Part B – Machine Learning and Software Integration

ü **Due** *11:59pm Sunday 30 November 2025*
ü **Contributes** 50% to your Assignment 2 result.
ü **Group of 3-4 students**.

The Part B of Assignment 2 (2B) requires you to work with your group to implement ML algorithms to train ML models for traffic prediction and integrate the traffic predictor with Part A to develop a fully functioned Incident Classification System (ICS).

In Part B of Assignment 2, the team can use any machine learning technique or combinations of them. You should take advantage of existing libraries such as PyTorch, Tensorflow, Keras, Theano, etc. We will provide you with a small dataset of Traffic Accidents. You should use this dataset for the following purposes: training ML models for predicting traffic incidents and testing/evaluating the performance of your models. At the very least, your ICS system will need to include the two basic deep learning techniques. Your ICS system will also need to implement at least one other technique (to be identified by you and approved by your tutor) to train another machine learning (ML) model and give a comprehensive comparison between different models. Your program will at least need to implement the following features:

ü At the minimum, the ICS will have to be able to train ML models using the dataset provided and give meaningful predictions based on these models
ü A GUI will be available for the user input, parameter settings and visualisation (and a configuration file for the defaults).

The Traffic-Incident Classification Problem

In this problem, there are four main tasks for the team:

1. Implement data processing program to extract data from the given dataset and store it in appropriate data structures to enable ML models to be trained/tested;
2. Implement ML/DL algorithms to train ML/DL models for traffic incident prediction using the provided dataset.
3. Implement an Incident Classification System that will notify users of incidents on a map.
4. Integrate Part A of Assignment 2 with the ICS to replace the nodes by the intersections, the cost of each edge by the travel time and perform the calculation to find the top-k paths to travel from O to D for any given pair (O,D) of origin and destination.

System requirements

For this assignment, we expect that the team will implement data processing methods and various ML/DL algorithms (including CNN) to train ML/DL models for incident prediction. The team will need to use the given dataset for training and also testing the ML/dl models to evaluate their performance.

The team will also need to port the programs you developed previously for Assignment 2A to enable it to search for optimal paths on the a map. The edge cost will need to be replaced by the predicted travel time and subsequently, the optimal paths can be calculated and returned.

The basic version of ICS will be for the Kuching area. A user can specify the origin and destination of their trip from the list of landmarks. The system then returns up to five (5) routes from O to D with the estimated travel time along each route. To simplify the calculation, you can make a number of assumptions: (i) The speed limit on every link will be the same and set at 60km/h; (ii) When an accident is detected it follows a simple formula following the attachment on CANVAS.

Note that, the objective is not to better Google Maps but to utilise the AI techniques you have learned (e.g., machine learning for image classification) to solve a real-world problem.

Report file

You must also include a report which has to be either in Microsoft Word or in PDF whose name is your Team ID (for example, A2B_Group123.PDF) containing your report. The aim of this report is for you to summarise your understanding of the problem and the algorithms you have used to solve the problem. We are most interested in the insights you have gained, especially those **using data obtained by running your software**.

Report Details: The report must be between 8 and 10 pages (excluding cover page and TOC).

- **Cover page:** including details of all group members (i.e., full names and student IDs), signed by all members of the group.
- **Table of contents (TOC)**.
- **Instructions:** Basic instructions of how to use your program. You can also include a **note** containing anything else you want to tell the marker, such as how to use the GUI version of your program, and something particular about your implementation.
- **Introduction:** Introduce the *Traffic Incident Classification Problem* and concise information about the algorithms the team have implemented.
- **Features/Bugs/Missing:** Include a list of the features you have implemented. Clearly state if a required feature has not been implemented. Failure to do this will result in penalties. Include a list of any known bugs. Also, anything else you want to tell the marker, such as how to use the GUI version of your program, and something particular about your implementation.
- **Testing:** Provide an overview of the test cases you have created to test your program (either manually or automatically). Report the results of testing your program. By testing, we refer to both the evaluation of the ML models and the software testing you have conducted to ensure that the software works well.
- **Insights:** Present and discuss the qualities of different ML algorithms used in your assignment. Which algorithms are better and why? **Use data collected to support your points.**
- **Research (if applicable):** If you managed to do some additional research to improve the program in some ways, please report it here.
- **Conclusion:** Conclude with a discussion about ML algorithms and their suitability to the problem and about the integration of different modules to produce the end-to-end ICS. Include thoughts about how you could improve performance.
- **Acknowledgements/Resources:** Include in your report a list of the resources you have used to create your work. A simple list of URL's is not enough. Include with each entry a basic description of how the person or website assisted you in your work.
- **References:** Cite the sources you referred to in your Assignment (implementation, report, etc.)

Tips:

- All figures and tables need to be properly captioned with sensible descriptions.
- Report presentation should include header/footer information (pages numbers, etc.)

Marking Scheme:

Requirements (or equivalent sections)	***Mark***
Implement appropriate data processing programs to extract data from the provided dataset and store data in appropriate data structures to enable ML models to be trained & tested	**9**
Implement at least 3 ML algorithms (including CNN and any algorithms of your choice) for traffic incident prediction.	**7(x3)**
Conduct comprehensive evaluation to compare different ML models	**15**
Integration of Part A & Part B to develop an end-to-end software for ICS	**15**
Testing: At least 10 test cases have been created to cover different problem scenarios. Test results have been checked and documented.	**10**

Report: Clear and provide sufficient information about the problem and your algorithm/solution AND the insights you have obtained through your program.	**15**
Research: If you show some initiatives in researching about the problem and solutions, or carrying out extensive tests to provide interesting data about the algorithms, or getting some clever optimization, etc. with a well-written **Research** section in the report to discuss and demonstrate these initiatives	**15**
Total	**100**
You need to follow good programming practice (e.g., well-designed, well-structure codes with clear and helpful comments). Failure to do so get penalty.	**Up to -10**

For each individual student: (the maximum mark a student can get for Assignment 2B is 100 and the minimum mark is 0)	***Mark***
Failure to provide in class updates of the assignment progress to your tutor. There is no 1 week gap between first & last update	**Up to -40**
Failure to contribute to the work performed by the team or to collaborate with other team members to achieve the desired outcomes for the project.	**Up to -80**

Some ideas for research initiatives

- The standard version of the system only deals with a small area of Kuching and uses a very small dataset accident images. Under this research initiative, you can look at more comprehensive datasets of different classes of accidents. Data processing for large amount of data will be a challenge.

- Visualising your system predictions and route recommendations. Inspired by Google Maps Traffic, can you build something similar using open source resources such as OpenStreetMap?

- For other ideas, please feel free to do the research yourselves and discuss with your tutors to get their approval first before doing the research component.