Sentiment classify program

# What can it do now?

It can extract features by two methods, filter useless features and do training and testing with the LIBSVM tool.

# More detail about its functions?

1. Use LIBSVM to classify the comment, every comment was built to a vector.

2. Use input configuration file to define parameters

3. Two method, please watch the following details

4. Delete useless features (using comments without emoticons: if all the comments do not contain the feature, delete it)

# More details about the project?

1. Project process flow

Read configuration file 🡪

generate new feature(two method) 🡪

add the new features to the feature file🡪

delete useless features(delete the features which never appeared)🡪

write all the formatted data by feature analysis🡪

choosing training data and testing data with given percentage🡪

do training and testing

2. Data

The data will be divided into several parts:



Note:

The comments for training should be half positive and half negative. “positiveNum” and “negativeNum” is used to balance the number. We could only choose (2 \* Min(PositiveNum, negativeNum)) pieces of data.

3.code details

<1>Read input configuration file

(1)Number

PercentOfTraingData: percentage of training data in commentsForTrainingAndTesting

DataNum: the number of all data choosing from database

FeatureRank:the number of top words to choose in word-frequency list

DataPercentForFeatureBuildingAndFilter: percentage of comments for feature building and filter (method1) in allCommentsWithEmoticon

percentForExtract=commentsForBuildNewFeatureMethod\_1/allCommentsWithEmoticon

percentForExtractMethod2=commentsForFeatureMethod\_2/allCommentsWithoutEmoticon

(2)file paths

AllDataFileName: training data+ testing data

TrainDataFileName=traing data

TestDataFileName=testing data

NewFeatureFileName= new features extracted by method 1(haven’t be filtered)

FilterFeatureFileName=filtered features by method 1

basicFeatureFilePath = features（7000 words）

FeatureFilePath= features(basic+new)

emotionWordFilePath=emotion words for feature extracting method 2

stopWordFilePath=stop words for feature extracting method 2

feature2FileName= extracted features for method 2

<2>extract feature method

(1) steps of method 1

A. get the word-frequency list of positive comments and negative comments

B. choose top X words from both lists

C. delete the words appear in both lists

D. filter the words by X2 examine.

(2) steps of method 2

A. get the data with emotion words

B. get the word-frequency list

C. delete the stop words

D. choosing top Y words in words list to be the new feature

<3>training and testing

(1)the format of training/testing data

For one comment, the traning example:

|  |
| --- |
| +1 128:1 1474:1 4362:1 5281:1 5756:1 6262:1 |

**“+1/-1”** : positive or negative

**“128:1”** : means the comment contain one 128th feature. “128” means the order number, “1” means the count of it in the comment.

**Notes:**

* If the count is 0, we could ignore this feature.
* If the comment contains no features, ignore it.
* The testing data is similarity except the “+1/-1”,it could be ignore in testing data. However, it could help us to auto-generate accuracy by libSVM.

(2) training and testing:

We use libSVM to training and testing

The command:

|  |
| --- |
| **svm-train** trainingFileName trainingModel  **svm-predict** testFileName trainingModel test.result |

svm-train: using traing data in training file to generate training model

svm-predict: using training model and testing data in testing file to generate test result

**Note:**

Using command line to test could get the accuracy.

<4> introduction of the main classes:

(1). Program.cs: the main class of the whole project, it contains the main function

(2). ReadConfigurationFile.cs: read config file and get the parameters

(3). GetAllData.cs:

getAllDataFromDatabase(): get all data from database

writeDataVector(): write all formatted data for training and testing

(4). BuildTrainingData.cs: choosing training data from all formatted data

(5).GetDataForBuildNewFeatureAndFilter\_method1:

Get commentsForBuildNewFeatureMethod\_1, commentsForFeatureCorrespondenceFilter

(6).GetDataForBuidNewFeatureMethod2AndFeatureFilter.cs:

Get commentsForFeatureMethod\_2, commentsForObscureFilter