OpenDIEL Software Development and GUI

Omar Tafiti, Yan Yan Lam, Tze Hong Wong (Neptune), Rocco Febbo
Mentor: Kwai Wong
What is OpenDIEL?

The open Distributive Interoperable Executive Library (openDIEL) is a parallel workflow framework with built in communication and scalability across many nodes.

Put Simply

It can be thought of as a way of launching applications in an organized manner.
Problem with OpenDIEL

Process can be tedious while working on command line. Not entirely user friendly.

Is there a way to make running modules using openDIEL easier?
What is a GUI?

Short for graphical user interface, a GUI is an interface that uses icons or other visual indicators to interact with electronic devices, rather than only text via a command line.
Tkinter is a python GUI programming toolkit. Tkinter is included with python and is the most commonly used python GUI programming toolkit. Tkinter is easy to learn and extremely accessible. Must be familiar with python.
Functionality of the GUI

- Worked is saved using a saved data class that holds all information and is passed to every tab within the GUI.
- Modules are loaded through parsing of configuration file and new modules are created.
- Groups are created and workflow is saved.
- Configuration file is created.
- Number of processes is determined and `mpirun` command is called.
Some tasks are also not easy to achieve using Tkinter.

Stylistically, Tkinter is not the most visually appealing and laying out widgets in an attractive manner can prove to be difficult.
KIVY

Cross Platform  Open Source  Rapid development

Fresh  Flash  Flexible  Focused  Fun  Free
How to Install Kivy

1. Install Anaconda 3
2. Check if it is installed by typing “Python --version”
3. Make a virtual environment:
   - Type “conda create -n 'name’”
4. Install Kivy
   - Open terminal
   - Type “conda install -n 'name' -c conda-forge kivy”
Benefit of Kivy

- Mobile app development
- Automatically format widgets to most appealing design
- Visually appealing
- Good cooperation with different OS
- Interface Logic Separation
What Kivy files look like

Tree Diagram

- **Tab**
  - **Layout**
    - **Widget**
      - **Widget's Properties**
Tkinter

Kivy
Search a hyperparameter space with a neural network. With features such as:

- Freeing unnecessary resources
- Loading and resuming a previous search
- Add new Search methods and Trainee types
- Live visualization during training
Two Methods of Interface

Simple Grid
Modify a configuration file to change the search space by resolution of the space, area of the space, and parameter type.

Advanced
Add your own search method such as PBT or LCM by introducing a new search class into the C++ code.

```cpp
void grid_search_method::trainer_loop()
{
  vector< grid_layer > layers;
  vector< grid_param > parameters;
  /*
   * do some modification of the hyperparameters
   */
  for (size_t i = 0; i < n_trainees; i++)
    send_hyperparameters(i, &parameters);

  /*get the metrics */
  for (size_t i = 0; i < n_trainees; i++)
    trainer_metric t;
    recv_metrics(i, &t);

  /*send the trainees the done signal */
  for (size_t i = 0; i < n_trainees; i++)
    send_hyperparameters(i, NULL, NULL);
}
```

```cpp
void grid_search_method::trainee_loop()
{
  t->recv_hyperparameters();
  t->train();
  t->send_metrics();
}```
GUI tab: Grid Engine
A new interface to the Grid Engine with live preview of training process

GUI tab: Examples
Add each OpenDIEL Example to the GUI for easy testing and teaching

GUI tab: Ligggghts
Get the old GUI code for Ligggghts working in Kivy

GUI tab: Applications
Add each OpenDIEL Application to the GUI

Grid Engine Trainee Types
Add more trainee types such as Tensorflow

Grid Engine Search Methods
Add more search methods to the grid engine such as Population Based Training with configuration file interface
THANK YOU!
Any questions?