National College of Ireland

National College of Ireland

Artificial Intelligence
(BSHC3, BSHCE3, BSHDS3)

Submission Deadline: ***Friday, 5th of December 2025 at 23:55***

Continuous Assessment (CA) Type: Project 40%

Sumit Thripati, Harshani Nagahamulla

Weight: The project will be marked out of 100.

Module Learning Outcomes

This assessment examines the following learning outcomes for the module:

LO3	Demonstrate the use of structures for knowledge representation and logical reasoning systems while solving practical AI problems.
LO4	Evaluate the architecture of intelligent agents to solve real-world problems.

TURNITIN

All report submissions will be electronically screened for evidence of academic misconduct (i.e., plagiarism and collusion)

Late submissions will not be penalized if the student applied for an extension through NCI360 and it was approved.

Project Overview: Intelligent News Categorization and City Navigation Agent

In a world where urban life moves faster than ever, the AI-driven journalism platform CityPulse AI revolutionizes how news is collected, categorized, and delivered from major cities across the Southwest.
The system operates across four interconnected offices—Phoenix (Headquarters), Los Angeles, San Diego, and Las Vegas—each acting as both news hubs and regional data intelligence centers.
In this project, you are required to implement an AI application for a news reporter. Building on your programming and machine learning skills, you are required to develop an intelligent agent called "CityPulse AI" that categorizes news articles and plans optimal travel routes based on the geographical location of the news.

Deliverables:

You are required to submit the following

1. Code in a zipped (.zip) folder containing Jupyter Notebook (.ipynb) files, Saved .pkl models and .csv results.
   The zip file should be named as: *Code_Firstname_Surname.zip*
2. Final report (PDF) 8-10 pages.
   The report should be named as: *Report_Firstname_Surname.pdf*

Instructions:

- Download the '*AIProject.zip'* file. The zipped folder contains three files
  1. *CityMap.ipynb* – Contains the city maps (distance between locations in each city) and heuristics (straight line distance between office and other locations in each city)
  2. *News_Categorizer_RDF.xml* – An ontology containing information about news articles
  3. *Daily_News.csv* – A list of daily news that need to be categorized.
- All submissions must be made via the Moodle page before the deadline. You will find two links on the Moodle page one each for the Report and the Code.
- The report should contain answers to all the questions; indicating the question number. You can include figures and graphs as necessary. Any graphs in the report must be generatable by running your code.
- Any work not your own should be referenced. Use IEEE referencing style.

Scenario Overview:

CityPulse AI consists of two specialized agents:

- The Executive Correspondent acts as a senior AI journalist responsible for monitoring regional operations across Phoenix, Los Angeles, San Diego, and Las Vegas. Their primary responsibility is to visit a subset of the branch offices, starting and ending at Phoenix (Headquarters) — analyzing breaking trends and coordinating cross-city stories.
  The agent must find the most efficient travel path between the chosen offices using Genetic Algorithms (GA) to solve the Traveling Salesman Problem (TSP).

  Distance between the offices are as follows

City	Phoenix	Los Angeles	San Diego	Las Vegas
Phoenix	0	370	355	300
Los Angeles	370	0	120	270
San Diego	355	120	0	330
Las Vegas	300	270	330	0

- The **City Reporter** is an intelligent local AI journalist assigned to a single city office (e.g., Phoenix, Los Angeles, etc.). Each City Reporter covers breaking stories in nearby suburbs and performs two key operations:
  - **News Categorization**
    Using a **Naive Bayes Classifier**, the City Reporter processes news data from the **News_Categorizer_RDF.xml** file.
  - **City Navigation**
    Once categorized, the City Reporter locates where the event occurred within the city's suburb network.
    It then finds the **shortest route** from the city office to the event's location using **search algorithms** (e.g., A* Search and Uniform Cost Search).

Q 1. Identify the most suitable agent architecture for "CityPulse AI". Discuss why you selected this architecture. *(10 Marks)*

Q 2. Developing the News Categorization Model using Naive Bayes Classifier *(30 Marks)*

a. Read and understand the RDF/XML ontology dataset in the '*News_Categorizer_RDF.xml'* file. This contains information about news articles. Give a small description of the ontology. Extract the necessary fields from the ontology. (Hint: Check the code from Lab 7)

b. Using this dataset, train three Multinomial Naive Bayes models; one for each of the following cases:
    - Headlines only.
    - Short descriptions only.
    - Combination of both headlines and short descriptions.

c. Evaluate the performance of each model. Print the accuracy and plot the confusion matrices for each model. In your opinion, what is the best model? Discuss your answer. (Use Screenshots of your results in the document.)

d. Save the best model using pickle (Hint: Check the code from Lab 7)

Q 3. Developing the Path Finder *(30 Marks)*

a. Implement 2 suitable search algorithms to find the shortest path from the office to the news location and back. Use the distances between locations in each city and the heuristics for each city from '*CityMap.ipynb'* as needed by your algorithms. (Hint: Check the code from Lab 3)

b. Analyze the results by comparing the performance of the search algorithms. Address the following questions:*
    - How do the costs and time compare between the Searches?
    - Do your search algorithms use heuristics? How does the heuristic affect its performance?
    - Did both algorithms find the same path? If not, why do you think this occurred?
    - In which scenarios would you prefer one algorithm over the other?

