Project ID: 116

Project Title

Paediatric Cancer Methylation Classifier Model Generation

Client Name

Ben Curran, James Bradley

Group Capacity

3 groups

Project Tags

Artificial Intelligence (Machine/Deep Learning, NLP), Big data Analytics and Visualization, Software Development

Company/Organization Name (Including Department)

Children's Cancer Institute of Australia, Computational Biology Group

Project Background

In the Computational Biology group at the Children's Cancer Institute, we have built a hierarchical classifier for paediatric cancers based on methylation data. The classifier is an ensemble of Random Forest models, with separate models being created and optimized for each node in a hierarchy. The current model-building process is restricted to a single server, and we have reached the limit of what a single server can do. As the number of samples increases the overhead for training new models becomes prohibitive. The challenge is to take the current model generation process and break it down to allow models to be generated in parallel as to opposed to sequentially. Generating models in parallel will significantly reduce the amount of time and allow us to explore different organisation of cancer types within the hierarchy which will allow us to more accurately model the underlying molecular data and provide more accurate classifications to our clinicians and curators.

Project Scope

The core scope of the project will be to build a model generation process that isolates tasks that can be executed in parallel and implement those tasks using either the current set of machine learning tools (scikit-learn) or platform optimised tools (sparkML). Data Isolation Data for each node in the hierarchy must be properly isolation to prevent the validation data leaking into training data and reducing the ability of each model to generalize to unseen patient data. Parallelization. During model generation, each model undergoes an extensive hyperparameter search to identify the optimal set of parameters. The identification of independent units of work within the hierarchical classifier creation process and restructuring them for concurrent execution will be required. Reproducibility

and Documentation
Establish reproducible workflows, record metadata, and produce documentation to support reuse, auditability and long-term maintainence.

Project Requirements

Task Isolation and Data Management
- Each node's model must be trained on an isolated subset of the data corresponding to that node.
- Data leakage between training, validation, and test sets must be strictly prevented.
- Metadata must record the data partitions used for each model to ensure reproducibility.
Hyperparameter Optimization
- The system must support automated hyperparameter search for each model
- Parallelization of hyperparameter searches across available compute resources must be supported. Output
- For each trained model, performance metrics, chosen hyperparameters, and training metadata must be stored in a structured and queryable format.
- Models must be saved in a standardized format (i.e. joblib)

Required Skills

Familiarity with python and git will be necessary. Some understanding of ML tools such as Scikit-learn would be useful. Data will be stored on our Databricks instance, which will also allow new virtual machines to be spun up and managed. Knowledge of this platform is not essential, but students must be willing to familiarise themselves with it.

Expected Outcomes

We would expect well documented source code for an application able to be scheduled to be periodically re-run to generate new models.

Disciplines

Software Development;Computer Science and Algorithms;Artificial Intelligence (Machine/Deep Learning, NLP);Big data Analytics and Visualization;Bioinformatics/Biomedical

Other Resources

Students will be provided access to
- a workspace on our Databricks instance. - git repository for code. - access to a suitable public data set