|  |  |  |  |
| --- | --- | --- | --- |
| **University of Chester -** Undergraduate Programmes Assignment Specification  **Faculty of Science, Business & Enterprise**  **Department of Computer Science** | | | |
| **Module No**  CO6226 | **Module Title**  Artificial Intelligence and Machine Learning | **Assessment No**  1 | **Weighting**  63% |
| **Submission Date**  21st February at 17:30 | | | **Feedback due by**  20th March 2024 |
| **Assignment Title**  Machine Learning Project | | | |
| **Learning Outcomes Assessed**   1. Demonstrate a critical awareness of the capabilities of artificial intelligence. 2. Apply theoretical solutions to issues of representation. 3. Critically analyse the role that systems such as neural networks or software agents, for example, can play in the development of intelligent technologies. 4. ~~Critically evaluate how organisations in various domains can utilise intelligent software.~~ 5. Use suitable software tools to design, implement, test, document and evaluate software that demonstrates intelligence, or to design an intelligent architecture within a suitable domain. | | | |
| **Submission Information**  For projects that include programming code:  The TurnItIn submission box will have multiple parts. You must submit to the appropriate part   * A PDF file with your report and a section with all programming code from your project (in a monospace font), followed by a reference list in APA format. * A ZIP file containing the project   Both files must be named with your assessment (J number), e.g. J123456.pdf and J123456.zip. The name for each entry on TurnItIn must also be your assessment number.  Files submitted in an incorrect format will usually be marked as zero.  All components must be submitted to avoid receiving a mark of zero.  Any late work penalties for assignments will be calculated using the latest submission date/time. | | | |
| **Extensions**  Extensions should be requested through the online system available on the Registry services pages on [Portal](https://v3apps.chester.ac.uk/oed/#!start). Late work is penalised at the rate of 5 marks per day or part thereof.  **Academic Conduct** The material you submit must be your own work. Please avoid colluding with peers on your work. The penalties for breaching the academic conduct policy are severe. The minimum penalty is usually zero for that piece of work. Further information is available at Portal > Support Departments > Academic Services > Academic Standards > Academic Conduct > Information for Students > [Academic Conduct](https://portal1.chester.ac.uk/aqs/Pages/aqss-academic-conduct-information-students.aspx) | | | |
| **Generative AI The use of generative AI tools where not permitted will be treated as a breach of the academic integrity policy.**  **This assignment does not permit the use of any generative AI tools, including but not limited to ChatGPT, Bard, Copilot, Midjourney, and others.** | | | |
| **Delete if not relevant**  **Referencing code**  Code adapted from third parties must be clearly referenced using comments to denote the start and end of the adapted code. You must also include an APA format reference in the PDF file.  **Example of referenced code**  *//code adapted from Thomson, 2012* **if** (someCharacter == **'z'** || someCharacter == **'Z'**) {  someCharacter -= 25;  } **else** {  someCharacter += 1; } //end of adapted code  **Example of reference entry in PDF file** Thomson, C. (2012). Rot-13 function in Java?. Stackoverflow. Retrieved October 25, 2021, from <http://stackoverflow.com/questions/8981296/rot-13-function-in-java> | | | |

**Assignment Brief**

**Assignment Brief**

Select **one** of the three assessments below. Assessments should be **2,500 words long, or equivalent.** Additionally, a draft submission is required for formative feedback to ensure your project is going in the right direction. This draft is due on the 2nd of January 2024, failure to do so will result in a 10% mark penalty – this penalty will not reduce your mark below 40%. If your work is not draft ready, e.g., an implementation project which has been designed but not implemented, then you should submit a report (max 200 words) stating your current progress.

**Note:** One of the challenges of this project is to condense as much information into a small count. Students who ignore the word count restriction could gain a significant advantage over those who do not. Therefore, word count penalties will be applied. Word counts will ignore words on title pages, titles, citations, references, diagrams, and tables. Therefore, you should make good use of diagrams and tables to help explain concepts.

1. **Algorithm Implementation (Implementation: 1000-word report, 1500-word equivalent program)**

Implement a classification or clustering algorithm in either Java or Python, e.g. Naïve Bayes, K-means, J48-Decision Tree, etc. You are required to implement the algorithm yourself; this means using a Data Mining package or wrapper is NOT allowed. You must find or create an appropriate data set for classification or clustering. Your project must contain:

1. The data set used
2. All programming code
3. Instructions on how to run your program
4. A short 500-word report, containing:
   1. Description of the data set:
      1. Attributes
      2. Attribute types
      3. Number of records (minimum of 20 records)
      4. Where you obtained it, or if you created it yourself
   2. Description of any pre-processing steps performed; if you did not need to use any pre-processing, then describe what steps may have been taken for your chosen algorithm
   3. An informal description of the algorithm
      1. What it is used for
      2. What it does
      3. How its algorithm operates
   4. A formal description of the algorithm, including any equations and pseudocode with references
   5. Discussion of how the algorithm performed, using appropriate performance metrics for that algorithm

(Report word counts over 1100 words will receive a minimum penalty of 5%)

1. **Demonstrate Data-mining Algorithms (Demonstrate: 2250-word report, 250-word equivalent project)**

Data mining is the extraction of knowledge and meaning from large sets of data. It can be used for prediction and analysis. There are many different data mining techniques and algorithms to achieve “knowledge discovery in databases”. This assignment is to look at least two possible data mining algorithms of your choice and analyse them.

1. Research **EACH** algorithm and describe: (**30 marks**)
2. What the algorithm is used for and how it works
3. Justification for selecting the algorithm for your dataset
4. Give a formal representation of the algorithm along with a description, including any equations and pseudocode with references
5. Describe what pre-processing steps are often required
6. Describe the metrics used to evaluate the accuracy of the technique, e.g. Kappa statistic, confusion matrix, etc.
7. Describe the strengths and weaknesses for each technique

(Note that your algorithm selection is important. Some algorithms are extremely simplistic, e.g., ZeroR and Decision Stump. Please avoid using algorithms that are considered `weak learners’ unless you use them as an ensemble learner, e.g., combining several algorithms together for bagging, boosting, or stacking)

1. The second part of the assignment is to test each algorithm on a data set using Python machine learning libraries. (Note, non-computing students, e.g., Electrical Engineering and Physics, may use WEKA if they wish). (**40 marks**)
2. Choose a specific data set which can be used to evaluate all your algorithms
   * You can use any data set on the internet, but only if you have permission to use it, e.g., it is described as an `open’ data set
   * Or you could create your own (not recommended)
3. Describe what pre-processing is required to execute each algorithm on the data set
4. Describe what test options you used to analyse the data set, and what it means, e.g., 10-fold cross validation and what 10-fold cross validation means
5. Evaluate the performance of the algorithm on the data, e.g., discuss the confusion matrix results, etc.
6. Finally, compare each algorithm (**30 marks**)
7. Discuss and compare the results of each algorithm
8. Mention strengths/weaknesses from literature, e.g., do your results match what the literature says.

**The WEKA tool for non-computing students is available at:** [**https://www.cs.waikato.ac.nz/ml/weka/**](https://www.cs.waikato.ac.nz/ml/weka/)

(Report word counts over 2750 words will receive a minimum penalty of 5%)

1. **Designing an Intelligent System (Design)**

AI techniques are deployed in a range of software applications. Examples include Expert Systems, Natural Language Processing Systems, Intelligent Agents, Intelligent Web Services, Information Retrieval Systems, Knowledge Management Systems, Robotics, and Computer Games.

Select a suitable domain where one or more intelligent technologies might be applied. This might be a domain with which you are familiar through working, domestic or leisure activities, for example.

Your task is to develop a high-level design for an intelligent system that might operate in your chosen domain. You might focus on a complete system, or you might choose to concentrate in detail on a subset of the system if it is large or complex.

Identify the specific tasks that the software will undertake and specify any interfaces to other systems that might be necessary. If possible, you should produce a graphical representation of the components of your system and of any communication that takes place between them. You should use a suitable tool, such as Visio, to do this.

You may include simulations or demonstrations using multimedia tools or other software to illustrate the functionality of your system.

You must also produce a critical evaluation of your system demonstrating how it features specific AI techniques. Do you expect to represent knowledge, for example, and if so how? Will your system do any reasoning? Will it use statistical methods, machine learning or natural language processing, for example? Consider how effective these or similar techniques might be in meeting the requirements of your system.

You must complete each of the following tasks:

1. Produce a rationale for your proposed system **(30 Marks)**
   1. Explain the context and evaluate any technology that is currently deployed.
   2. Assess what sort of intelligence is required and how it will be implemented.
   3. Consider the tools/techniques that might be required.
   4. Discuss the advantages and limitations of deploying intelligent technologies.
2. Produce a design for your proposed system showing key components and the relationships between them **(40 Marks)**
   1. You may use techniques such as storyboards or pseudo code.
   2. You may use a formal approach such as Data Flow Diagrams, Flowcharts or UML diagrams
   3. You might find it helpful to create hierarchical diagrams
3. Critically evaluate the effectiveness of the technology that you have recommended **(30 Marks)**
   1. Assess the scope for future development.
   2. Address the current aims and limitations of intelligent technologies.

(Report word counts over 2750 words will receive a minimum penalty of 5%)

**Assessment Criteria**

**Draft Submission**

The draft submission must contain which assignment project you have chosen and an outline of what you intend to do. The purpose of the draft submission is to ensure that all the choices you have made are suitable, so you must make all your decisions are clear in this submission. The following is a breakdown of what is expected for each assignment type:

1. For the algorithm implementation, you must have selected your algorithm for implementation. Your tutor will be available to help you make this decision, so make use of their office hours and time during lectures. Your draft must contain a formal and informal description of your algorithm to demonstrate how you intend to implement the algorithm. Additionally, you should submit any code you have written, although this is not expected to be complete.
2. For the algorithm demonstration, you must have a suitable dataset and three algorithms chosen with the appropriate pre-processing completed. Your report does not need to be completed, but there should be a full description of at least one of the algorithms.
3. For the designing of an intelligent system, you must have chosen your project with enough detail to explain what you intend to do. This will allow you tutor to determine if your project has sufficient complexity for the report.

**Algorithm Implementation**

To obtain a pass mark you will have discussed the algorithm and data set along with an implementation. The implementation does not need to fully work but it must show a good understanding of the algorithm.

Additional marks will be given for a fully working algorithm that has been implemented, executed on an appropriate data set and evaluated.

|  |  |
| --- | --- |
| Category | Mark |
| Data Set (At least 20 records) | 5 |
| Appropriate Pre-Processing | 5 |
| **Description of data set in report:** |  |
| Attributes stated | 2 |
| Attribute types stated | 2 |
| Number of records stated | 2 |
| Where obtained with a reference | 2 |
| Informal description of the algorithm | 5 |
| Formal description of the algorithm | 5 |
| Discussion of how the algorithm performed | 6 |
| Metrics used to evaluate the performance | 6 |
| Implementation of the algorithm | 60 |
| **Total** | **100** |

**Data Mining**

Below is a table with a breakdown of the marks for both using Python and for using Weka. The Python task has additional marks allocated to the programming code, as it requires technical ability to build the models.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Python Category | Marks |  | Weka Category | Mark |
| Algorithm 1: |  |  | Algorithm 1: |  |
| Description | 10 |  | Description | 15 |
| Python Code | 10 |  |  |  |
| Formal Representation | 4 |  | Formal Representation | 4 |
| Pre-Processing – What pre-processing the algorithm requires and what you have done for your dataset for this algorithm | 3 |  | Pre-Processing – What pre-processing the algorithm requires and what you have done for your dataset for this algorithm | 3 |
| Evaluation Metrics | 5 |  | Evaluation Metrics | 5 |
| Strengths/Weaknesses | 5 |  | Strengths/Weaknesses | 5 |
| Algorithm 2: |  |  | Algorithm 2: |  |
| Description | 10 |  | Description | 15 |
| Python Code | 10 |  |  |  |
| Formal Representation | 4 |  | Formal Representation | 4 |
| Pre-Processing – What pre-processing the algorithm requires and what you have done for your dataset for this algorithm | 3 |  | Pre-Processing – What pre-processing the algorithm requires and what you have done for your dataset for this algorithm | 3 |
| Evaluation Metrics | 5 |  | Evaluation Metrics | 5 |
| Strengths/Weaknesses | 5 |  | Strengths/Weaknesses | 5 |
| Test Options Used - along with explanation of what it does and why it was chosen. |  |  | Test Options Used - along with explanation of what it does and why it was chosen. |  |
| Alg1 | 2 |  | Alg1 | 2 |
| Alg2 | 2 |  | Alg2 | 2 |
| Algorithm Evaluation: |  |  | Algorithm Evaluation: |  |
| Alg1 | 6 |  | Alg1 | 6 |
| Alg2 | 6 |  | Alg2 | 6 |
| Evaluation: |  |  | Evaluation: |  |
| Cross-Comparison | 5 |  | Cross-Comparison | 10 |
| Results to literature | 5 |  | Results to literature | 10 |
| Total | 100 |  | Total | 100 |

**Designing an Intelligent System**

To obtain a pass mark, answers should demonstrate that questions i – iii from **option 1** have been completed satisfactorily. A satisfactory description of the problem domain, a high-level design of a possible solution, and a satisfactory critical evaluation of the limitations of the technology should be supported by appropriate references showing evidence of relevant research.

Additional marks will be awarded for more detailed and accurate coverage of these points, including evidence of extended system design, and demonstrating a critical awareness of the theory underpinning the work and clear insights into the potential for applying intelligent technologies to the problem domain.

|  |  |
| --- | --- |
| Category | Mark |
| Context and Current Tech | 10 |
| Tech Required | 7 |
| Tools/Techniques | 8 |
| Advantages/Disadvantages | 5 |
| **Design** |  |
| Description of system | 30 |
| Use of diagrams/code/flowcharts | 10 |
| **Evaluation** |  |
| Assess scope for future | 10 |
| Current aims and limitations | 10 |
| Grammar/Structure | 10 |
| **Total** | **100** |