Source Code Documentation

Twitter Sarcasm Classification Challenge

Sahil Rishi - sahilr2@illinois.edu

```
from simpletransformers.classification import ClassificationModel
```

This is the library we are using for transfer learning.

```
train_args={
    'reprocess_input_data': True,
    'overwrite_output_dir': True,
    'num_train_epochs': 8,
    'fp16': False,
    'sliding_window': False,
    'learning_rate': 3e-05,
    'max_seq_length': 256,
    'do_lower_case': True,
    'train_batch_size': 8,
    'evaluate_during_training': False
}
```

```
model = ClassificationModel('distilbert', 'distilbert-base-uncased',
args=train_args, use_cuda=False)
```

Train_args: It specifies the parameters for the model. We define the learning rate, epoch and max_seq_length here.

We also define the do_lower_case as **true** as we use the uncased model.
For final training, we train on all the data so 'evaluate_during_training' is False.

```
model = ClassificationModel('distilbert', '/content/drive/MyDrive/Colab
Notebooks/model.zip (Unzipped Files)/checkpoint-5000-epoch-8',
args=train_args, use_cuda=False)
```

We can load a saved model via this API. Just put the path of the model in.

The Classification Model API handles

Model instantiation
Saving per epoch
Training loops
Prediction Loops
Converting/Preprocessing of data

So we don’t have to explicitly handle all these details.