```
    os.environ['TF_CPP_MIN_LOG_LEVEL'] = str(
        tf.logging.__dict__[args.verbosity] / 10)

    # Run the training job
    hparams = hparam.HParams(**args.__dict__)
    train_and_evaluate(hparams)
```

Note the following in the preceding code:

- The method '_get_session_config_from_env_var()' defines the configuration for the runtime environment on Cloud MLE for the Estimator.
- The method 'train_and_evaluate()' does a number of orchestration events including
    - Routing training and evaluation datasets to the model function in 'model.py'
    - Setting up the runtime environment of the Estimator
    - Passing hyper-parameters to the Estimator model
- The line of code "if __name__ == '__main__':" defines the entry point of the Python script via the terminal session. In this script, the code will receive inputs from the terminal through the 'argparse.ArgumentParser()' method.

Training on Cloud MLE

The training execution codes are bash commands stored in a shell script. Shell scripts end with the suffix '.sh'.

Running a Single Instance Training Job

The bash codes for executing training on a single instance on Cloud MLE is shown in the following. Change the bucket names accordingly.

```
DATE=`date '+%Y%m%d_%H%M%S'`
export JOB_NAME=iris_$DATE
```