Lab 6-7-8: Project 1 – Prediction using NN and Fuzzy Systems

(Weeks 6-7-8)

Part A (Preliminary Version)

1 – Objectives

In this work you will try to predict when will an IoT gateway device fail based on the requests it receives and its processor load. Most of the requests received by the device are simply related with routing, but some require data processing, and can be quite computationally demanding.

2 – Dataset

We will use the pre-processed artificial dataset "IoTGatewayCrash".

The IoTGatewayCrash.csv file is composed of 1000 records. Each record corresponds to a 20 minute period of operation of the device that contains two inputs and one output:

a) Input 1: The normalized number of requests received by the device during the 20 min period.
b) Input 2: The processor average load during that period.
c) Output: Normal Operation (0) or Crash (1).

The dataset is highly imbalanced, and there are very few crash instances.

3 – Will it crash or not?

This is not a simple classification problem since we are dealing with sequential data, and probably what happened before is an important factor on the failure of the device. Therefore, you will likely need to consider the past. Since there are not many crash instances, it is unlikely that a fully automatic unsupervised method will be able to learn anything useful out of the data, but why not try?

a) Start by using a NN with 2 inputs and 1 output and see if it provides any results. Decide which hyperparameters to use (but don't lose too much time optimizing them), and evaluate the performance using Precision, Recall and F-measure.
b) Try to see how a LSTM performs in this problem. Use different sequence lengths (e.g.: 3, 4, 5). Use the Keras library, and the following article as an introductory guide https://machinelearningmastery.com/how-to-develop-lstm-models-for-time-series-forecasting/
c) You will likely reach the conclusion that you will probably need to use more than this kind of brute force approach. So look at the data and see if you can get some insights on the problem, and what kind of features you should try to use as the inputs to your model…

4 – Enter the expert…

An expert in intelligent systems was contacted to give us some insights regarding this problem. Here is what he has to say:

[Expert1]

"A quick look at the data shows that this is a very imbalanced problem, with very few positive instances. So, you should try to balance your training set before any attempt to learn from the data" (Note: DON'T balance the validation or test sets, only the training set).

"I also performed some quick basic data analysis and found out that all crashes happen during a High processor load, even though this is not a sufficient condition for the crash to happen. However, it seems that the number of requests itself isn't related with the crash. Therefore, what causes the crash must be related with something that happened in the past. This is definitely a system where order is important." (Note: Maintain the order of data in the training, validation and data sets)

"So, my advice would be to try to learn something using Processor_Load as one of the inputs, and use several previous instances of Number_of_Requests as additional inputs" (Example: try using the following features as the inputs for the NN: Processor_Load_t; $\text{Number\_of\_Requests}_{t-1}$; $\text{Number\_of\_Requests}_{t-2}$)

5 – Let's get another expert…

[to come soon]

6 – Evaluation and Validation

Use the previously acquired knowledge to evaluate and validate the performance of all the systems you implemented in 3, 4 and 5.

7 – Report

Write a report where you indicate the options you made regarding the data preparation, the experimental setup, the construction of the models, and all the evaluation and validation processes during all the phases of the project.

Send the report until Wednesday, May 20th, at 19:00 to:

joao.paulo.carvalho@tecnico.ulisboa.pt