly known as hidden layers, and the last layer is denoted as the output layer [6]. Additionally, artificial neural networks are remarkably useful in performing on-line spectral analysis after being trained. During a RBS measurement, it can effectively assess the ongoing process, swiftly indicating whether it is proceeding as expected or encountering issues, which can be extremely useful for the scientists in the laboratory.

In this paper, we start by developing a way to automate simulated data acquisition. After that, we test and compare different neural network architectures. First, we test with simulated data and, afterwards, with experimental data, more specifically gold targets produced in the laboratory. Finally, we present the results for each neural network, make some conclusions and give suggestions of improvements to be done in the future.

2 Experimental procedure

2.1 Automation

Because of the lack of substantial experimental data, simulated data was employed in order to train the ANN. The acquisition of data was possible with the help of SIMNRA, which is a software responsible for generating RBS simulated data. SIMNRA allows to define a numerous amount of parameters, such as the setup and target, and after that simulate the RBS process, extracting the spectrum for each run.

Unfortunately, SIMNRA does not offer a feature that allows automatic extraction of large quantities of data, whereupon we need to develop a solution to this problem. To our luck, SIMNRA User Guide explains OLE automation: "SIMNRA is an OLE automation server, which allows other applications to control SIMNRA. All functions accessible through the user interface are also available through OLE functionality." [7]. Given that, we can use Python to automate spectra generation. By assigning each parameter to a random number withing a certain range, we can iterate achieving multiple spectra, each with different characteristics. The simulations were all performed under

[a] e-mail: soaresberna@gmail.com

the same circumstances, changing only the target element and its thickness.

2.2 Input and training data

We utilized Gold (Au), Tin (Sn), and Lead (Pb) as single-element targets and additionally, we employed Calcium Fluoride (CaF2) in a Carbon substrate and Aluminum (Al) in Tin (Sn) as multi-element targets. Each element has a unique range of thickness values. This approach made more sense instead of picking random targets because the ultimate goal of this study is to develop a tool that is useful for the team of scientists of NUC-RIA [8] using the Van de Graaff at CTN, represented in Figure 1 [9].

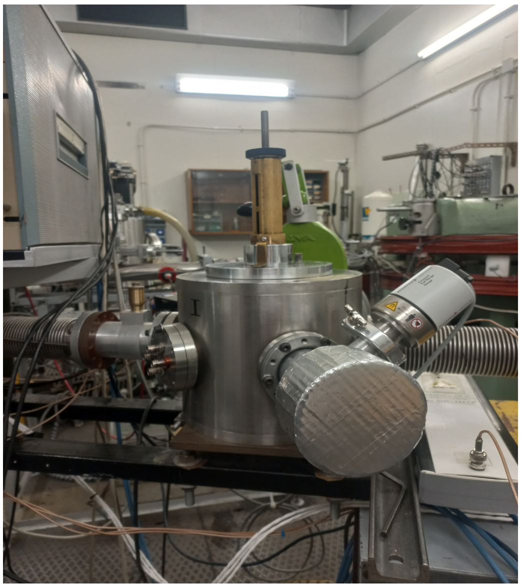


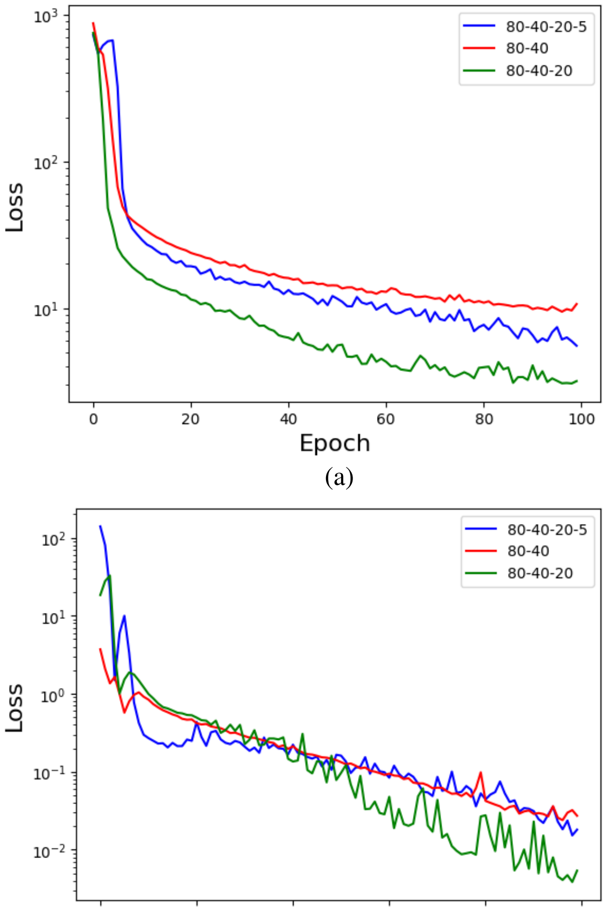
Figure 1: RBS chamber at CTN

All simulated spectra are histograms of counts with 2800 channels. The number of channels was a defined value for all spectra, allowing us to input each channel into an input node. This would not be possible with a variable number of channels.