Q 4. Employee Task Automation *(30 Marks)*

a. Get User Input for Employee Type: Prompt the user to select their Employee type: Executive Correspondent or Reporter.

b. **For Executive Correspondent**: If the user selects an Executive Correspondent, prompt them to select the offices to visit from the following options: Los Angeles, San Diego, and Las Vegas. Implement the Traveling Salesman Problem (TSP) using a Genetic Algorithm (GA) to optimize the travel route through the selected cities, starting and ending at Phoenix (Head office). (Hint: Check the code from Lab 4)

c. **For City Reporter**: If the user selects a City Reporter, ask for the city they work in and the news category they report on (e.g., Politics, Sports, Business, etc.).

   Read the *'Daily_News.csv'* file and load the pre-trained Naive Bayes (NB) model. Categorize the news from the *'Daily_News.csv'* file using the NB model. If any news articles match the reporter's city and category: Find the shortest path to the news location from the reporter's office using the appropriate search algorithm from Q 3.

Grading Rubric: Using the following scale, each task will be awarded a percentage of the total mark.

Grade Criterion	H1 (> 70%)	H2.1 (> 60%)	H2.2 (> 50%)	Pass (> 40%)	Fail (< 40%)
Q 1. Identify the most suitable agent architecture for “CityPulse AI” *(10 Marks)*	Comprehensive and well-justified selection of agent architecture with clear explanations on why it is suitable for "CityPulse AI"; demonstrates in-depth understanding of agent types and their application to the tasks.	Clear identification of a suitable architecture with good rationale, though may lack some depth or minor details.	Architecture selection is mostly suitable, but justification is weak or lacks depth in understanding.	Architecture selected may be somewhat applicable but lacks adequate justification or contains significant gaps.	Inappropriate architecture selection or missing justification; no clear understanding of the agent types.
Q 2. Developing the News Categorization Model using Naive Bayes Classifier *(30 Marks)*	a. Ontology Understanding: Provides a thorough and accurate description of the RDF/XML ontology with relevant field extractions. b. Naive Bayes Models: Effectively trains three NB models and presents a robust comparison among them. c. Model Evaluation: Comprehensive analysis with accuracy metrics, confusion matrices, and a thoughtful discussion of results. d. Best Model: Correctly identifies and saves the best model using pickle.	Meets all requirements with minor gaps in detail or analysis. All NB models are trained, evaluated, and compared, though some explanations may lack depth.	Reasonably completes the tasks but with noticeable gaps, such as incomplete model evaluations or partial field extraction from ontology.	Partially fulfills the requirements, such as training only one model or missing evaluation metrics.	Minimal to no work shown; significant steps missing, such as failing to train the NB models or lack of ontology extraction.
Q 3. Developing the Path Finder *(30 Marks)*	a. Implementation: Successfully implements two appropriate search algorithms with clear coding and execution. b. Comparison & Analysis: Thorough analysis of cost, time, heuristic use, and path differences, including scenarios where one algorithm is preferable.	Mostly completes implementation and comparison, though some details may be underexplored or missing; performance analysis is clear but could be more robust.	Reasonable implementation of algorithms with partial comparison; some answers are shallow or missing key points.	Partial completion, such as only implementing one algorithm or very limited comparison; lacks meaningful analysis.	No clear implementation or major components missing, such as the comparison or analysis section.
Q 4. Employee Task Automation *(30 Marks)*	a. User Input: Correct and intuitive prompts for Employee type selection. b. Executive Correspondent Pathfinding: Efficient implementation of TSP with GA for office route optimization, returning to Phoenix. c. News Categorization and Pathfinding: Accurate categorization using the pre-trained NB model and relevant search algorithm for reporters, with clear results.	Sufficiently addresses all aspects with minor gaps or inefficiencies in execution; demonstrates understanding of TSP and search algorithm integration.	Addresses most parts but with noticeable limitations in implementation or minor errors in logic; TSP or NB model categorization may be partially incomplete.	Partial completion, such as only implementing one part (TSP or news categorization); lacks coherence in workflow.	Major sections missing or incorrectly implemented; little to no work demonstrated in automation tasks.