NaiveBayesClassifier

Es wurde der Bernoulli Naive Bayes Klassifikator mit Laplace-Glättung mit Java implementiert.

Benutzung des Programms

Trainings und Testdateien

Um dem Programm Trainingsdaten hinzuzufügen, müssen diese entsprechend ihrer Klasse in das Verzeichnis `trainingData/neg` oder `trainingData/pos` abgelegt werden. Die Trainingsdaten müssen `.txt` Dateien sein und jedes Dokument der Trainingsdaten muss in einer separaten Datei sein, damit der Klassifikator die Anzahl der Dokumente korrekt zählen kann. Die zu klassifizierende Datei muss in das Verzeichnis `trainingData/unclassified`. Diese muss ebenso eine `.txt` Datei sein.

Kompilieren

Um das Programm zu kompilieren, gibt es zwei make Dateien. Eine `make.sh` für Linux und eine `make.bat` für Windows. Die `make.sh` muss möglicherweise Ausführungsrechte bekommen.

Aufruf:

```
.\make.bat
```

```
./make.sh
```

Programm starten

Um das Programm zu starten muss eine zu klassifizierende Datei im Verzeichnis `trainingData/unclassified` sein. Für den Aufruf des Programms stehen wieder eine `run.bat` und eine `run.sh` zur Verfügung. Die `run.sh` muss möglicherweise Ausführungsrechte bekommen. Mit dem Aufruf muss der Dateiname der zu klassifizierenden Datei übergeben werden.

Aufruf:

Windows:

```
.\run.bat <Filename>
```

Bsp.:

```
.\run.bat document.txt
```

Linux:

```
./run.sh <Filename>
```

Bsp.:

```
./run.sh document.txt
```

Als Ergebnis werden die Wahrscheinlichkeiten für die Klasse `pos` und `neg` ausgegeben und die höhere Wahrscheinlichkeit bestimmt.

Beispielausgabe:

```
probability neg: 0.0576
probability pos: 0.0024000000000000007
probability for class neg is higher
```

Class NaiveBayesClassifier

java.lang.Object
    NaiveBayesClassifier

```
public class NaiveBayesClassifier
extends Object
```

Constructor Summary

Constructors

Constructor	Description
`NaiveBayesClassifier ()`	Constructs a Naive Bayes Classifier object.

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
`double`	`calculateClassProbability (ClassData classData)`	Calculates the probability of a class based on the training data.
`double`	`calculateTotalProbability (ClassData classData, String document)`	Calculates the total probability of a document belonging to a specific class.
`void`	`classifieDocument (String file)`	Classifies a document based on the given training data.
`String`	`deleteUnclassifiedWords (String document)`	First replaces every non-letter with a whitespace.
`ClassData`	`train (String directory)`	Trains a ClassData object using documents from the specified directory.

Methods inherited from class java.lang.Object

`clone , equals , finalize , getClass , hashCode , notify , notifyAll , toString , wait , wait , wait`

Constructor Details

NaiveBayesClassifier

```
public NaiveBayesClassifier()
```

Constructs a Naive Bayes Classifier object. Initializes the classifier by training it with the "neg" and "pos" directories containing the negative and positive training data, respectively.

Method Details

classifieDocument

```
public void classifieDocument(String  file)
```

Classifies a document based on the given training data. This method reads the file from the "trainingData/unclassified" directory, cleans it from unclassified words, and calculates the probabilities for two classes (Positive and Negative). Based on these probabilities, it determines which class has the higher probability.

Parameters:

`file` - The filename of the file in the "trainingData/unclassified" directory.

deleteUnclassifiedWords

```
public String  deleteUnclassifiedWords(String  document)
```

First replaces every non-letter with a whitespace. Deletes all words from the document that are not classified by the test data. The document is filtered so that only words present in the training data (both negative and positive class word lists) are retained.

Parameters:

`document` - The document to be processed, as a String.

Returns:

A String containing only the classified words from the document.

train

```
public ClassData  train(String  directory)
```

Trains a ClassData object using documents from the specified directory. This method reads all text files from the given directory, processes their content, and adds them to the ClassData object for training purposes.

Parameters:

`directory` - The name of the directory within "trainingData" containing the training documents.

Returns:

A ClassData object containing the training data from the specified directory.

calculateTotalProbability

```
public double calculateTotalProbability(ClassData  classData,
                                        String  document)
```

Calculates the total probability of a document belonging to a specific class.

Parameters:

`classData` - The ClassData object representing the class for which the probability is calculated.

`document` - The document for which the probability is calculated.

Returns:

The total probability of the document belonging to the specified class.

calculateClassProbability

```
public double calculateClassProbability(ClassData  classData)
```

Calculates the probability of a class based on the training data.

Parameters:

`classData` - The ClassData object representing the class for which the probability is calculated.

Returns:

The probability of the class.

Class ClassData

java.lang.Object
    ClassData

```
public class ClassData
extends Object
```

Represents the training data for a specific class in the Naive Bayes Classifier.

Constructor Summary

Constructors

Constructor	Description
`ClassData(String directory)`	Constructs a ClassData object with the specified directory.

Method Summary

All Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
`void`	`addDocuments(String document)`	Adds a document to the ClassData object for training purposes.
`double`	`calculateProbability(String document)`	Calculates the probability of a document, based of the words in the documents and the training data.
`double`	`calculateWordProbability(String word)`	Calculates the probability of a word based on the training data using Laplace smoothing.
`String`	`getDirectory()`	Gets the directory of this class.
`int`	`getTrainingDocumentCounter()`	Gets the number of training documents for this class.
`Map <String ,Integer >`	`getWordCount()`	Gets the word count map for this class.

Methods inherited from class java.lang.Object

`clone , equals , finalize , getClass , hashCode , notify , notifyAll , toString , wait , wait , wait`

Constructor Details

ClassData

```
public ClassData(String  directory)
```

Constructs a ClassData object with the specified directory.

Parameters:

`directory` - The directory representing the class.

Method Details

addDocuments

```
public void addDocuments(String  document)
```

Adds a document to the ClassData object for training purposes. Replaces every non-letter with a whitespace.

Parameters:

`document` - The document to be added.

calculateProbability

```
public double calculateProbability(String  document)
```

Calculates the probability of a document, based of the words in the documents and the training data.

Parameters:

`document` - The document for which the probability is calculated.

Returns:

The probability of the document belonging to this class.

calculateWordProbability

```
public double calculateWordProbability(String  word)
```

Calculates the probability of a word based on the training data using Laplace smoothing. Laplace smoothing is applied to avoid zero probabilities for unseen words.

Parameters:

`word` - The word for which the probability is calculated.

Returns:

The probability of the word based on the training data.

getDirectory

```
public String  getDirectory()
```

Gets the directory of this class.

Returns:

The directory representing the class.

getTrainingDocumentCounter

```
public int getTrainingDocumentCounter()
```

Gets the number of training documents for this class.

Returns:

The number of training documents.

getWordCount

```
public Map <String ,Integer > getWordCount()
```

Gets the word count map for this class.

Returns:

The word count map.