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.