

CENG 463

Assignment 1

Cem Gündoğdu
cem.gundogdu@metu.edu.tr

December 16, 2021

Contents

1	Naive Bayes Classifier	2
1.1	Introduction	2
1.2	Metrics for different preprocessing techniques	2
1.2.1	Simplest version	2
1.3	Evaluation	2
1.3.1	Calculation of metrics	2

1 Naive Bayes Classifier

1.1 Introduction

I used the `NaiveBayesClassifier` class from the NLTK library. I have tried various preprocessing techniques, and logged the accuracy along with the confusion matrix and metrics such as precision.

1.2 Metrics for different preprocessing techniques

In this section, I list the accuracies for different preprocessing techniques. I haven't tried all combinations of them, since the number grows exponentially. However, I added each technique incrementally, with the hope that the change caused by the added technique will be representative of its usefulness.

These trials were made on the *dev* set, since I did them during the development phase. The evaluation in [subsection 1.3](#) will be done on the *test* set.

Complete logs from each try is available in the data directory. These include the confusion matrix, accuracy, and for each category the precision, recall and F_1 measure. I include only the last log here, to keep this report short.

Details of the calculation of precision, recall and F_1 measure are given in [subsection 1.3](#).

A table of the accuracies for each technique is in ??.

1.2.1 Simplest version

The words in title are counted twice — in this version and in all the following ones. Other than this, there is no processing in this version. The text is given to `nltk.word_tokenize()` and the resulting tokens are used to train the classifier.

Accuracy in this version is 64.2%. Confusion matrix and other metrics for each category are listed in [Figure 1](#).

1.3 Evaluation

1.3.1 Calculation of metrics

Recall is the ratio of true positives for a class to the number of input documents of that type. To find recall, we divide each diagonal entry by the sum of corresponding row.

Precision is the ratio of true positives for a class to the number of documents that are identified to be in that class. To calculate it, we divide diagonal entries by the sum in that column.

Figure 1: Metrics for the simplest version

Loaded classifier from cache.

Accuracy: 0.6420150053590568

		p		s	c	i	e	n	c	e	-	f	i	s
	m	l	r											
h	y	o	l	r	s	c	i	c	p					
o	s	s	i	m	a	e	t	n	i	r				
r	t	o	g	a	n	n	c	o	t					
r	e	p	i	n	c	c								
o	r	h	o											
r	y	y	n	e	e	n	s							

horror		<44>	23	.	3	27	6	8	7	
mystery		8	<88>	.	.	18	.	1	5	
philosophy		.	2	<44>	20	.	47	1	.	
religion		.	3	13	<76>	4	12	3	4	
romance		1	3	.	3	<82>	2	4	19	
science		.	.	.	3	.	<107>	4	1	
science-fiction		11	7	.	3	23	15	<52>	9	
sports		8	3	.	<106>	

```
(row = reference; col = test)
```

	Precision	Recall	F1-Measure
philosophy	0.7719	0.3860	0.5146
sports	0.7020	0.9060	0.7910
mystery	0.6984	0.7333	0.7154
religion	0.7037	0.6609	0.6816
science	0.5573	0.9304	0.6971
romance	0.5062	0.7193	0.5942
horror	0.6875	0.3729	0.4835
science-fiction	0.7123	0.4333	0.5389