Project Description

(*Use additional pages as needed for this report.*)

PART I.

1. General Information

Project Title: Applying Machine Learning/deep learning to Astronomy
Date submitted: Nov 4th, 2021

2. Project Overview (*Describe the project & its purpose*)

Visual galaxy morphologies are used by astronomers to study the dynamical structure of the systems for about 90 years. A wide range of morphological features such as bulges, and bars provide information about the history of the host systems. The Galaxy Zoo project combined the power of thousands of volunteers and produced an impressive amount of output of galaxy morphology classifications. However, a project of its magnitude would not be efficient enough for modern study of astronomy.

Our project would be aiming for automatically classification of the galaxy images based on their morphological features. We would first do some relative research to compare current approaches implemented through machine learning and carefully select 2-3 of them. After studying the algorithms and cases of practical applications, one would be selected and implemented. Our final product would be a galaxy morphology classifier with optimized function and an ideal accuracy.

3. Deliverables (*Describe all products to be produced*)

We plan to design and implement a program working as galaxy morphology classifier. After the procedure of testing and evaluation, a final reflection report with details of the implementation would be generated. We would also introduce our project structure, functionalities, and its competitive power by giving a presentation.

4. Requirements (*Describe the required resources*)

- Systematically learn Machine Learning algorithms.
- Search and read articles to get a rough idea of the popular approaches in the relative area.
- Master techniques of programing in Python.
- Adequate amount of labeled galaxy images as training set.
- Access to necessary computational resources to run, test and evaluate the program.

5. Constraints

- Scope: development of a galaxy morphology classifier.
- Quality: we expect a classifier with accuracy greater than or equal to 85%.
- Methodology: Random Forest Algorithm/Convolutional Neural Networks would be used.
- Resource: PyCharm and some relative packages would be used for development.
- Time: we plan to complete this project by March 2022.
- Cost: as long as our budget could afford.

PART II.

1. Risk analysis

Potential Risks

- Inadequate design of the classifier at first step
- Inadequate or inefficient management at any step of the project
- Overly optimistic plans on any one of the components
- Silver-bullet syndrome among group members

Risk Costs & Effects

- Cost of Inadequate design: We may spend extra time and resource to develop redundant components or even worse, develop a program with improper structure.
- Cost of Inadequate or inefficient management: This may lead to different members of the team working on same component repeatedly or none of the members working on some certain component. Inadequate management may potentially lead to delays as well.
- Cost of Overly optimistic plans: This would probably cause the lack of backup plan or backup technology. If we find our strategy does not work at some certain point during the project without a backup plan to fix it, that would be a fatal problem for the whole project.
- Cost of Silver-bullet syndrome: Silver-bullet syndrome among group members would cause an underestimation of the issues we meet. And this is where waste of resource and delays usually be triggered.

Alternatives

- Alternatives if we have Inadequate design: We could choose to come up with a new design or rather modify the original one.
- Alternatives if we have Inadequate or inefficient management: A project management tool could be introduced if necessary or choose one of our team members as a leader to get everything organized.
- Alternatives if we have Overly optimistic plans: We could make a new plan and reschedule all the tasks or just keep the original plan and simply add the overdue tasks in remaining time slot.
- Alternatives if we have Silver-bullet syndrome: New technology could be implemented to solve the problems caused by Silver-bullet syndrome or we stick with the original technology with more workload on each member's shoulder.

Evaluating & Comparing Alternatives

1)New design or rather modify the original one?

- New design: We would have to redo part of the job already done to implement the new design. And to make the new design itself needs humanity and time resources. But this might help to solve the whole problem.
- Modify the original one: We need comparatively less time to modify the original design without redoing all the previous job. But we might need to deal with some remaining issues in the future.

2)Project management tool or team leader?

- Management tool: If we choose to use a management tool, we might need extra time for everyone to learn how to use it and get used to it at first. But after that, everything goes smoothly if the tool is properly chosen.
- Team leader: If one of the team members is chosen to be a leader and take responsibility of organization, it would not take everyone's time to get used to a new system. But this chosen member need to contribute extra time and energy regularly on management tasks.

3)New plan or modify the original plan?

- New plan: A new plan would help to rearrange the remaining tasks in a more efficient way compared to the current plan. But we need extra time to discuss and make this brand-new plan.
- Modify original plan: Modifying the current plan would save our team a lot of time. But if we just simply insert the unfinished tasks into the future time slots, it would definitely affect the implementation of the future tasks.

4)New technology or more workload?

- New technology: The introduction of a new technology would probably help us with the current issue in a considerable extend. But it might also potentially cause problems—even more than before—if we are not careful enough when doing the research and choosing the new technology.
- More workloads: More workloads might cause poor motivation and affect the progress of future tasks. But this is a more reliable and economical way compared to the introduction of new technology.

PART III.

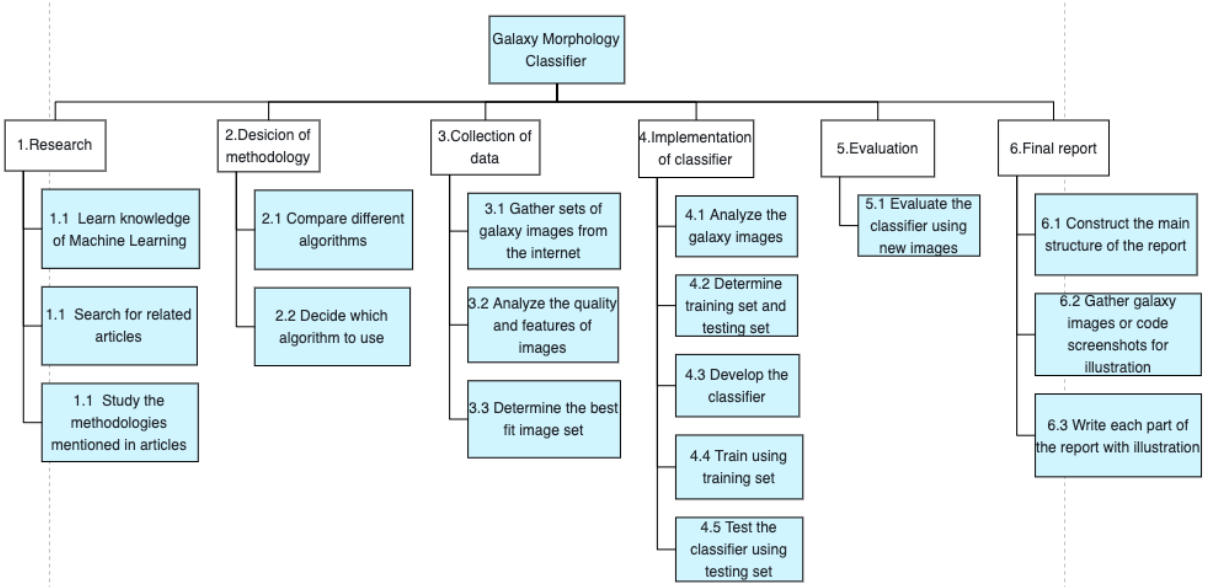
1. Work Breakdown Structure (WBS)

Highest Layer/Deliverable:
Galaxy Morphology Classifier

Middle Layer/Milestones:
1.Research
2.Desicion of methodology
3.Collection of data
4.Implementation of classifier
5.Evaluation
6.Final report

Lowest layer/Work Packages:
1.1 Learn knowledge of Machine Learning (Oct 1 – Oct 31)
1.2 Search for related articles (Nov 1 – Nov 15)
1.3 Study the methodologies mentioned in articles (Nov 15 – Nov 30)
2.1 Compare different algorithms (Dec 1 – Dec 7)
2.2 Decide which algorithm to use (Dec 8 – Dec 10)
3.1 Gather sets of galaxy images from the internet (Dec 11- Dec 20)
3.2 Analyze the quality and features of images (Dec 21 – Dec 23)
3.3 Determine the best fit image set (Dec 24 – Dec 26)
4.1 Analyze the galaxy images (Dec 27 – Dec 28)
4.2 Determine training set and testing set (Dec 29 – Dec 31)
4.3 Develop the classifier (Jan 1 – Jan 31)
4.4 Train using training set (Feb 1 – Feb 10)
4.5 Test the classifier using testing set (Feb 11 – Feb 20)
5.1 Evaluate the classifier using new images (Feb 21 – Feb 28)
6.1 Construct the main structure of the report (Mar 1 – Mar 1)
6.2 Gather galaxy images or code screenshots for illustration (Mar 2 – Mar 2)
6.3 Write each part of the report with illustration (Mar 3 – Mar 3)

(WBD structure on next page)


Galaxy Morphology Classifier
1.Research
1.1 Learn knowledge of Machine Learning
1.1 Search for related articles
1.1 Study the methodologies mentioned in articles
2.Desicion of methodology
2.1 Compare different algorithms
2.2 Decide which algorithm to use
3.Collection of data
3.1 Gather sets of galaxy images from the internet
3.2 Analyze the quality and features of images
3.3 Determine the best fit image set
4.Implementation of classifier
4.1 Analyze the galaxy images
4.2 Determine training set and testing set
4.3 Develop the classifier
4.4 Train using training set
4.5 Test the classifier using testing set
5.Evaluation
5.1 Evaluate the classifier using new images
6.Final report
6.1 Construct the main structure of the report
6.2 Gather galaxy images or code screenshots for illustration
6.3 Write each part of the report with illustration