

Train and Test Supervised Text Classifier using fasttext

FastText Tutorial : We shall learn how to train and test supervised text classifier using fastText and check Precision and Recall values for the generated model.

Train and test Supervised Text Classifier using fasttext

Text Classification is one of the important NLP (Natural Language Processing) task with wide range of application in solving problems like Document Classification, Sentiment Analysis, Email SPAM Classification, Tweet Classification etc.

FastText provides “supervised” module to build a model for Text Classification using Supervised learning.

To work with fastText, it has to be built from source. To build fastText, follow the fastText Tutorial – [How to build FastText library from github source](#). Once fastText is built, run the fasttext commands mentioned in the following tutorial from the location of fasttext executable.

Prepare Training Data

Prepare a text file such that each line is an example. During the start of the line mention the labels. To mention a label, precede the label name with “__label__” (underscore underscore label underscore underscore).

Example of an entry is shown below.

```
__label__wish Good Morning
```

where ‘**wish**’ is a label ‘**Good Morning**’ is the data for the example.

Multiple labels could be mentioned for an entry as below.

```
__label__wish __label__question Good Morning. Did you have break-fast ?
```

Prepare a text document containing multiple entries of such to train a text classifier with supervised training using FastText.

```
_label_greet Good Morning  
_label_greet Good Evening  
_label_greet Good Day  
_label_greet Good Afternoon  
_label_greet All the best  
_label_greet Good luck  
_label_greet Happy Birthday  
_label_greet Happy Journey  
_label_wish _label_question Good Morning. Did you have break-fast ?  
_label_question When did you come ?  
_label_question When did you reach office ?  
_label_question Where did you go in the morning ?  
_label_question What did you bring for lunch ?
```

Run the following command to train supervised classifier with input as sampleData.train and the generated output model to supervised_classifier_model.

```
$ ./fasttext supervised -input trainingData.txt -output supervised_classifier_model
```

```
Read 0M words  
Number of words: 32  
Number of labels: 3  
Progress: 100.0% words/sec/thread: 204861 lr: 0.000000 loss: 0.917794 eta: 0h0m
```

- Number of words represent number of unique words in the training data.
- Number of labels represent number of unique labels in the training data.
- words/sec/thread is the number of words that could be processed per second per thread.
- loss is 0.9
- supervised_classifier_model.bin would be the model generated as a result of training the supervised classifier.

Test the model

We shall test the generated model using test data. The test data has the format same as that of training data.

```
_label_greet Good Night  
_label_greet Good luck  
_label_question What is your name ?
```

Run the following command in the terminal.

```
$ ./fasttext test supervised_classifier_model.bin testData.txt
```

```
$ ./fasttext test supervised_classifier_model.bin testData.txt  
N 3  
P@1 0.667  
R@1 0.667  
M 1.000000
```

Precision is at 0.667 (66.7%) and Recall is at 0.667 (66.7%).

Conclusion

In this [Fasttext Tutorial](#) – Train and test supervised text classifier using fasttext, we have learnt to train a supervised Text Classifier using training data containing examples, and generate a model. The model is then tested to evaluate its Precision and Recall.

FastText Tutorial

- ◆ [FastText Tutorial Home](#)
- ◆ [FastText - BUILD](#)
- ⇒ **[FastText - Train and Test Supervised Text Classifier](#)**
- ◆ [Learn Word Representations in FastText](#)

FastText Python

- ◆ [FastText Python - Learn Word Representations](#)