AI Workshop – Lab 1

Setting Up Your AI Environment

Copyright (2019) by Terra Weather Pte Ltd. All rights reserved.

Lab Summary

- **Introduction to machine learning infrastructure and frameworks.**
- **Installating Smojo.**

Introduction

There are a number of modern frameworks for neural networks. There are two purposes for these frameworks:

- To aid the development of neural networks by humans (some popular examples are Tensorflow, Keras, Theano and Caffe)

- To efficiently run (ie train) these networks on GPUs (Graphical Processing Units) and CPUs. (examples are the opensource BLAS & Intel's MKL libraries and CuDNN) Of these, CuDNN specialises in providing highly tuned primitives for Nvidia's GPU products, while Intel's MKL libraries offer very fast performance on the CPU.

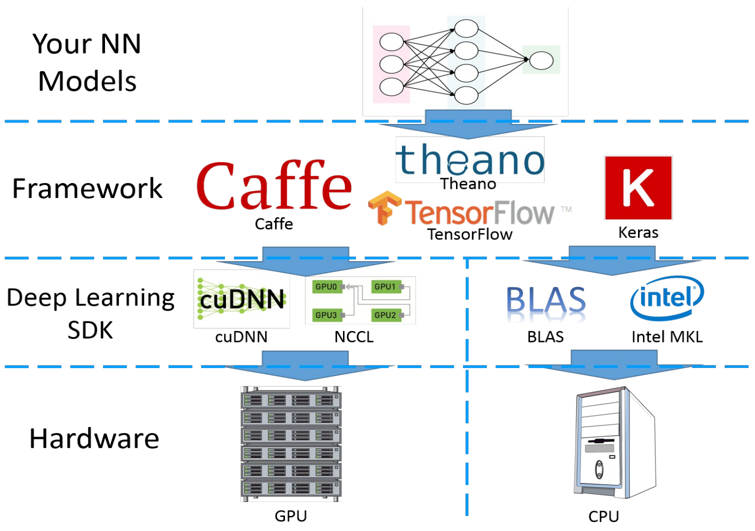


Figure 1.0 : Typical NN setup.

You won't need to install any of these software, since we have prepared the software infrastructure needed to run NNs for you.

Copyright (2019) by Terra Weather Pte Ltd. All rights reserved.

Autocaffe

Autocaffe is Terra-AI.SG's deep learning software. Autocaffe also includes automation, with which you can:

1. Automatically schedule NNs for running,
2. Automate the collection of performance statistics,
3. Automate visualization of numerous experiments in a single PDF report,
4. Automatically create and run NNs having different configuration parameters,
5. Automatically generate NNs from pre-defined templates (called prefabs). This can greatly simplify creating more complicated networks like stacked autoencoders.

This is very useful to systematically try many different configurations or if you need to train complex networks, as you will likely need to for this workshop.

For the data science challenge, you will not have to use Autocaffe, but it is needed for you to run the labs. There is no installation needed for Autocaffe as it is cloud-based.

Smojo

To run AI algorithms on Autocaffe, you will use the Smojo programming platform. We designed it to make creating, training and using AI models easy.

Smojo is a complete software development platform, and we have used it extensively for writing all production software at our parent company, Terra Weather.

We use Smojo because we have found that non-programmers find it easy to learn and use.

Smojo is free to use. In this lab, you will install Smojo on your laptop.

