In [1]:
```
#library needed for text class prediction
import spacy

#library to load and process data
import os
import pandas as pd
from operator import itemgetter
from sklearn.utils import shuffle

#metrics calculation
from sklearn.metrics import precision_recall_fscore_support as score
```

In [7]:
```
#class_labels = [u'0',u'1',u'2',u'3',u'4',u'5',u'6',u'7',u'8',u'9']
#class_labels = [0,1,2,3,4,5,6,7,8,9]
```

In [ ]:
```
#class prediction
def main():
    model = '/home/bala/Documents/Hackathon/MachineHack'
    if model is not None:
        nlp = spacy.load(model)  # load trained text classifier
        print("Loading the trained model")
    else:
        print("Where is the trained model?")

    # load the dataset
    print("Loading data...")

    #option 1
    os.chdir('/home/bala/Documents/Hackathon/MachineHack/spaCyModel2')
    temp_1 = open(
        "//home/bala/Documents/Hackathon/MachineHack/spaCyModel2/WhoseLineIsItAnywayTRAIN_val.csv", 'r', encoding
    df = pd.read_csv(temp_1)
    df = df[['text', 'author']]
    df = df.dropna()
    df['text'].replace(r'\s+', ' ', regex=True, inplace=True)

    #option 2
    #df = pd.read_excel("QoR-20171.xlsx", sheet_name="Unique_Risks")


    print("Dataset has this many rows:", len(df))


    # test the saved model
    actualClass = []
    predictedClass = []
    texts = df['text']
    ActualClass = df['author']

    for aa in range(len(df)):
        test_text = texts.iloc[aa]
        doc = nlp(test_text)
        lis = doc.cats.items()
        actualClass.append(ActualClass.iloc[aa])
        predictedClass.append(max(lis, key=itemgetter(1))[0])
```

In [10]:
```
#loading previously created actuals and predictions SUB-DIMENSION
temp_1 = open("actualClass.csv", 'r', encoding='latin-1')
actualClass = pd.read_csv(temp_1)

temp_1 = open("predictedClass.csv", 'r', encoding='latin-1')
predictedClass = pd.read_csv(temp_1)

results = pd.concat((actualClass, predictedClass), axis=1)
results.columns = ("x","actualClass","y","predictedClass")
results = results[["actualClass","predictedClass"]]
```

In [15]:
```
#author available in the training dataset that are not in the predictions
print("number of author in training dataset: ",len(results.actualClass.unique())) #number of author in training d
print("number of author in predictions: ",len(results.predictedClass.unique())) #number of author in predictions
actualClass = list(results.actualClass)
predictedClass = list(results.predictedClass)
print("author missing in predictions:",list(set(actualClass).difference(predictedClass)))#author missing in predi
```

```
number of author in training dataset:  10
number of author in predictions:  1
author missing in predictions: [0, 1, 2, 3, 4, 5, 7, 8, 9]
```

In [16]:
```
precision, recall, fscore, support = score(actualClass, predictedClass, labels = class_labels)
```

```
/home/bala/anaconda3/lib/python3.5/site-packages/sklearn/metrics/classification.py:1135: UndefinedMetricWarning:
Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples.
  'precision', 'predicted', average, warn_for)
```

In [17]:
```
results1 = pd.concat((pd.Series(precision), pd.Series(recall), pd.Series(fscore), pd.Series(support)), axis=1, jo
results1.columns=('precision', 'recall', 'fscore', 'support')
#print(results1.shape) #shows 2 extra rows in which support = 0
#results1 #confirming 0 support rows
results2 = results1[results1.support != 0]
results2 = results2.reset_index(drop=True)
#print(results2.shape)
#results2 #rows with 0 support dropped
```

In [18]:
```
#adding class labels. The documentation for "sklearn.metrics precision_recall_fscore_support" says classes
#are arranged in sorted order
actualClassVals = pd.Series(sorted(actualClass)).unique()
#print(actualClassVals.shape)
results3 = pd.concat((pd.Series(actualClassVals), results2), axis=1, join = 'outer')
results3.columns=('author', 'precision', 'recall', 'fscore', 'support')
#print(results3.shape)
print(results3.support.sum())#confirming this adds up to the total number of rows in the dataset
(results3)
```

```
949
```

Out[18]:

	author	precision	recall	fscore	support
0	0	0.000000	0.0	0.00000	189
1	1	0.000000	0.0	0.00000	33
2	2	0.000000	0.0	0.00000	122
3	3	0.000000	0.0	0.00000	72
4	4	0.000000	0.0	0.00000	162
5	5	0.000000	0.0	0.00000	168
6	6	0.046365	1.0	0.08862	44
7	7	0.000000	0.0	0.00000	61
8	8	0.000000	0.0	0.00000	44
9	9	0.000000	0.0	0.00000	54

In [ ]: