Trustpilot

Deep learning on ML Engine

Balázs Tóth
Software Engineer at Trustpilot

Trustpilot

About me

- Software engineer at Trustpilot
- Applied machine learning to detect fake reviews and fraudulent user behavior

Agenda

Overview of an ML engine service in the fraud detection context

A tutorial on training neural networks with Keras on ML Engine

The problem

- Limitations of your local computer
- Limitations of your time

Why do we use ML Engine in Trust and Transparency?

Why do we use ML Engine in Trust and Transparency?

- Train multiple models in parallel
- Easy access to powerful computers
- Easy to deploy and maintain models
- Hyperparameter optimization
- Good ecosystem with Dataflow and BigQuery

Why do we use ML Engine in Trust and Transparency?

- Train multiple models in parallel
- Easy access to powerful computers
- Easy to deploy and maintain models
- Hyperparameter optimization
- Good ecosystem with Dataflow and BigQuery

Why do we use ML Engine in Trust and Transparency?

- Train multiple models in parallel
- Easy access to powerful computers
- Easy to deploy and maintain models
- Hyperparameter optimization
- Good ecosystem with Dataflow and BigQuery

Why do we use ML Engine in Trust and Transparency?

Trustpilot

- Train multiple models in parallel
- Easy access to powerful computers
- Easy to deploy and maintain models
- Hyperparameter optimization
- Good ecosystem with Dataflow and BigQuery

Why do we use ML Engine in Trust and Transparency?

Trustpilot

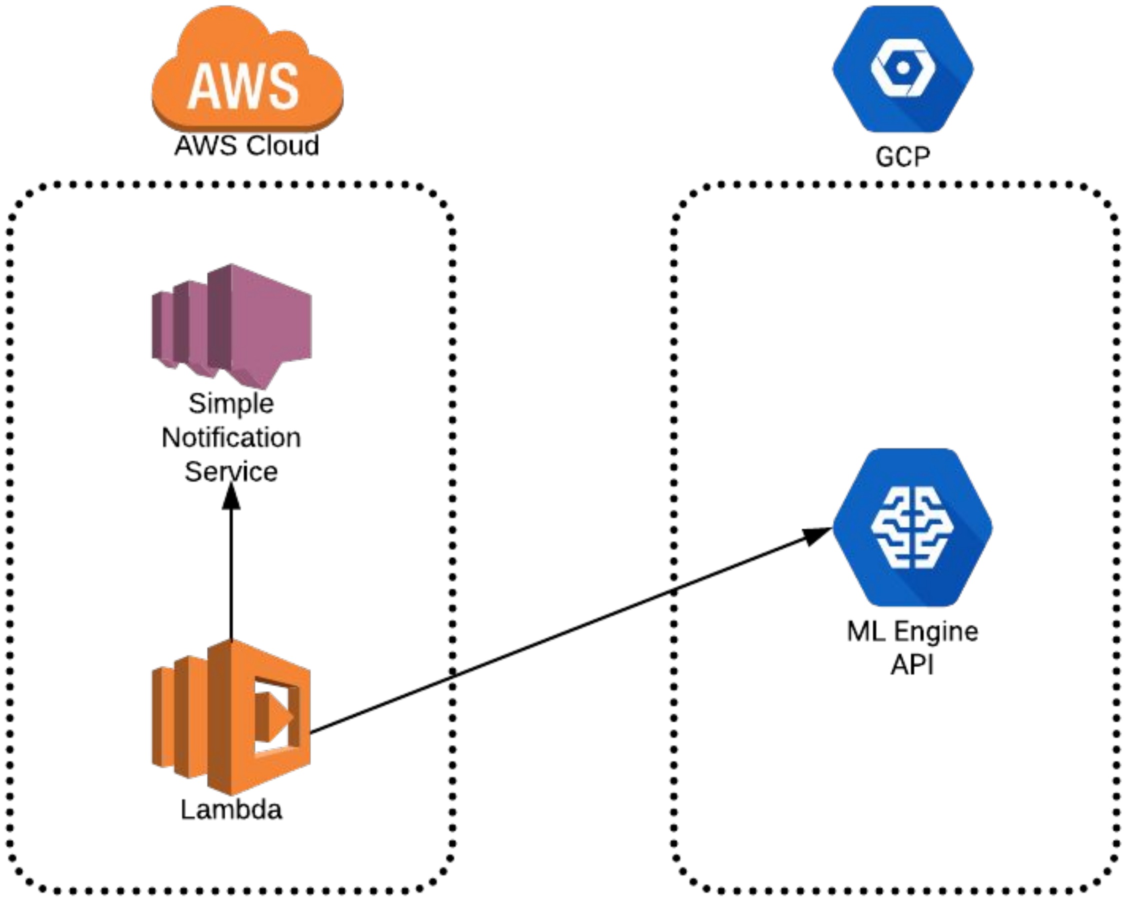
- Train multiple models in parallel
- Easy access to powerful computers
- Easy to deploy and maintain models
- Hyperparameter optimization
- Good ecosystem with Dataflow and BigQuery

Overview of an ML Engine Service

Trustpilot

Spammer adversaries in Trustpilot

Solution: Spam review detector


AWS
AWS Cloud
Simple Notification Service
Lambda
GCP
ML Engine API

Evolution of spam reviews with the iterations of the spam review detector

Machine learning workflow

- Preprocessing - Google Dataflow
- Training and evaluation - Google ML Engine
- Deployment of the model - Google ML Engine API
- Batch prediction (2nd part of evaluation) - Google ML Engine

Machine learning workflow

- Preprocessing - Google Dataflow
- Training and evaluation - Google ML Engine
- Deployment of the model - Google ML Engine API
- Batch prediction (2nd part of evaluation) - Google ML Engine

Machine learning workflow

- Preprocessing - Google Dataflow
- Training and evaluation - Google ML Engine
- Deployment of the model - Google ML Engine API
- Batch prediction (2nd part of evaluation) - Google ML Engine

Machine learning workflow

- Preprocessing - Google Dataflow
- Training and evaluation - Google ML Engine
- Deployment of the model - Google ML Engine API
- Batch prediction (2nd part of evaluation) - Google ML Engine

A tutorial on training neural networks with Keras on ML Engine

Example application: Classification of newsgroups

Public Dataset:
http://qwone.com/~jason/20Newsgroups/

Code:
https://github.com/tothbalazs0920/ml-engine-example

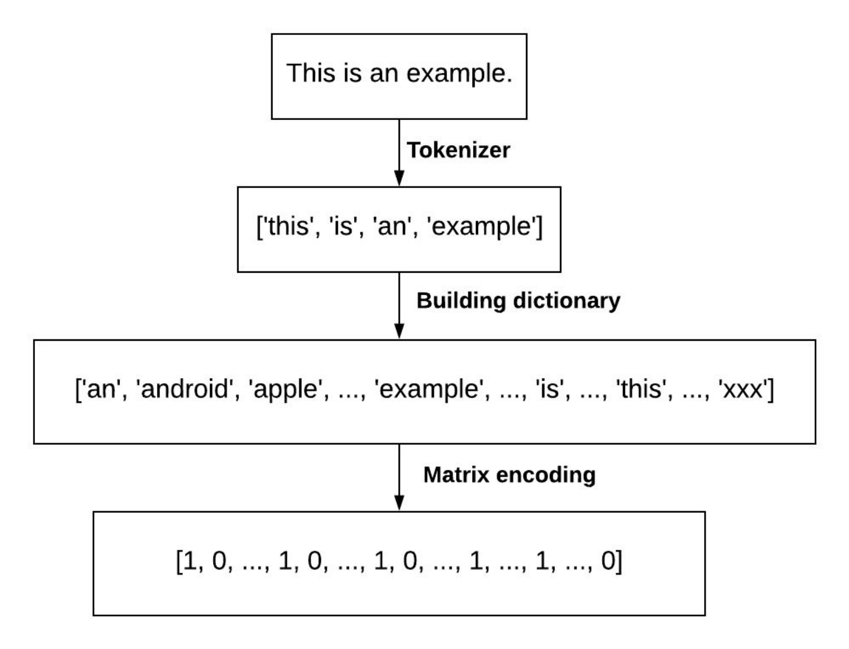
Use only 4 groups

- atheism
- religion
- graphics
- science

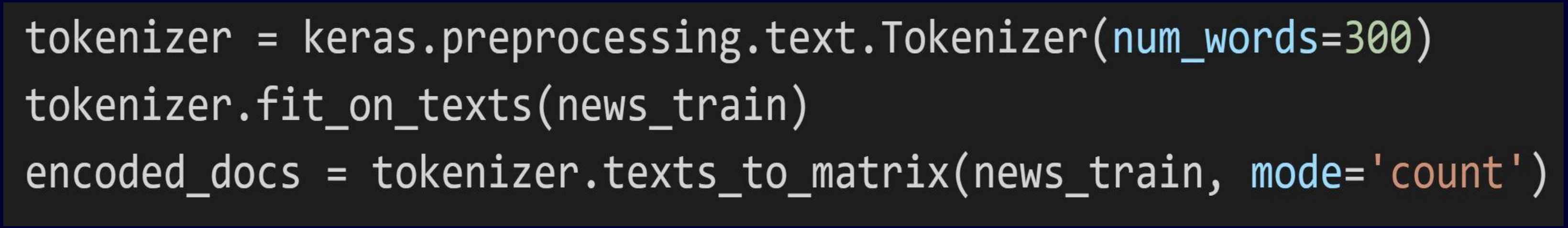
1. Setup a project on Google Cloud Console

- Enable ML Angine API’s
- Install Google cloud command line tools

2. Preprocess the data - bag of words

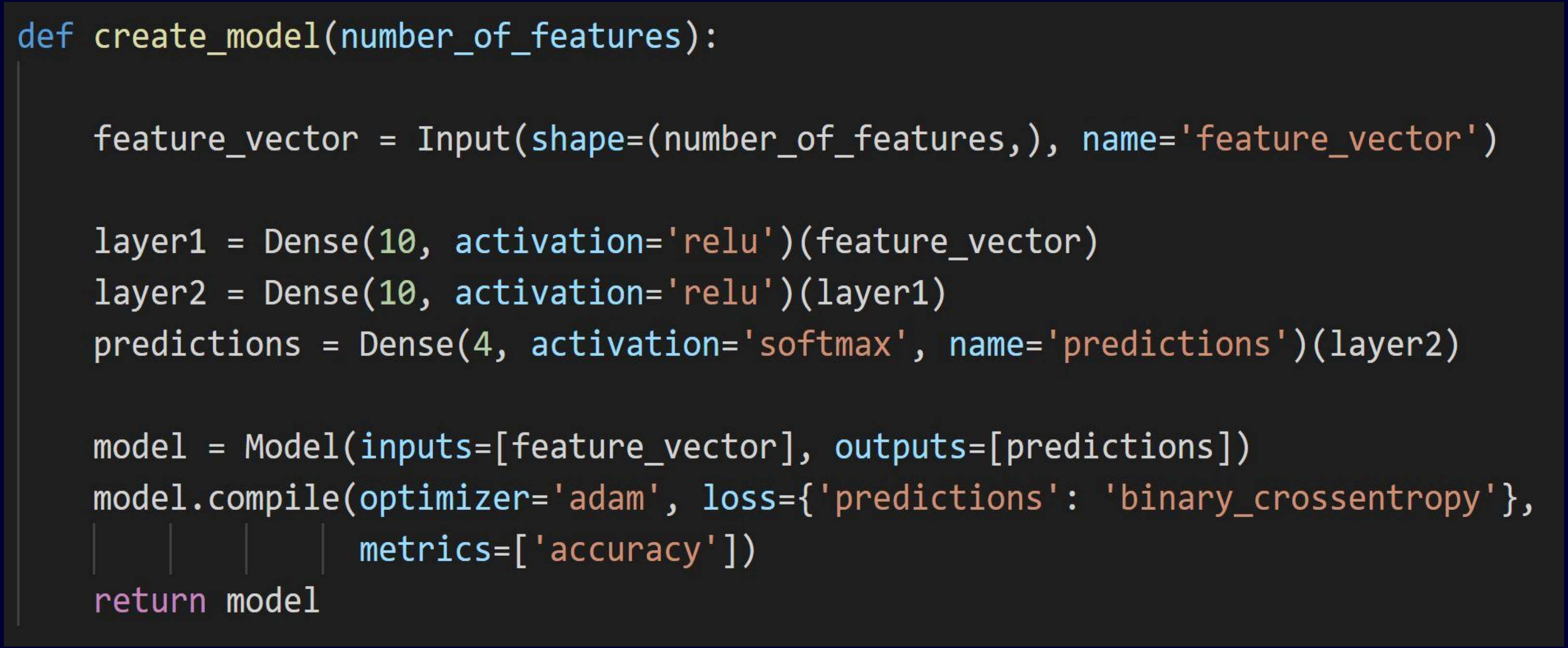


2. Preprocess the data and upload the preprocessed data to a gcs



```
tokenizer = keras.preprocessing.text.Tokenizer(num_words=300)
tokenizer.fit_on_texts(news_train)
encoded_docs = tokenizer.texts_to_matrix(news_train, mode='count')
```

3. Create a Keras model



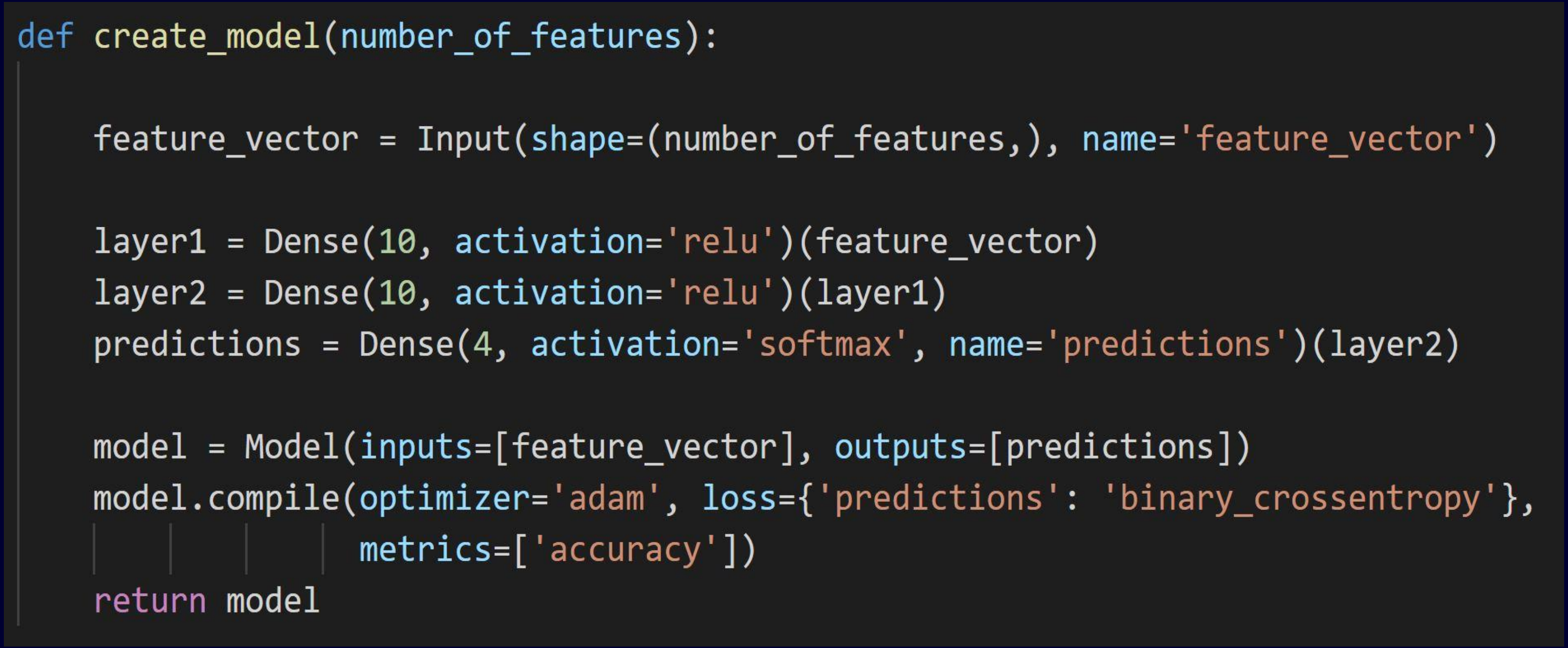
```
def create_model(number_of_features):

    feature_vector = Input(shape=(number_of_features,), name='feature_vector')

    layer1 = Dense(10, activation='relu')(feature_vector)
    layer2 = Dense(10, activation='relu')(layer1)
    predictions = Dense(4, activation='softmax', name='predictions')(layer2)

    model = Model(inputs=[feature_vector], outputs=[predictions])
    model.compile(optimizer='adam', loss={'predictions': 'binary_crossentropy'},
                  metrics=['accuracy'])
    return model
```

