Zurich University
of Applied Sciences

Information Engineering 2

Bonus Material: Deep Learning Pipelines

Prof. Dr. Kurt Stockinger

Deep Learning Pipelines

- Use existing deep learning libraries (Keras, TensorFlow, PyTorch, XGBoost) with Spark

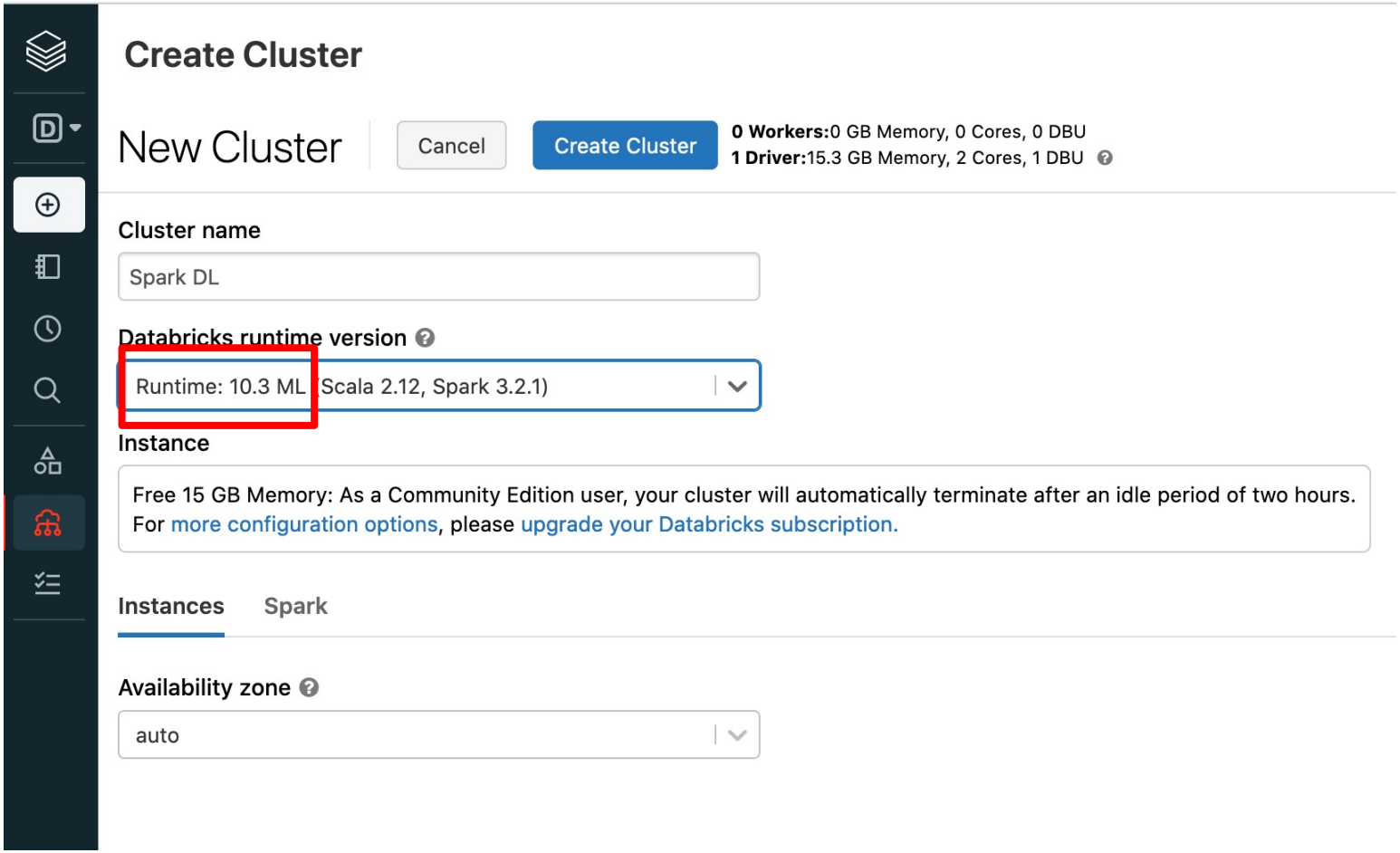
Deep Learning Guide

Databricks provides an environment that makes it easy to build, train, and deploy deep learning (DL) models at scale. Many deep learning libraries are available in Databricks Runtime ML, a machine learning runtime that provides a ready-to-go environment for machine learning and data science. For deep learning libraries not included in Databricks Runtime ML, you can either install libraries as a Databricks library or use init scripts to install libraries on clusters upon creation.

Graphics processing units (GPUs) can accelerate deep learning tasks. For information about creating GPU-enabled Databricks clusters, see GPU-enabled Clusters. Databricks Runtime includes pre-installed GPU hardware drivers and NVIDIA libraries such as CUDA.

Documentation and example notebooks: https://docs.databricks.com/applications/deep-learning/index.html

Databricks Community Edition: Runtime for Machine Learning



Deep Learning with Keras+Spark: Example Code Snippet

```
import tensorflow as tf
from tensorflow.keras.layers import Dense
from tensorflow.keras.models import Sequential
import mlflow
import mlflow.keras
import mlflow.tensorflow
```

Create the neural network

```
def create_model():
  model = Sequential()
  model.add(Dense(20, input_dim=8, activation="relu"))
  model.add(Dense(20, activation="relu"))
  model.add(Dense(1, activation="linear"))
  return model
```

Compile the model

```
model = create_model()

model.compile(loss="mse",
              optimizer="Adam",
              metrics=["mse"])
```

Detailed example:

https://docs.databricks.com/_static/notebooks/getting-started/get-started-keras-dbr7ml.html