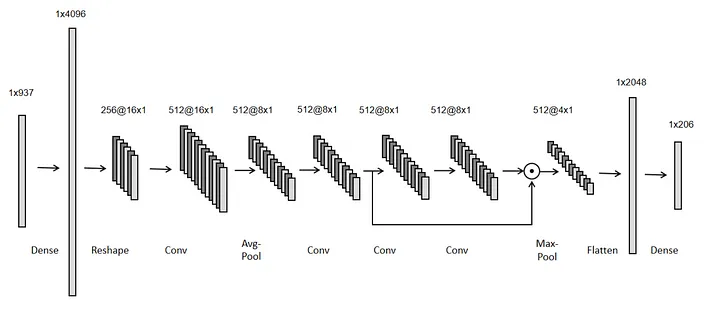
**Overview**

**Convolution Neural Network:**

* Imported all the necessary packages
* The pre processed data is loaded as df
* The search space for hyper parameter tuning is defined to find the optimal parameters that gives best performance
* Here we have used **Keras Tuner** and **RandomSearch** to perform hyperparameter tuning with the defined search space
* Here we have used 1D-CNN

**1D-CNN**

* CNN structure performs well in feature extraction, but it is rarely used in tabular data because the correct features ordering is unknown.
* The idea is to reshape the data directly into a multi-channel image format, and the correct sorting can be learned by using Fully Connected layer through back propagation.



(NOTE: The image is downloaded from internet – it illustrates the similar architecture, not the exact number of layers.)

* The best model and its configuration is then saved
* Then the model performance is evaluated with accuracy and loss values, precision, recall, f1 Score with the help of confusion matrix

**Tabnet**

* Imported all the necessary packages
* The pre processed data is loaded as df
* The search space for hyper parameter tuning is defined to find the optimal parameters that gives best performance
* Here we have used **Keras Tuner** and **RandomSearch** to perform hyperparameter tuning with the defined search space
* Tabnet is based on attention mechanism.
* Tabnet converts the categorical values into embeddings

Brief on Tabnet - <https://syslog.ravelin.com/classification-with-tabnet-deep-dive-49a0dcc8f7e8>