

PROJECT DEVELOPMENT DELIVERY OF SPRINT 1

TEAM ID	PNT2022TMID03174
PROJECT NAME	Smart solutions for Railways

Sprint 1 import

```
tensorflow as tf import
```

```
sys import os
```

```
# Disable tensorflow compilation warnings
```

```
os.environ['TF_CPP_MIN_LOG_LEVEL']='2'
```

```
import tensorflow as tf image_path =
```

```
sys.argv[1] # Read the image_data
```

```
image_data = tf.gfile.GFile(image_path, 'rb').read()
```

```
# Loads label file, strips off carriage return label_lines
```

```
= [line.rstrip() for line
```

```
    in tf.gfile.GFile("logs/trained_labels.txt")]
```

```
# Unpersists graph from file with
```

```
tf.gfile.GFile("logs/trained_graph.pb", 'rb') as f:
```

```
    graph_def = tf.GraphDef()
```

```
graph_def.ParseFromString(f.read())
```

```
    _ = tf.import_graph_def(graph_def, name="")
```

with tf.Session() as sess:

 # Feed the image_data as input to the graph and get first prediction

softmax_tensor = sess.graph.get_tensor_by_name('final_result:0')

predictions = sess.run(softmax_tensor, \

 {'DecodeJpeg/contents:0': image_data})

 # Sort to show labels of first prediction in order of confidence

top_k = predictions[0].argsort()[-len(predictions[0]):][::-1] for

node_id in top_k:

 human_string = label_lines[node_id]

score = predictions[0][node_id]

 print('%s (score = %.5f)' % (human_string, score))