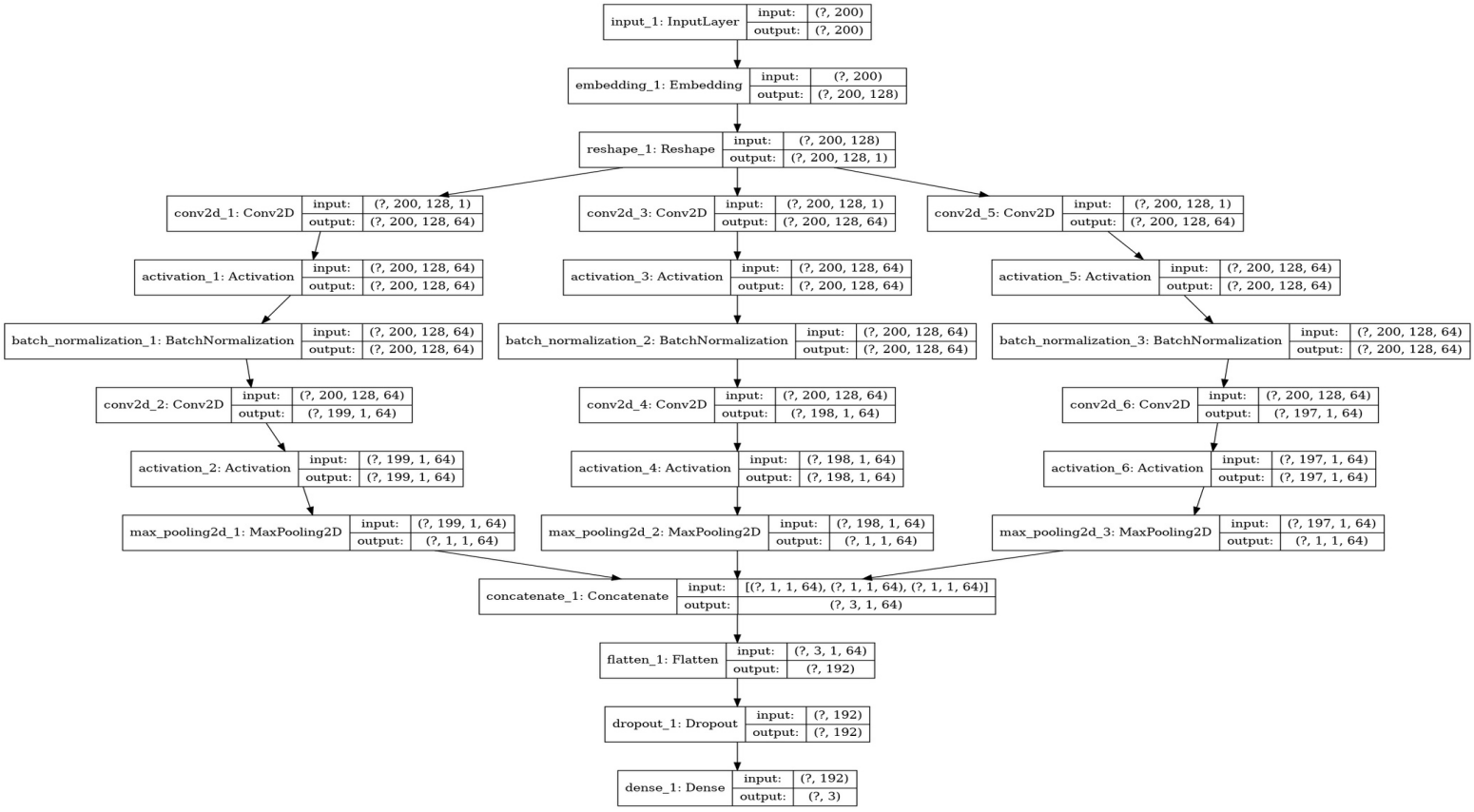
CNN LANGUAGE CLASSIFIER
Assignment 1

On this text classification assignment I used **Convolution Neural Network** model in which has shown promised results on some **NPL tasks** such as text classification.

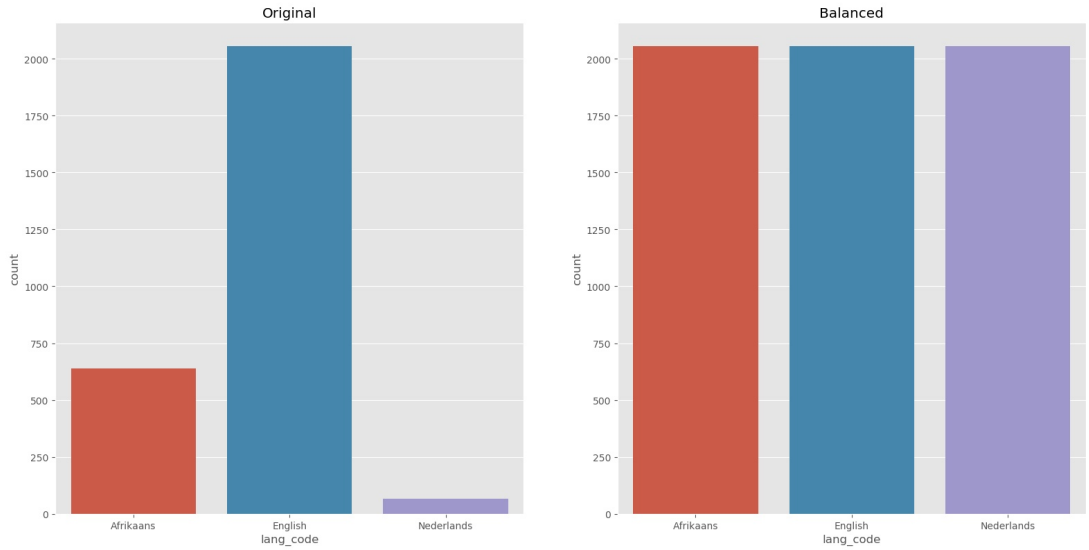
The model has been implemented in python using Keras and TensorFlow libraries. It's compose of an embedding layer with 512 of dimension and 3 convolution branches where each branch applies convolution over 3, 4 and 5 words, then merged in fully connected network fashion.

