Ranking Websites - Clustering websites by categories

Supervisor: Diego Esteves

Xiaotian Zhou
2743095

Introduction

- Given: Dataframe stores URLs with categories
- Scrape contents from URLs and create datasets
- Transform the data: TF-IDF
- Apply the algorithms using scikit-learn:

  NB, SVM, DT, etc.
- Plot the graph and confusion matrix
- Helpful tool: BeautifulSoup

Benchmark

Data Set Characteristics:	Multivariate	**Number of Instances:**	422937	**Area:**	N/A
Attribute Characteristics:	N/A	**Number of Attributes:**	5	**Date Donated**	2016-02-28
Associated Tasks:	Classification, Clustering	**Missing Values?**	N/A	**Number of Web Hits:**	46262

Data Set Information:

News are grouped into clusters that represent pages discussing the same news story.
The dataset includes also references to web pages that, at the access time, pointed (has a link to) one of the news page in the collection.

422937 news pages and divided up into:

152746 news of business category
108465 news of science and technology category
115920 news of business category
45615 news of health category

2076 clusters of similar news for entertainment category
1789 clusters of similar news for science and technology category
2019 clusters of similar news for business category
1347 clusters of similar news for health category

https://archive.ics.uci.edu/ml/datasets/News+Aggregator

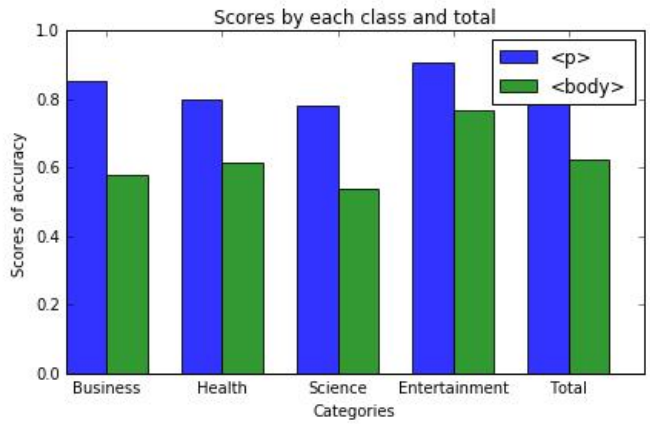
Evaluation:

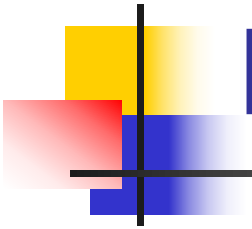
- 2 ways to scrape web contents:
1. get all contents under paragraph tag <p>
2. simply get body tag <tag>

- Test the accuracy for different classifiers

```
In [42]: #Draw the bar chart and compare the results with data scraped by 2 ways (<p> and <body>)
         print("          Comparison by Naive Bayes classifier: ")
         plot_bar_chart(score_NB, score_NB_body)
```

```
          Comparison by Naive Bayes classifier:
```





More details in the code...