sa_binary_training_log_reg_m

May 29, 2016

0.1 Load already prepared data

```
In [1]: TEST_SIZE=0.2
```

```
In [2]: import pandas as pd
```

```
In [3]: X = pd.read_csv('../valt_sa_data/x_m.csv')
        y = pd.read_csv('../valt_sa_data/y_m.csv', header=None)[0]
```

0.2 Split data into training and test sets

Let's perform a train/test split with 80% of the data in the training set and 20% of the data in the test set. We use `random_state=0` so that every execution yields the same result.

```
In [4]: from sklearn.cross_validation import train_test_split

        X_train, X_test, y_train, y_test = train_test_split(X.as_matrix(),
                                                            y.as_matrix(),
                                                            test_size=TEST_SIZE,
                                                            random_state=0)
```

1 Train a sentiment classifier with logistic regression

We will now use logistic regression to create a sentiment classifier on the training data.
Note: This line may take a few minutes.

```
In [5]: from sklearn import linear_model

        #logreg = linear_model.LogisticRegression(C=1e5)
        logreg = linear_model.LogisticRegression(C=1e5, solver='lbfgs', multi_class

        model = logreg.fit(X_train, y_train)
```

2 Evaluate the trained model

We will now use the cross-validation set to evaluate our model.

```
In [6]: from sklearn.metrics import confusion_matrix
        cm = confusion_matrix(y_test, model.predict(X_test))

        print 'Confusion matrix:'
        print cm

        from sklearn.metrics import classification_report

        print 'Classification report:'
        print classification_report(y_test, model.predict(X_test))
```

```
Confusion matrix:
[[   510      9      4      4   1288]
 [   219     11      9     13   1017]
 [   144      7     14     10   1651]
 [    63      8     10     22   3509]
 [   141      2      5     18  11312]]
Classification report:
             precision    recall  f1-score   support

          1       0.47      0.28      0.35      1815
          2       0.30      0.01      0.02      1269
          3       0.33      0.01      0.01      1826
          4       0.33      0.01      0.01      3612
          5       0.60      0.99      0.75     11478

avg / total       0.50      0.59      0.47     20000
```