CNN model architecture:



Data Analysis:

The dataset consist of **2761 rows** (after remove null values) and **3 labels – Afrikaans, English and Nederlands** where the **dominant samples are labeled as English** and **few samples as Nederlands**. Classification problems tends to work poorly on imbalanced dataset. For this porprose it has been applied sampling strategy where each other classes are sampled to align with English samples count.



Data Cleaning:

I removed from data-frame the entire row where **column** value is nan and/or sentence has duplicated labels.

	text	label count
34	A dog is a man's best friend	2
76	A rolling stone gathers no moss	2
204	s different as chalk and cheese	2
209	As fit as a butcher's dog	2
960	For all intents and purposes	2
1399	It's not rocket science	2
2179	So maklik soos brood en botter.	2
2291	Take potluck	2
2372	The law is an ass	2

Data Pre-processing:

Using keras Tokenizer the entire sentence list are converted into list of list of codes where each code represents a word, then padded to same length using pad_sequences function. Similar conversion technique is applied for labels where each label is represent as an integer index (0 - Afrikaans, 1 – English, 2 – Nederlands) and then converted into categorical representation ([1,0,0] - Afrikaans, [0,1,0] – English, [0,0,1] – Nederlands).

	text	label
	Ship shape and Bristol fashion	English
Encoded	[33 1 33 2 19 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]	[0. 1. 0.]

Train Model

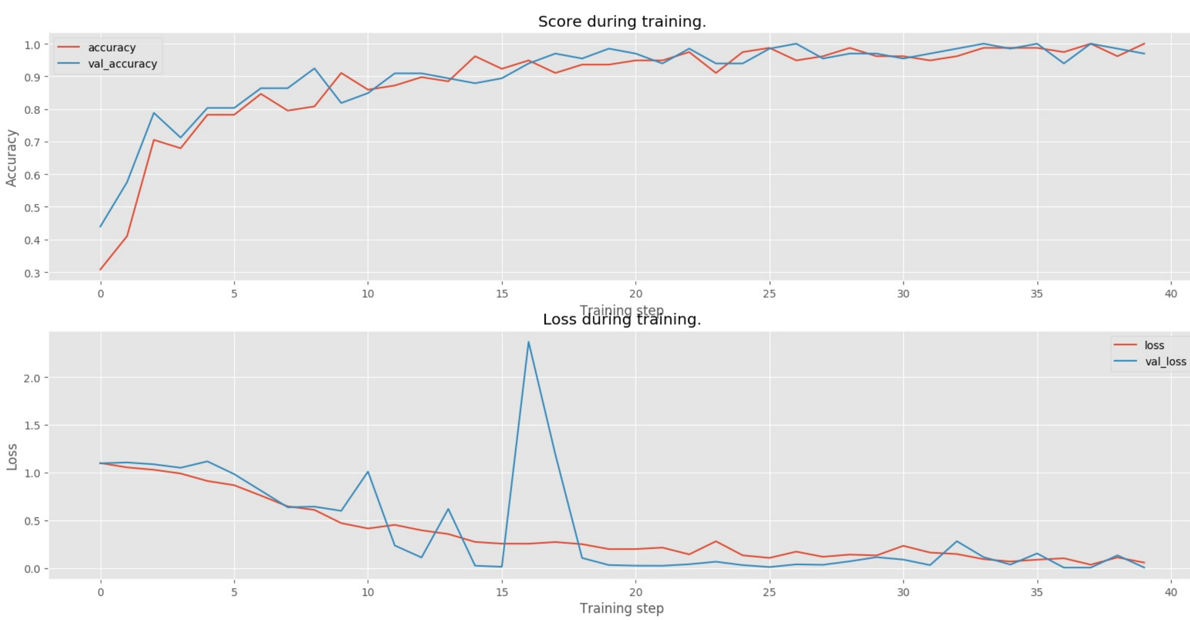
Using tensorflow Dataset class, samples are shuffle and split into:

- 60 % train
- 30 % validation
- 10 % test

During training process I manually adjust hyper-parameters such as embedding dimension (embed_dim), filter_sizes, dropout, optimizer and it's learning rate to improve validation and test accuracy. Semi automated algorithm can be used such as scikit-learn function GridSearchCv for fine-tunning hyper-parameters.

Model Train Accuracy:

The model has been trained for 40 epochs, with embedding dimension of 128 and 64 filters per convolution layer and validation accuracy reached 99%.



Model Test Accuracy: