

Configuration Manual

MSc Research Project
Data Analytics

Baris Anik
Student ID: x21236178

School of Computing
National College of Ireland

Supervisor: Rejwanul Haque

National College of Ireland
Project Submission Sheet
School of Computing



Student Name:	Baris Anik
Student ID:	x21236178
Programme:	Data Analytics
Year:	2022
Module:	MSc Research Project
Supervisor:	Rejwanul Haque
Submission Due Date:	14/08/2023
Project Title:	Configuration Manual
Word Count:	923
Page Count:	4

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	14th August 2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission , to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project , both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Baris Anik
x21236178

1 Introduction

This document contains the software tools and settings that must be installed in order to carry out academic work.

2 System Configuration

2.1 Hardware Configuration

All training and testing operations were carried out with a computer equipped with the following equipment:

- **Operating System:** Windows 10 (64 bit) Version: 22H2
- **Processor:** Intel(R) Core(TM) i7-10870H CPU @ 2.20GHz 2.21 GHz
- **RAM:** 16,0 GB
- **GPU:** NVIDIA GeForce RTX 3050 Laptop GPU
- **Device:** DELL G15 5510

The GPU has been used for training and test purposes.

2.2 Software Configuration

The following software and libraries were used for the production of the models produced in the study:

- **Software Tool:** Jupyter Notebook (version: 6.5.4)
- **Coding Language:** Python (version: 3.10.11)
 - **Libraries:**
 - * tensorflow (version: 2.9.0)
 - * keras (version: 2.9.0)
 - * Keras-Preprocessing (version: 1.1.2)
 - * pandas (version: 2.0.2)
 - * seaborn (version: 0.12.2)

- * opencv-python (version: 4.7.0.72)
- * numpy (version: 1.23.5)
- * glob2 (version: 0.7)
- * matplotlib (version: 3.7.1)
- * Pillow (version: 9.5.0)
- * jsonpointer (version: 2.3)
- * jsonschema (version: 4.17.3)

– **Jupyter Notebook-Related Libraries:**

- * IPython (version: 8.14.0)
- * ipykernel (version: 6.23.2)
- * jupyter_client (version: 8.2.0)
- * jupyter_core (version: 5.3.1)
- * jupyter_server (version: 2.6.0)
- * nbclient (version: 0.8.0)
- * nbconvert (version: 7.5.0)
- * nbformat (version: 5.9.0)
- * notebook (version: 6.5.4)
- * traitlets (version: 5.9.0)

3 Data Source

The BreakHis dataset (Spanhol et al.; 2016) can be obtained from the Federal University of Parana’s website. ¹ The directory structure of the dataset used in the study is shown in the image below:

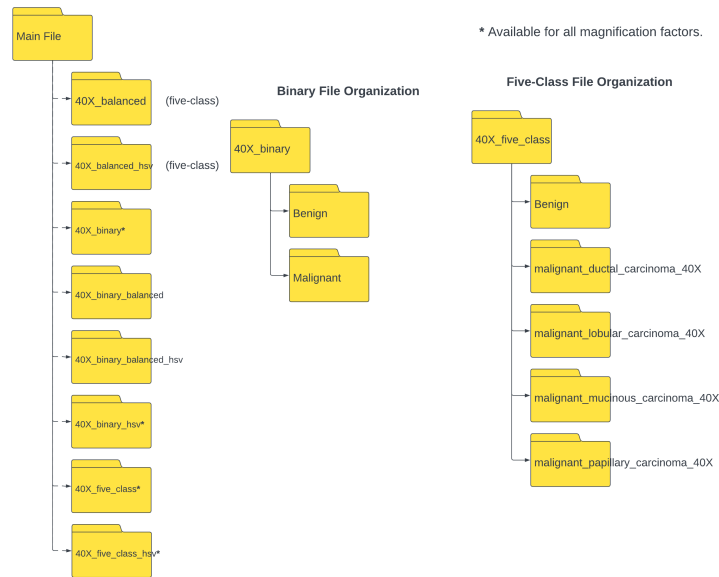


Figure 1: Directory Structure

¹<https://web.inf.ufpr.br/vri/databases/breast-cancer-histopathological-database-breakhis/>

4 Model Weights and Histories

The weight and history files of the models used in the study can be accessed from the cloud storage. ²

5 Model Training and Testing

5.1 Training an Individual Model

To perform an individual model training, any of the following files can be run, depending on class preference:

- Individual Models Binary Classification Training.ipynb
- Individual Models Five-Class Classification Training.ipynb

An individual model can then be trained by following the steps below:

1. First, the libraries in the first cell of the notebook need to be imported.
2. The second cell that is assigned data to the variables that store the hyperparameter settings required for the creation of the models must be executed.
3. The third cell containing the data sets to be used for training, validation and testing should be run.
4. To create a model, *createModel* must be run, and for the training process of the model, cells must be run in which *model_trainer* functions are defined.
 - (a) **Warning:** Before defining *model_trainer* function the *modelWeightSavePath*, *modelHistorySavePath* and *testSavePath* variables must be set as an existing directories.
5. To create any model, the *createModel* function must be given the desired base model, class, size, and version arguments.
6. Finally, to train the model, the *model_trainer* function should be given the first and second index outputs of the *createModel* function, version, data type and debug mode arguments. To set the epoch count and set the early stopping patience value in the models, the debug mode argument must be set to True. If the debug mode argument is True, the epochs and patience arguments should be given.

5.2 Training an Ensemble Model

To perform an ensemble model training, “Ensemble Models.ipynb” file can be used. An ensemble model can then be trained by following the steps below:

1. First, the libraries in the first cell of the notebook need to be imported.
2. The second cell that is assigned data to the variables that store the hyperparameter settings required for the creation of the models must be executed.

²<https://drive.google.com/drive/folders/1T1BNoH4jWfO-ZFVSPEXMn7z5e-M2nDES?usp=sharing>

3. The third cell containing the data sets to be used for training, validation and testing should be run.
4. To create an individual model, *createModel* must be run, and for the training process of the model, cells must be run in which *ensemble_model_trainer* functions are defined. In addition, to create an ensemble model, the cell in which the *createEnsembleModel* function is defined must be executed.
 - (a) **Warning:** Before defining *ensemble_model_trainer* and *createEnsembleModel* function the *modelWeightSavePath*, *modelHistorySavePath* and *testSavePath* variables must be set as an existing directories.
5. The individual model can be trained with the *model_trainer* function, or if it has saved weights, the weights can be loaded with the *load_weights* method.
6. A list called *models* should be created where the models are combined.
7. To create any model, the *createEnsembleModel* function must be take the desired model list, and version arguments.
8. Finally, to train the model, the *ensemble_model_trainer* function should be given the first and second index outputs of the *createEnsembleModel* function, version, data type and debug mode arguments. To set the epoch count and set the early stopping patience value in the models, the debug mode argument must be set to True. If the debug mode argument is True, the epochs and patience arguments should be given.

After creating the model with the *createModel* or *createEnsembleModel* function to test any model used in the study, the models can be made ready for use with the *load_weights* method by specifying the file directory downloaded from the address specified in Section 4.

Warning: Before performing this operation, it is important to ensure that the tensorflow library does not have a version mismatch.

References

- Spanhol, F. A., Oliveira, L. S., Petitjean, C. and Heutte, L. (2016). A Dataset for Breast Cancer Histopathological Image Classification, *IEEE Transactions on Biomedical Engineering* **63**(7): 1455–1462.
URL: <http://ieeexplore.ieee.org/document/7312934/>