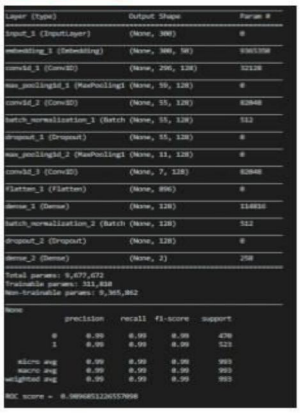
Classifying Native Advertisements and Sponsored Content with NN

Dellarontay Readus
Michael Kaplan

- Native advertisements are sponsored content produced by a business or institution to appear as if they belong with the content a user regularly views
- These advertisements are sometimes difficult for a human being to discern, so we decided to create a Convolutional NN model to classify an html page's content as sponsored or non-sponsored
- Our model was able to correctly classify labels from the test dataset to 97% accuracy.

Model Summary and ROC Curve



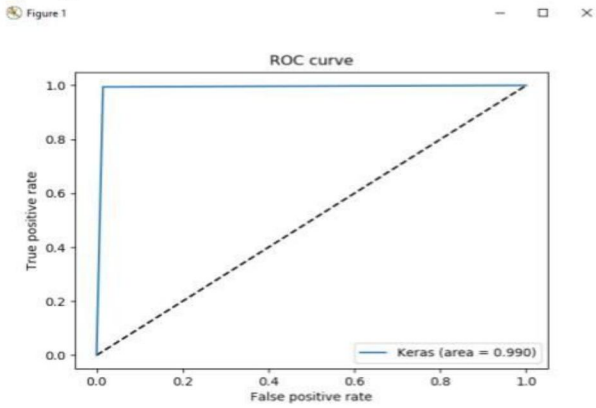
Layer (Type)	Output Shape	Param #
input_1 (InputLayer)	(None, 300)	0
embedding_1 (Embedding)	(None, 300, 50)	9365350
conv1d_1 (Conv1D)	(None, 296, 128)	32128
max_pooling1d_1 (MaxPooling1	(None, 59, 128)	0
conv1d_2 (Conv1D)	(None, 55, 128)	82048
batch_normalization_1 (Batch	(None, 55, 128)	512
dropout_1 (Dropout)	(None, 55, 128)	0
max_pooling1d_2 (MaxPooling1	(None, 11, 128)	0
conv1d_3 (Conv1D)	(None, 7, 128)	82048
flatten_1 (Flatten)	(None, 896)	0
dense_1 (Dense)	(None, 128)	114816
batch_normalization_2 (Batch	(None, 128)	512
dropout_2 (Dropout)	(None, 128)	0
dense_2 (Dense)	(None, 2)	258

Total params: 9,677,672
Trainable params: 311,810
Non-trainable params: 9,365,862

None

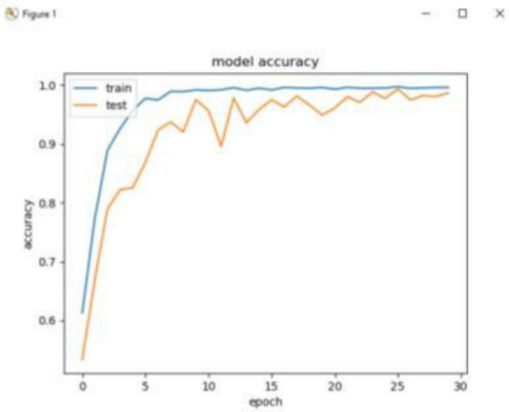
	precision	recall	f1-score	support
0	0.99	0.99	0.99	470
1	0.99	0.99	0.99	523
micro avg	0.99	0.99	0.99	993
macro avg	0.99	0.99	0.99	993
weighted avg	0.99	0.99	0.99	993

