File : ScikitCompare.py

```
import os.path as op
import numpy as np

from sklearn.naive_bayes import MultinomialNB
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.pipeline import Pipeline
from sklearn.model_selection import cross_val_score

##############################################################################
# Load data
print("Loading dataset")

from glob import glob
# filenames_neg = sorted(glob(op.join('..', 'data', 'imdb1', 'neg', '*.txt')))
# filenames_pos = sorted(glob(op.join('..', 'data', 'imdb1', 'pos', '*.txt')))
filenames_neg = sorted(glob(op.join('imdb1', 'neg', '*.txt')))
filenames_pos = sorted(glob(op.join('imdb1', 'pos', '*.txt')))
#filenames_neg = ["ciao.txt", "help.txt", "ciao1.txt"]
#filenames_pos = ["ciao2.txt", "ciao3.txt", "ciao4.txt"]

texts_neg = [open(f).read() for f in filenames_neg]
texts_pos = [open(f).read() for f in filenames_pos]
stopwords = open("english.stop").read()
texts = texts_neg + texts_pos
y = np.ones(len(texts), dtype=np.int)
y[:len(texts_neg)] = 0.

print("%d documents" % len(texts))

##############################################################################

# Create the set of stopwords
stopwords_set = set()
for word in stopwords.split(" "):
    stopwords_set.add(word)

# Pipeline = CountVectorizer + NaivaBayes from Scikit
pipeline = Pipeline([
    ('vect', CountVectorizer()),
    # ('tfidf', TfidfTransformer()),
    ('nb', MultinomialNB()),
])
# Pipeline - vect : parameters - fit
pipeline.set_params(vect__stop_words=list(stopwords_set)).fit(texts[::2], y[::2])
scores = cross_val_score(pipeline, texts[1::2], y[1::2], cv=5)
print("Accuracy: %0.2f (+/- %0.2f)" % (scores.mean(), scores.std() * 2))
```