Contents

ECMM427 - Group Project Course work 1

Project specification and plan

October 24, 2017

1 Problem definition

As the information about the state of our environment becomes more readily available, the conversations humanity are having about the correct course of actions to take regarding our impact becomes ever more inclusive and expansive. Our understanding of the systems that affect our climate have so far been very useful in predicting the changes that have occurred, however they are by no means entirely accurate and more to the point, the models that are created do not all agree on the trends and the impacts of those systems. This discrepancy is a focus point of some to illustrate a lack of understanding of the subject, but this is likely due to subtle differences in the simplification of the natural world. The climate models in current use span decades of engineering and understanding and are formed from a aggregate of climate model systems. Depending on the task at hand, a subset of the total models are used to generate a prediction.

In the endeavour to improve our models predictive capability, with a future goal to reduce the variance of prediction results, many iterations of the models are produced and run. Each iteration testing the effect of varies possible changes, for example: input parameters are adjusted; a more realistic representation for climate systems are used; and improved historical data is operated on. This results in a large collection of climate model outputs, with no discernible way of determining which offers the best prediction or insight . The task of deciding whether the changes have caused an improvement becomes a discussion point of a group of researchers, demanding a great portion of time. They are required to compare the merits of each model so that they can be ordered relatively. This process is one they hope to automate with the use of machine learning techniques

Yass bitch

.

Problem definition ::

1. As information in a domain increases, so does the complexity of the models used to model said domains.
2. Through the introduction other climate related components, large volumes of modes are produced and they are increasingly more complex. The task of identifying which model best represents the present and future becomes

more of a challenge.

3. Climate research is a vast field that is critical in predicting the violent hidden nature of weather systems, what environmental factors tend to increase this behaviour, and what we can expect from the natural world as we continue to inhabit and apply change to the world.
4. The met office produce hundreds of models every year that try and include new forms of statistical data with the intention of increasing our confidence in what the future holds.
5. These models are evaluated by a team of experts that look to see how well it predicts the known past with a hope in finding a model that shall predict the future of Earth's Climate.
6. These models however tend to have huge variance over their final outputs which highlights the fact the system is complex and hard to replicate.
7. Experts are looking to make use of machine learning techniques to help in the refinement of this process by replacing the cumbersome requirement for human time and discussion on what the merits of each model. The machine would help choose models by minicing experts opinions.
8. Along with that, a easy to use web tool that facilitates the interaction between the experts and the machine such that they might get the most out of its operation.

2 Solution specification

2.1 Web tool

1. Experts would like to be able to annotate their opinion on models.
2. Experts would like to upload models to be evaluated.
3. Evaluation should come with the functionality to toggle on and off other expert opinions.
4. Personal expert dashboards and examination.

3 Maintenance Plan

This section will cover what measures will be taken to ensure that the software artefact is well maintained throughout the project and after the project is completed. Maintaining the software artefact entails ensuring that all its features remain up to date and remain in the scope of the project and also ensures that the addition of new features does not break other features or render them obsolete.

An agile approach to development ensures good maintenance of the project on the macro scale. All new features are discussed and directly assessed by the problem owner so that they are guaranteed to be in scope of the project and the scope itself is also regularly revised. On top of this, the developers are split to manage different features of the project therefore ensuring that all features work together in a cohesive fashion. On the micro scale frequent testing using automated and manual methods further ensures the artefact is well maintained during the project.

Once the project is completed, the required long term maintenance is low. The learner will continue to improve using automated online learning and its performance can be monitored and validated by expert opinion to ensure that the learner is still accurate. Regular checks for bugs in the webtool and monitoring the performance of the server throughout automated checks is enough to maintain the artefact.

4 Cost/benefit analysis

As this is a research focused project, there is no plan to sell or monetise the intellectual property of the software artefact at the end of the project timeline. Rather, this project adheres to a more open source approach. The webtool and all it's functionality will be free to access and use and it is planned that the machine learners will improve over time from continual use after this project is completed. Therefore, in order to create a cost/benefit analysis which is relevant to this project, the most significant 'currency' counts as developer or climate expert time since time spent on this project is time taken away from other projects.

