Assignment 5 - TensorFlow with UNSW-NB15

Niranjan Jagannath

Data Preprocessing

The UNSW dataset contains 42 Features for each network packet, one label describing if it is malicious or normal, and a category which can be any of the 10 types: Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode and Worms and Normal.

1. Of the 42 features, 3 of them are categorical features (proto, service and state) which need to be encoded into numbers so that they can be analysed to make predictions. For this we use Pandas's `get_dummies()` function, which creates new dummy columns for each categorical feature. Due to this expansion, we now have 196 features to work with.

2. Next we need to make sure the features of the training and testing dataset are the same. (As categorical features different in both training and testing dataset, `get_dummies()` will generate different number of columns.)

3. Next we normalise all numerical features to scale them between 0 and 1 by using min max scaling.
$$x = \frac{x_i - min(x)}{max(x) - min(x)}$$
4. The Categories are also strings which need to be converted into one hot labels. We use Sklearn's `LabelEncoder()` to encode the strings into numbers and later convert them to one hot using tensorflow's one hot function.

Hyperparameters

1. **Minibatch size**: 9.
2. **Learning rate**: 0.0002 to 0.00005 with polynomial decay over all the epochs.
3. **Epochs**: 20
4. Neural network:
   Input(**196**) -> HIdden 1 (**100**) -> Hidden 2 (**50**) -> Output (**10**)

5. Each of the hidden layers has a **ReLU** activation applied to it to produce non linearity. This transforms the input into values usable by the output layer.
6. **Softmax** is applied to the output layer to get probabilities of categories. This also helps in learning with **cross entropy loss function**.
7. **Adam optimiser** is used for the back propagation.

Results:

Training results:
Label accuracy: **94.29% - 94.50%**
Category accuracy: **80.69% - 80.84%**

Testing results:
Label accuracy: **87.35% - 88.41%**
Category accuracy: **75.02% - 76.10%**

Requirements:

Python 3
Tensorflow
Numpy
Pandas
Sklearn