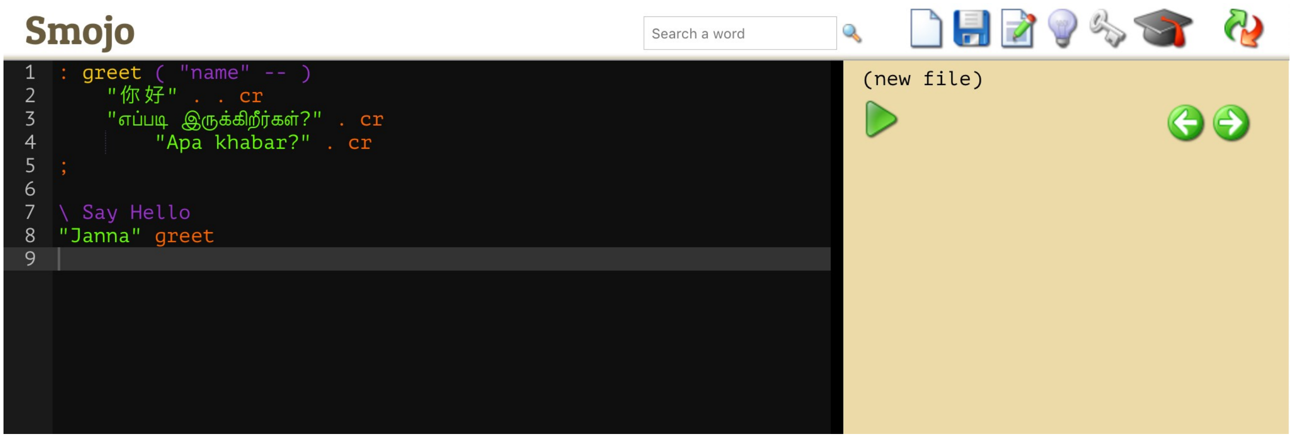
Installing Smojo

Step 1: Download the appropriate Smojo version from these links:

- Windows: https://1drv.ms/u/s!An58mTz89jfx2SmGQbrDf8lMZac5
- Mac: https://1drv.ms/u/s!An58mTz89jfx2SjI3MFaoH4NSKTj
- Linux: https://1drv.ms/u/s!An58mTz89jfx2SqrhK3tBCUwni07

Copyright (2019) by Terra Weather Pte Ltd. All rights reserved.

In the folder “ide”, click on “launch.exe” (Windows) or “launch” (Mac/Linux). This activates the Smojo server. For Windows only, it also launches a browser. For Mac/Linux, open a browser and use the URL: http://localhost:9080. You should see the Smojo GUI in your web browser after a few seconds. (see Figure 1.1)



```
: greet ( "name" -- )
    "你好" . . cr
    "எப்படி இருக்கிறீர்கள்?" . cr
        "Apa khabar?" . cr
;

\ Say Hello
"Janna" greet
```

Figure 1.1 : Smojo GUI

Step 2: Check that your setup is working: Click on the green “play” button. The results should appear on the right hand panel.

Troubleshooting:

- If your browser launches but you don't see the Smojo GUI, it could be that you need to install Java.
- If you accidentally close the browser, you can get the Smojo GUI by typing in the URL http://localhost:9080

To quit the Smojo server, right click on the icon on the system tray (Windows) or close the terminal window (Mac/Linux)

Obtain your Smojo Account

For all the lab sessions, you need a Smojo account. This is free, but you need to be invited by someone with an account. Please get further directions from your workshop instructor.

Copyright (2019) by Terra Weather Pte Ltd. All rights reserved.

Lab 1: Training a Neural Network

In this lab, we will go through the mechanics of training a neural network model.

DON'T PANIC!
You don't have to understand what you are doing just follow the instructions!
It will become clear in later labs.

Pay attention to:

- The types of files and their purpose,
- The steps involved in training an model and understanding its performance.

The Files

1. In your Smojo editor, click on the "Edit File" button.
2. Navigate to the folder "./dsc/lab-1"
3. There, you should see 5 files: config.m, lab-1.m, network.m, test.csv and train.csv.
4. Clicking on any one will open it in the black editor pane.

Each file serves a distinct purpose. You need to understand the purpose. Don't get confused by the code. Open up each file and read the commentary:

config.m

This file contains the settings for the models you want to train. Each setting generates a "Config" (short for Configuration). All configs are collected under a single **Project**. Your configuration files for a single project must be named **config.m**

lab-1.m

This file acts as the main "conductor" for all actions involved in training a model. This is the only file you should "run" using the green "play" button (or by typing Ctrl-R). We'll go through this step-by-step shortly. Don't run this file now.

network.m

This file contains the definition of the neural network model. Each model is defined in 3 parts: **train**, **deploy** and **solver**. You can see these three "words" in the file. Don't worry about syntax or how to define the models now. We'll do it later.

test.csv & train.csv

These are data files containing the test and train datasets. They are optional. Later, and in deployment, you will use "realtime" datasets which are from an API, not in static files. But we'll use these files for our simple experiments now.

1.1 Training the Model

Open up **lab-1.m**. Lines beginning with the slash (\) are comments and ignored by Smojo. They appear in deep purple in the editor. You can enable or disable a line by removing or adding a comment. Note: you must put a space after "\" (ie, "\ ") to make it a comment.

Step 1: Log into your Smojo account. You need internet access for this. Ensure that you have the Smojo cloud (lightbulb button) switched on.

Step 2: Test the model has no syntax errors. Uncomment the line:

```
test  network.m
```

Run the program again. You should see no error messages on the Output pane. If you click on the "Edit File" button, you should see 3 extra files:

```
debug-deploy.prototxt
debug-solver.prototxt
debug-train.prototxt
```

You can open these up. They contain the raw model definitions used by Autocaffe. You don't need these files. Delete them on the right Output panel by right-clicking and selecting "Delete".

After you've done this, re-comment the line "test network.m" so that it's purple again.

Copyright (2019) by Terra Weather Pte Ltd. All rights reserved.

Step 3: Save the training and test data to the server – you need to uncomment the lines:

```
"train.csv" csv>data
"test.csv" csv>data
```

Run the program. You should 2 messages on the Output pane indicating that the data has been saved.

Before proceeding to the next Step, <u>comment this line to prevent re-saving the data.</u>

Step 4: Train the Model – Uncomment the line:

```
train network.m
```

Then run the program. This submits the network(s) to Autocaffe for training. Before proceeding to the next Step, <u>comment this line to disable training!</u>

Step 5: Examine training & test losses – Uncomment the 2 lines:

```
1 train-loss . cr
1 test-loss  . cr
```

Smojo programs should be read like English, from left to right. In this case “1” is the Config you want to examine. You have only just 1 config in this setup (**Quiz**: in which file is this determined?). train-loss and test-loss are Smojo “words” (ie, functions) that connect to the Autocaffe server. The period word (ie, .) prints out the results and the word CR prints a new line.

The results are the losses from the “train” and “test” phases of training the model. The period prints out a “hash”, in the format:

iteration number = loss

Quiz: Look at the losses closely. Can you observe 2 important things about the losses?

Remember to comment the lines and save the file (Ctrl-S) when you're done.

Copyright (2019) by Terra Weather Pte Ltd. All rights reserved.

1.2 Multiple Configurations

You often need to run many different configurations of the same network. We'll do that in this section.

Step 1: Open up config.m. Comment the old configuration:

```
123 := maximum-iterations
```

This line sets the maximum-iterations variable to the single value "123". The word := takes its LHS and assigns it to the name on the RHS. You must include spaces, or you'll get an error.

Uncomment the line:

```
{{ 100 200 300 }} := maximum-iterations
```

The two words {{ and }} start and end a list definition. Note you need spaces between words. This line makes the name "maximum-iterations" take on 3 different values: 100, 200 and 300 in Configs 1, 2 and 3 respectively.

Be sure to save your changes.

Step 2: Open up lab-1.m and re-train the model as before. You don't have to "test" your model or save the data files since you've already done this. You may want to run the word:

```
clean
```

(all by itself) prior to training. This word cleans out old models for the current project previously created on your Autocaffe server. If you are re-training a model, be sure to clean it first or you may get bugs from old files.

Step 3: Examine the training and test losses for Config #3.

Quiz:

- Do the 2 observations you made earlier still hold up for Config #3?
- Which Config performs best?
- How did you determine the best performance?

Step 4: Plot the losses – Comment the lines for printing the losses and Uncomment the lines at the bottom to plot them:

```
include plot.m
1  3  plot-losses
```

In this line, losses from Config #1 to Config #3 are plotted on a single graph. You should see the loss graph appear in the output pane.

Step 5: Go back to the config.m file and comment the line to assign maximum-iterations. Instead, uncomment the line to train 5 models:

```
100 200 ... 5 take := maximum-iterations
```

Take note of the following:

- 100 200 … generates an infinite list, 100, 200, 300 …
- 5 take takes the first 5 terms from the infinite list.
- := assigns this 5-item list to the name “maximum-iterations”

Clean and Re-train the model and plot out all 5 losses in a single graph.

Quiz: What observations can you make about your models?

Copyright (2019) by Terra Weather Pte Ltd. All rights reserved.