2.3 Data cleaning and ANN architecture

After generating the simulated RBS spectra, data normalization was performed, which proved to achieve better performance in comparison with raw data or even standardized data, this is in accordance to literature procedures [5]. This could be explained by the disparity of counts in some spectra, where the peak value of counts in one can be much higher than in others, potentially impacting the ANN performance.

As stated previously, each channel is given as input in each node, meaning our input layer has 2800 nodes. We tested and compared several network architectures to determine the one that better suits our problem. The neural networks employed were only *Fully Connected Neural Networks (FCNN)*, which consist of a series of fully connected layers [10]. In our case, the only parameters that affect the performance of the neural network are the number of layers and neurons, because the activation function is the same for every node (except the input and output nodes), that being the *Rectified Linear unit (ReLu)*. In the output layer, we have a regression output node and multi classification output nodes. The regression output is responsible for predicting the thickness value, as it predicts continuous values. Its activation function is simply linear. On the other hand, the multi classification output is in charge of predicting the target's elements, therefore predicting only discrete values. Here, the activation function is *softmax*, where it outputs the elements that have more probability of being the real value. The multi classification output has as many nodes as target's elements, while the regression output has a single output node.



(a)

(b)

Figure 2: Loss function of the regression (a) and classification (b) outputs per epoch: (a) the loss metric used was the mean squared error (MSE). All the network architectures almost perform equally as good; (b) the loss metric used was the categorical cross entropy. All the network architectures almost perform equally as good.

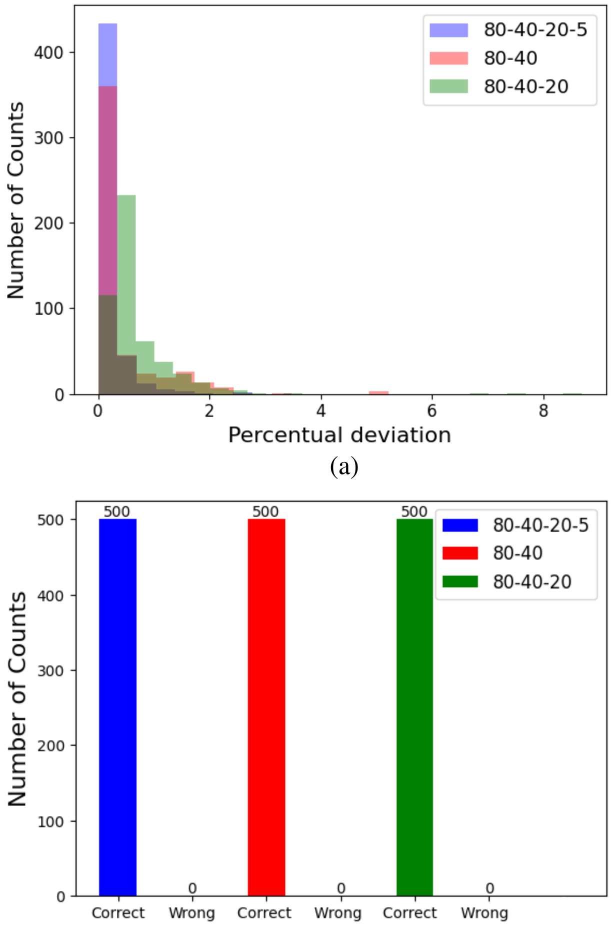
The optimizer used was the *Adaptive moment estimation (Adam)*, which is an extension of the stochastic gradient descent algorithm [11]. Early stopping was also implemented to prevent overfitting. In Figure 2, graphs are presented with the loss function of each neural network architecture for the regression and classification output, respectively. For a better context, the network 80-40-20-5, for example, means that there are 4 hidden layers, each having 80, 40, 20 and 5 neurons.

3 Results and discussion

In this section, the test results for simulated and experimental data are presented.