3. Create a Keras model



```python
def create_model(number_of_features):

    feature_vector = Input(shape=(number_of_features,), name='feature_vector')

    layer1 = Dense(10, activation='relu')(feature_vector)
    layer2 = Dense(10, activation='relu')(layer1)
    predictions = Dense(4, activation='softmax', name='predictions')(layer2)

    model = Model(inputs=[feature_vector], outputs=[predictions])
    model.compile(optimizer='adam', loss={'predictions': 'binary_crossentropy'},
                  metrics=['accuracy'])
    return model
```

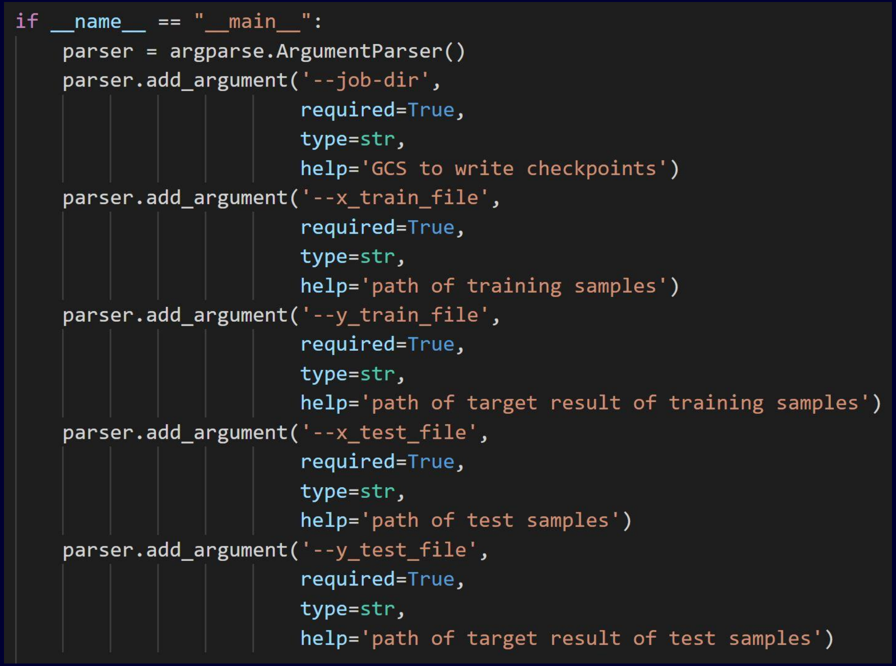
4. Set up the training in Keras

- model.fit() and use a machine with a lot of memory (256 GB)
- model.fit_generator() and use a machine with less memory

4. Set up the training in Keras

- model.fit() and use a machine with a lot of memory (256 GB)
- model.fit_generator() and use a machine with less memory

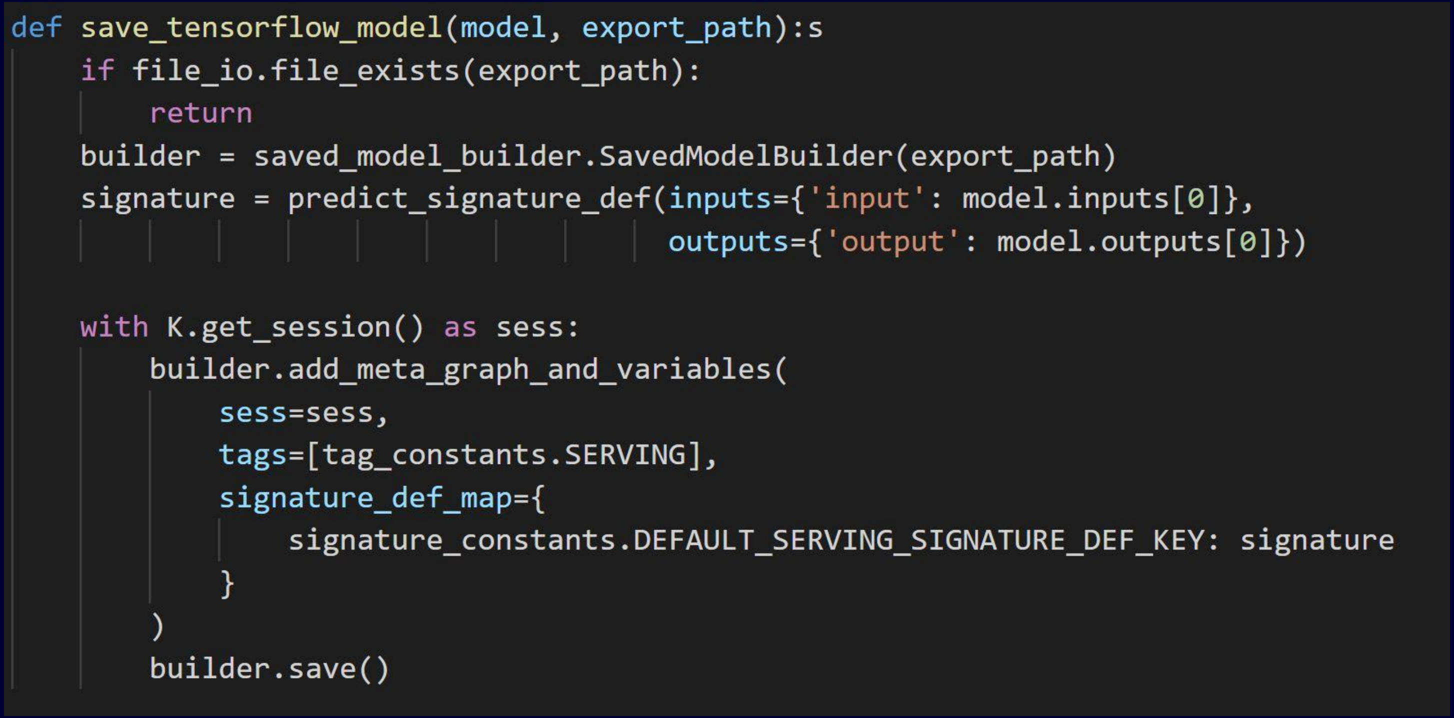
5. Add arguments to the entry file



```python
if __name__ == "__main__":
    parser = argparse.ArgumentParser()
    parser.add_argument('--job-dir',
                        required=True,
                        type=str,
                        help='GCS to write checkpoints')
    parser.add_argument('--x_train_file',
                        required=True,
                        type=str,
                        help='path of training samples')
    parser.add_argument('--y_train_file',
                        required=True,
                        type=str,
                        help='path of target result of training samples')
    parser.add_argument('--x_test_file',
                        required=True,
                        type=str,
                        help='path of test samples')
    parser.add_argument('--y_test_file',
                        required=True,
                        type=str,
                        help='path of target result of test samples')
```

