

Finance Based Sentiment Analysis

Predict the sentiment in the financial news. Classification outputs a class from the set of 5 classes. Classes: ('very positive', 'positive', 'neutral', 'negative', 'very negative') and 2 swing class (positive swing or negative swing)

Most accurate with Financial Headlines as articles contains past or future reference of our subject hence confusing the model about present situation.

Dependencies

- **from** sklearn.feature_extraction.text **import** TfidfVectorizer
- **from** sklearn.pipeline **import** Pipeline
- **from** sklearn.linear_model **import** SGDClassifier

Features

- positive and negative words of financial system
- Used SGDclassifier to classify our input into +ve or -ve swing
- Used probabilities to classify further into very negative, negative or neutral similarly done for positive
- E.g. swing is negative with probability of 0.52 then it is neutral similarly
 - 0.45-0.55 (-ve swing) Neutral Class
 - 0.55-0.75 (-ve swing) Negative Class
 - 0.75- 1.0 (-ve swing) Very Negative Class

Challenge

Submission of technical assignment for the Internship Position at Credwatch.

Datasets

Used research paper of Bill McDonalds and Tim Loughran (from University of Notre Dame) to get positive and negative words of financial system

Results

give your input :

input:

Results with different INPUTS

```
input:
banks are staring at a bleak fourth quarter impacted by bad loans, weaker margins and slower loan growth, the profitability
in the sector may see a rise over last year due to even lower bad loan numbers.
negative
swing : ['negative']
```

```
input:
The commercial vehicle (CV) finance companies or microfinance companies will have higher credit cost because of low collecti
on efficiency. NBFC space or the private banks will probably standout compared to the others
very positive
swing : ['positive']
```

```
input:
today is friday
neutral
swing : ['positive']
```

```
input:
i am sued
negative
swing : ['negative']
```

Code

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.pipeline import Pipeline
from sklearn.linear_model import SGDClassifier

pos = open('/media/rushil83/5c8871fb-f3a3-4e54-bd63-de6fc6f68f0e/Downloads/untitled1/finance/pos.txt','r').read().replace('\n','')
neg = open('/media/rushil83/5c8871fb-f3a3-4e54-bd63-de6fc6f68f0e/Downloads/untitled1/finance/neg.txt','r').read().replace('\n','')

words = [pos,neg]
classes = ['positive','negative']

text_clf = Pipeline([('vect', TfidfVectorizer(stop_words='english')),('clf', SGDClassifier(loss='modified_huber', penalty='l2',
alpha=1e-3, n_iter=8, random_state=42))])

text_clf = text_clf.fit(words,classes)
x_test=[]
print('input: ')
temp = str(input())
x_test.append(temp)
output=text_clf.predict(x_test)

if text_clf.predict_proba(x_test)[0][0]>=0.75:
    print('very negative')
elif 0.75>text_clf.predict_proba(x_test)[0][0]>=0.55:
    print('negative')
elif 0.55>text_clf.predict_proba(x_test)[0][0]>=0.45:
    print('neutral')
elif 0.45>text_clf.predict_proba(x_test)[0][0]>=0.25:
    print('positive')
elif text_clf.predict_proba(x_test)[0][0]<0.25:
    print('very positive')

text_clf2 = Pipeline([('vect', TfidfVectorizer(stop_words='english')),('clf', SGDClassifier(loss='hinge', penalty='l2',
alpha=1e-3, n_iter=8, random_state=42))])

text_clf2 = text_clf2.fit(words,classes)
output=text_clf2.predict(x_test)
print('swing : ',output)
```