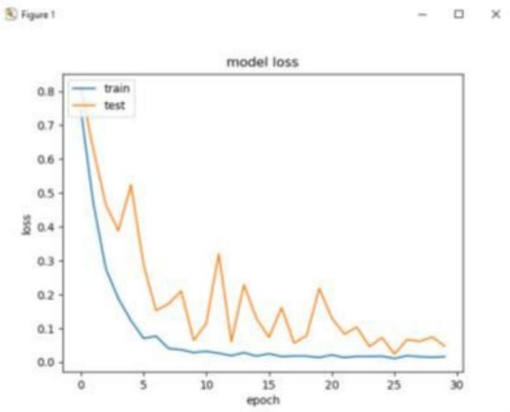
ROC score = 0.9994051126557998

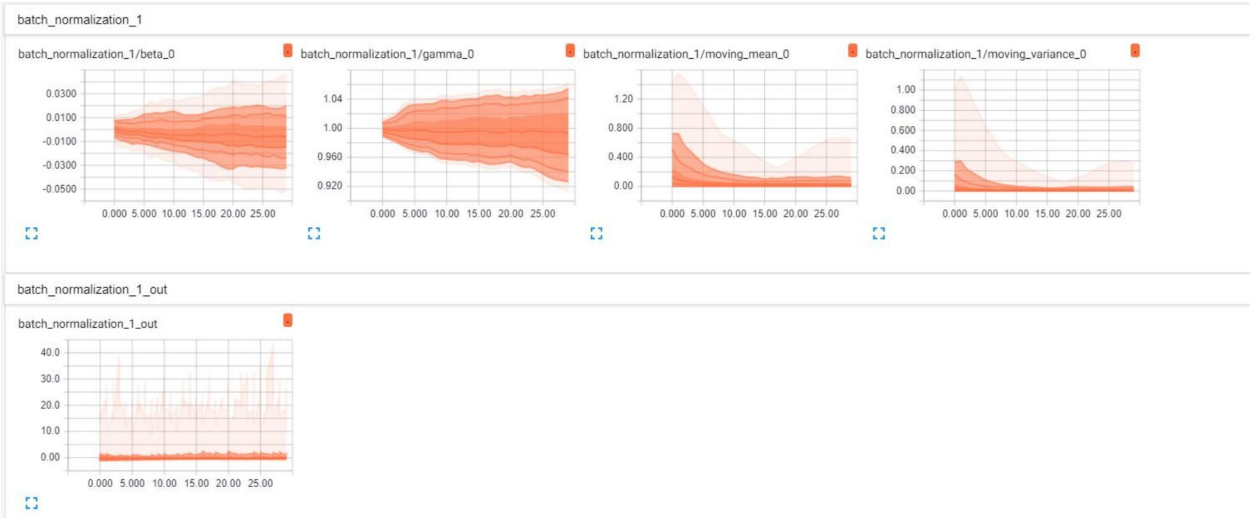


```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         (None, 300)               0
_________________________________________________________________
embedding_1 (Embedding)      (None, 300, 50)           9365350
_________________________________________________________________
conv1d_1 (Conv1D)            (None, 296, 128)          32128
_________________________________________________________________
max_pooling1d_1 (MaxPooling1 (None, 59, 128)           0
_________________________________________________________________
conv1d_2 (Conv1D)            (None, 55, 128)           82048
_________________________________________________________________
batch_normalization_1 (Batch (None, 55, 128)           512
_________________________________________________________________
dropout_1 (Dropout)          (None, 55, 128)           0
_________________________________________________________________
max_pooling1d_2 (MaxPooling1 (None, 11, 128)           0
_________________________________________________________________
conv1d_3 (Conv1D)            (None, 7, 128)            82048
_________________________________________________________________
flatten_1 (Flatten)          (None, 896)               0
_________________________________________________________________
dense_1 (Dense)              (None, 128)               114816
_________________________________________________________________
batch_normalization_2 (Batch (None, 128)               512
_________________________________________________________________
dropout_2 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_2 (Dense)              (None, 2)                 258
=================================================================
Total params: 9,677,672
Trainable params: 311,810
Non-trainable params: 9,365,862
_________________________________________________________________
None
              precision    recall  f1-score   support

           0       0.99      0.99      0.99       470
           1       0.99      0.99      0.99       523

   micro avg       0.99      0.99      0.99       993
   macro avg       0.99      0.99      0.99       993
weighted avg       0.99      0.99      0.99       993

ROC score =  0.9896851226557098
```