6. Convert the keras model to tensorflow model after training

Trustpilot



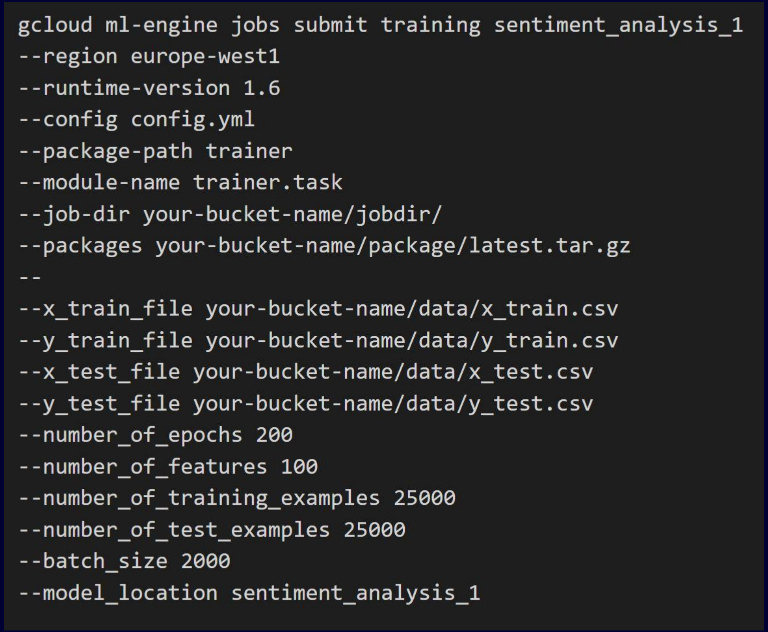
```python
def save_tensorflow_model(model, export_path):s
    if file_io.file_exists(export_path):
        return
    builder = saved_model_builder.SavedModelBuilder(export_path)
    signature = predict_signature_def(inputs={'input': model.inputs[0]},
                                      outputs={'output': model.outputs[0]})

    with K.get_session() as sess:
        builder.add_meta_graph_and_variables(
            sess=sess,
            tags=[tag_constants.SERVING],
            signature_def_map={
                signature_constants.DEFAULT_SERVING_SIGNATURE_DEF_KEY: signature
            }
        )
        builder.save()
```

7. Create a python package and upload it to a gcs

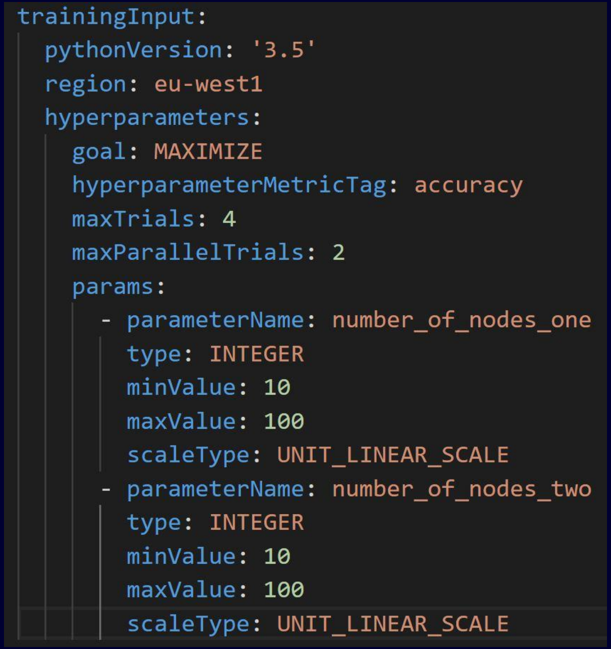
```
python setup.py sdist --formats=gztar
```