3.1 Simulated data

To make sure the model could learn RBS spectra, it was tested by giving with simulated spectra. The training and testing size consisted of 9500 and 500 samples, respectively. In Figure 3, a plot is provided with a histogram according to the percentage deviation of the thickness prediction to the real value and the number of counts of correct and wrong target's elements predictions, respectively. The thickness predictions appear to be highly accurate, being the greatest deviation about 5% of the real value for the thickness, while the elements predictions are always correct. This is a positive indication that the neural network architectures can accurately recognize spectral patterns and make precise predictions. Note that the tested data contains all types of materials and thicknesses. This means that, by looking at Figure 3, we can not differentiate if a count in 5% corresponds to a thick or thin target, therefore giving us only a rough measure of the neural networks performance.



(a)

(b)

Figure 3: Number of counts per percentual deviation of the thickness (a) and number of correct/wrong counts of the material prediction (b).

3.2 Experimental data

After testing with the simulated spectra, the model was tested with experimental data gathered at CTN. The first problem that arose is the fact that the training set has no noise introduced, while the testing set, which consists of experimental data, has noise in the spectra. As a result, the neural network was unable to make accurate predictions consistently throughout each simulation. To fix this issue, there was a need to treat the data before testing it.

The first approach was to maintain the main peak untouched and fit a quadratic regression to the spectra, after that subtract it to every channel. It was observed that the model still could not predict accurately the target element, every target was predicted to be a multi elemental one. A reason for this might be that all training spectra are simulated, which means there is a well outlined peak (or peaks for multi elemental targets), and the number of counts on other channels is equal to 0 (it has low background). Even when subtracting the quadratic fit to the experimental spectra, there is still a certain amount of counts on most channels. Because these channels do not have exactly 0 counts, the model might interpret these spectra as multi elemental targets.

In order to fix this, we tried a less orthodox solution: leave the peak unchanged and set the number of counts to 0 on every other channel. This way, the data gets much more similar to the simulated one, and it proved to work. In Figure 4, a plot of the simulated, raw experimental and clean experimental Gold RBS spectra with approximately the same thickness is shown.

We tested the model with 6 RBS spectra of Gold targets that were produced in the lab and, in Figure 5, results are presented for each neural network architecture. Even thought the thickness prediction is not as accurate as previously, the prediction was still pretty satisfactory, where the maximum deviation reached is about 16% of the real value.

On the other hand, the elements prediction is where the different networks demonstrate very distinct results. It is visible that the 80-40-20-5 architecture always predicts correctly the gold targets, while the remaining two always predict it wrong. For this reason, the blue network (80-40-20-5) outperforms the red and green architectures (80-40 and 80-40-20, respectively).

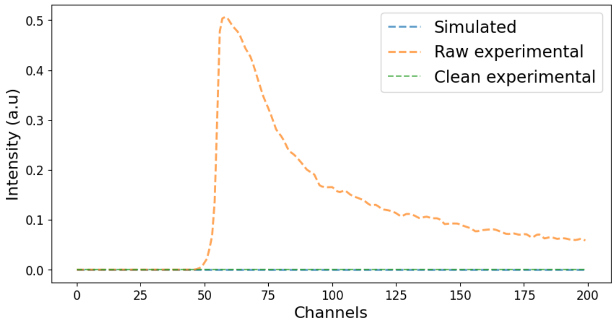


Figure 4: Simulated vs raw experimental vs clean experimental data: a cut of 200 channels was made for better visualization. The raw experimental data contains noise, especially low energy noise. To obtain the clean experimental data, all the channels except the peak channels were set to 0, completely vanishing the noise. In this way the model makes predictions more accurately since the clean data is much more similar to the training data, which are simulated spectra.

Although this way of treating data worked, it might not be the ideal way of treating it, because it requires to locate the peak manually for every spectra. Another point is that setting channels that contain some counts to 0 might erase relevant data that the spectra might contain. Introducing noise, such as Poisson noise, to the training set could prove to be a much better approach, mostly because ideally we do not want to convert experimental spectra into perfect simulated one, but the inverse: make the training set less perfect in a way that resembles more the experimental data and its imperfections, namely noise.

Something important to mention is that the results in Figure 5 are highly dependable on the training test. For random training sets, there were two visible result patterns: the neural network would either predict the material always correctly or always wrongly. The results presented Figure 5b) picture a run where one network always predicts correctly (the 80-40-20-5 architecture) and the remaining ones always predict incorrectly. While it is unclear why this happens, one reason might be simply because some training sets might contain more gold targets than others, which might lead to a neural network better suited to predict solely gold targets.

4 Conclusions

To conclude, we evaluated the performance of neural networks to analyze great amounts of RBS spectra. First, we managed to automate the acquisition of simulated spectra via the SIMNRA software. Afterwards, we proved that every neural network architecture that was tested can successfully predict the thickness and material of targets for simulated spectra. For experimental data, we demonstrated that for a certain training set and cleaned data, the network architecture 80-40-20-5 managed to predict the material correctly of gold targets, while maintaining reasonably accurate thickness predictions (between 4% and 16%).

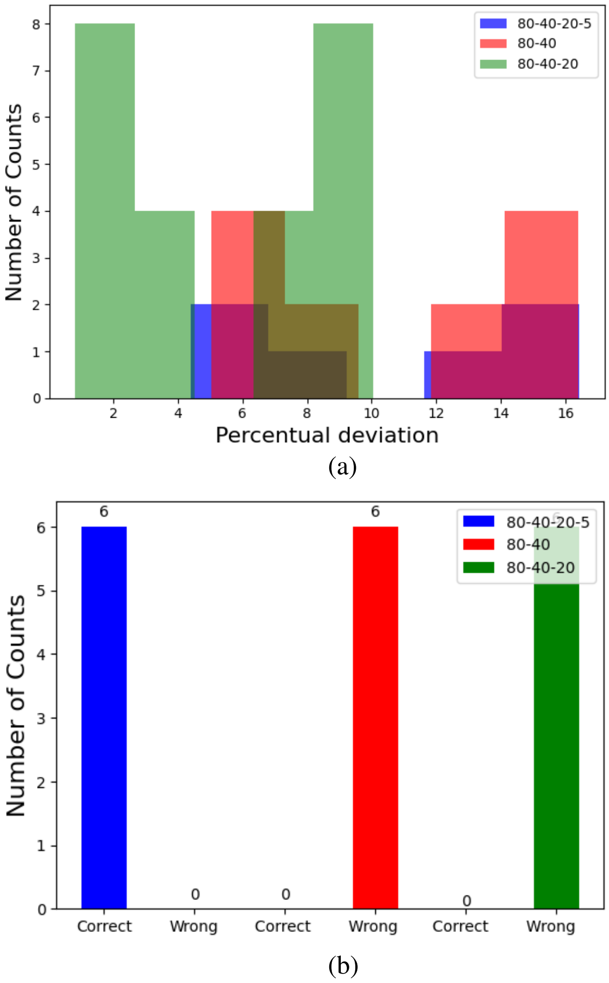


Figure 5: Number of counts per percentual deviation of the thickness (a) and number of correct/wrong counts of the material prediction (b). Note: in (a), the green network counts were multiplied by 4 while the red network counts got multiplied by 2. This was done to reduce the overlapping between the graphs, making the plot more readable and clear.

In the future, one can try to implement noise into the training set, avoiding less radical data cleaning methods such as the one adopted in this work. Additionally, it is also possible to use neural networks to ensure that the RBS measurement is proceeding as expected, by spotting possible issues that might occur (on-line analysis).

In the end, artificial neural networks proved to be a powerful tool that can be used to facilitate scientists jobs, especially analysing huge quantities of spectra due to its capacity to "learn" from previous analysis, just like us humans.

Acknowledgements

During this research, I have acquired and developed many different skills. The most notable were the devel-

opment of my Python skills and the acknowledgment acquired on Machine Learning. Additionally, I gained experience on automation of apps using *win32com* and knowledge on ion beam analysis, especially RBS.

Lastly, I would like to thank the whole group of NUCRIA, especially my supervisors Tomás, Ricardo and Francisco, for being very supportive and allowing me to take part on this internship.

References

[1] J. Mayer et al., *Handbook of Modern Ion Beam Materials Analysis* (1977)

[2] V. Oliveira et al., Journal of Physics: Conference Series (2022)

[3] J. Demeulemeester et al., Nuclear Instruments and Methods in Physics Research B (2010)

[4] N. Nené et al., Nuclear Instruments and Methods in Physics Research B (2006)

[5] R. Guimarães et al., Nuclear Instruments and Methods in Physics Research B (2021)

[6] P. Mehta et al., Physics Reports (2019)

[7] M. Mayer, *SIMNRA User's Guide*, https://mam.home.ipp.mpg.de/Report%20IPP%209-113.pdf (1997)

[8] https://www.lip.pt/index.php?section=research&page=research-group-details&details=project&projectid=80&line=Structure-of-matter

[9] https://www.ctn.tecnico.ulisboa.pt/

[10] B. Ramsundar et al., *TensorFlow for Deep Learning* (O'Reilly Media, Inc., 2018)

[11] A. Gupta (2021)