Model Accuracy & Model Loss

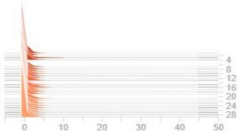






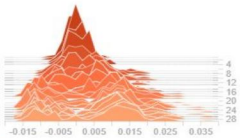
batch_normalization_1_out

batch_normalization_1_out

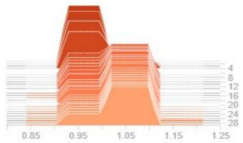


batch_normalization_2

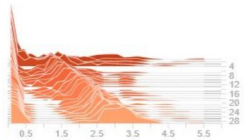
batch_normalization_2/beta_0



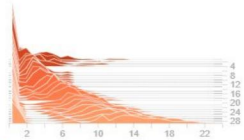
batch_normalization_2/gamma_0

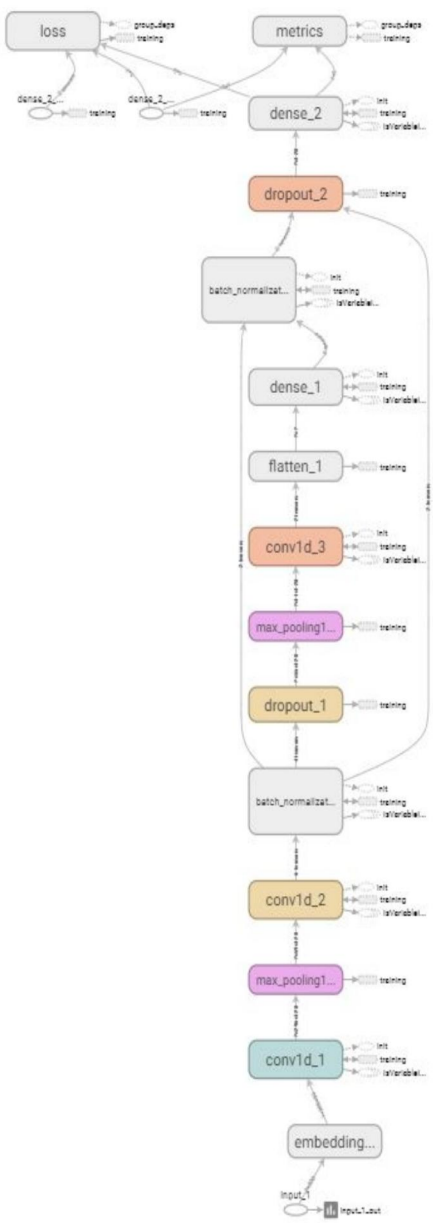


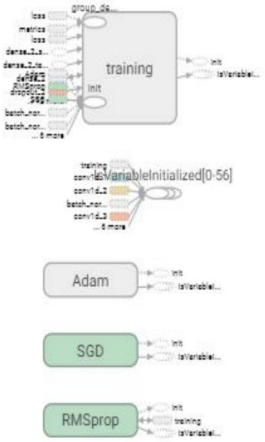
batch_normalization_2/moving_mean_0



batch_normalization_2/moving_variance_0




loss
metrics
dense_2
dropout_2
batch_normalizat...
dense_1
flatten_1
conv1d_3
max_pooling1...
dropout_1
batch_normalizat...
conv1d_2
max_pooling1...
conv1d_1
embedding...
input_1


training
Adam
SGD
RMSprop