8. Submit a training job to ML Engine



```
gcloud ml-engine jobs submit training sentiment_analysis_1
--region europe-west1
--runtime-version 1.6
--config config.yml
--package-path trainer
--module-name trainer.task
--job-dir your-bucket-name/jobdir/
--packages your-bucket-name/package/latest.tar.gz
--
--x_train_file your-bucket-name/data/x_train.csv
--y_train_file your-bucket-name/data/y_train.csv
--x_test_file your-bucket-name/data/x_test.csv
--y_test_file your-bucket-name/data/y_test.csv
--number_of_epochs 200
--number_of_features 100
--number_of_training_examples 25000
--number_of_test_examples 25000
--batch_size 2000
--model_location sentiment_analysis_1
```

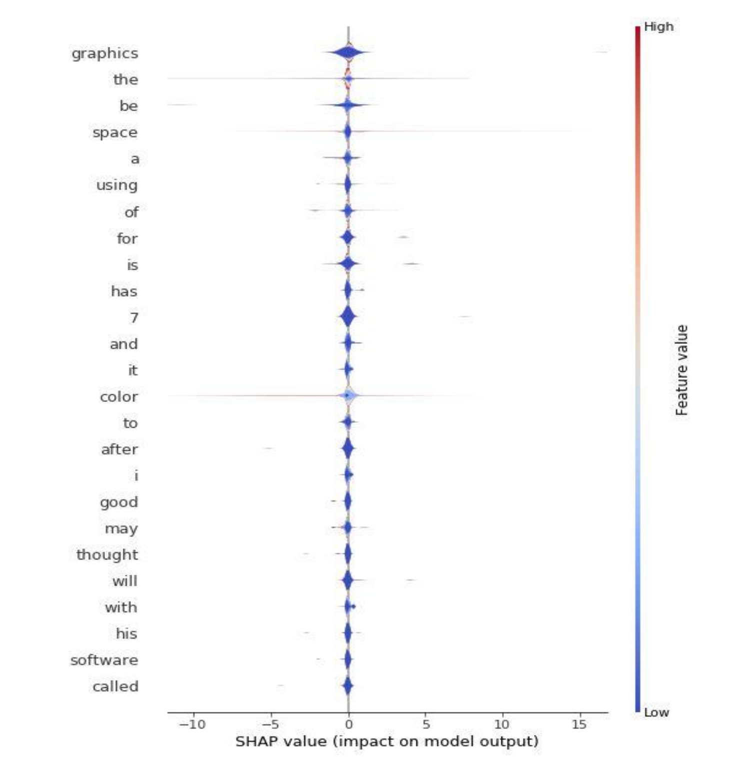
Hyperparameter tuning



```yaml
trainingInput:
  pythonVersion: '3.5'
  region: eu-west1
  hyperparameters:
    goal: MAXIMIZE
    hyperparameterMetricTag: accuracy
    maxTrials: 4
    maxParallelTrials: 2
    params:
      - parameterName: number_of_nodes_one
        type: INTEGER
        minValue: 10
        maxValue: 100
        scaleType: UNIT_LINEAR_SCALE
      - parameterName: number_of_nodes_two
        type: INTEGER
        minValue: 10
        maxValue: 100
        scaleType: UNIT_LINEAR_SCALE
```

Trustpilot

Interpreting the model with SHAP



Thank you

Email: bat@trustpilot.com
LinkedIn: https://www.linkedin.com/in/balazs-toth-6186005a/