4.1 Development costs

A significant intial cost to this project and most other machine learning based projects is the time taken to pre-process an appropriate data set for training and testing of the learner. For this project a supervised learning approach is used therefore training and testing of the model requires labelled data. In the case of this project, the project owner (our expert) will supply the data (output from surface vegetation models) and must label each model output with an appropriate score. When training a machine learner, the larger the training (and testing) data set, the more reliable the learner predictions will be. Hence, this initial cost can be very large however the larger this cost, the better the pay-off is further down the development of the project. This cost can be partly offset by incrementally adding to the labelled data set as development progresses. To begin with, 100 labelled model outputs are used from preliminary testing of the prototype learner with the aim of increasing the size of the labelled set as much and as soon as possible.

This project follows an agile approach to development therefore frequent meetings with the project owner are necessary throughout the development phase of the project. Meetings with the project owner happen on a weekly basis thus creating an additional cost with regards to the project owners time. An agile approach to development also means that testing and maintenance of the software artefact happens simultaneously to development. And so the pre-completion maintenance costs of the software artefact must be taken into account. The work involved in the pre-completion maintenance of the software artefact is described in the aforementioned maintenance plan. The cost of testing the software artefact for this project factors in solely for user acceptance tests (UATs) as these require the project owner themselves to test an iteration of the product.

A dedicated server is required since the only interface between the user and the learner is the webtool. The server is required from the development phase so that development can take place in the appropriate environment and that reliable tests can be carried out. During the development of the software artefact, the webtool will be hosted on university provided server space therefore the server cost is the resources taken away from other potential uses.

The final significant cost to be considered is the time devoted by us, the developers, towards the completion of this project. Time spent on this project is time spent away from other potential projects.

4.2 Post-completion costs

Once the development phase is complete, there are still maintenance costs to consider. General upkeep of the webtool and planned or unplanned patches and improvements. Once the webtool is available to the wider scientific community there is the inevitability of required bug fixes to account for as well as planning for performance checks to ensure the learner is appropriately adjusting to new labelled data sets and making accurate predictions on 'real' data.

Post-completion the server keeping the webtool operational will be the main running cost. **Where will the webtool be hosted post-completion?** Scalability of the server must also be taken into account. **The server is potentially dealing with high loading due to model or learner size?**

The ability of the learner to carry out online learning means that training can continue post-completion. However this means that experts time is still required to label new data so that the prediction of the learner can be improved. However this cost has less impact compared to the development phase since the learner is already trained to an acceptable standard therefore any further training carried out post development, although still important has less impact on the performance of the learner.

4.3 Benefits

The goal of this project is to provide an efficient way of choosing the best models using a large selection of model outputs. The model outputs represent the observed effects of surface vegetation on carbon feedbacks over the last 200 years. These model outputs are relatively cheap to obtain as they are based on existing observational data hence they can be run a large number of times. However running these models into the future is expensive and requires greater computational power. This computational power is expensive to use therefore only the best models should be chosen. Creating an efficient method to select these models will save the expense of using sub-optimal models for future predictions.

Furthermore, by efficiently sorting model outputs, expert time is saved from manual sorting. Hence one of the main benefits of this project is to save expert time spent manually qualifying model outputs by using a machine learner to do this sorting automatically. The long term aim is that the expert time spent labelling data is largely outweighed by the time saved by this automation.

In an agile approach to software development, having the problem owner closely involved with the development process ensures that the scope of the project is well defined and that the software artefact is always meeting the problem owners requirements. Agile development also allows for flexibility of the project, new functionality can be added or removed throughout the development without major backtracking. This is valuable for this project in particular where the final requirements have a degree of flexibility and the requirements may change during the development process. And so the cost of the project experts time is outweighed by the benefit of a more efficient and flexible development process leading to a better end result.

Using an online learning approach means that the learner can be improved over time without input from the developers. Re-training of the model is not necessary and so a significant portion of learner maintenance is done automatically. New labelled data sets can only be supplied by trusted experts therefore the continuous learning is reliable.

Using Docker for this project offers easy scalability as the software artefact can run independently of its environment. Therefore the webtool can easily be ported from the university provided servers to another dedicated server. (https://success.docker.com/Architecture/Docker_Reference_Architecture%3A_Designing_Scalable%2C_Portable_Docker_Container_Networks)