Web Crawling and Text Classification

SHUBHAM PATIL
ECKOVATION
PYTHON PROGRAMMING

Web Crawling

Introduction

Web crawlers are mainly used to create a copy of all the visited pages for later processing by a search engine, that will index the downloaded pages to provide fast searches.

Crawlers can also be used for automating maintenance tasks on a Web site, such as checking links or validating HTML code.

In this project, we aim to perform Web Crawling of website named Figshare.
Figshare is an online open access repository where researchers can preserve and share their research outputs, including figures, datasets, images, and videos.

Requirements

Software

- Python 3.x
- Anaconda Distribution
- NLTK Toolkit
- Beautiful Soup

Hardware

- Core i5/i7 processor
- At least 8 GB RAM
- At least 60 GB of Usable Hard Disk Space

Beautiful Soup

Beautiful Soup is a Python library for pulling data out of HTML and XML files. You can use the pip package manager to install BeautifulSoup.

Steps for using Beautifulsoup are as follows:

1. Request library is used to fetch content from a given link.
2. Initialize the argument parser and parse the filename argument.
3. Find the length of links and print this information.
4. Create a function to accept an image URL and download it.
5. Using the get method in requests library, fetch the URL.
6. html.parser defines that the given content has to be parsed as HTML.

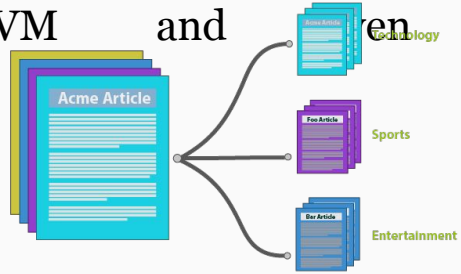
Text Classification

What is Text Classification?

Text Classification is an automated process of classification of text into predefined categories.

We can classify Emails into spam or non-spam, news articles into different categories like Politics, Stock Market, Sports, etc.

This can be done with the help of Natural Language Processing and different Classification Algorithms like Naive Bayes, SVM and even Neural Networks in Python.



Support Vector Classifier

The objective of a Linear **SVC** (Support Vector Classifier) is to fit to the data you provide, returning a "best fit" hyperplane that divides, or categorizes, your data.

LinearSVC are classes capable of performing multi-class classification on a dataset.

- SVM Classifiers offer good accuracy and perform faster prediction compared to Naïve Bayes algorithm.
- They also use less memory because they use a subset of training points in the decision phase.
- SVM works well with a clear margin of separation and with high dimensional space.

In [5]:
```
text_clf = Pipeline([('vect', CountVectorizer()),
                     ('tfidf', TfidfTransformer()),
                     ('clf', LinearSVC()),
                     ])

text_clf.fit(X_train, y_train)


predicted = text_clf.predict(X_test)

print(metrics.classification_report(y_test, predicted))
```

	precision	recall	f1-score	support
0	0.82	0.80	0.81	319
1	0.76	0.80	0.78	389
2	0.77	0.73	0.75	394
3	0.71	0.76	0.74	392
4	0.84	0.86	0.85	385
5	0.87	0.76	0.81	395
6	0.83	0.91	0.87	390
7	0.92	0.91	0.91	396
8	0.95	0.95	0.95	398
9	0.92	0.95	0.93	397
10	0.96	0.98	0.97	399
11	0.93	0.94	0.93	396
12	0.81	0.79	0.80	393
13	0.90	0.87	0.88	396
14	0.90	0.93	0.92	394
15	0.84	0.93	0.88	398
16	0.75	0.92	0.82	364
17	0.97	0.89	0.93	376
18	0.82	0.62	0.71	310
19	0.75	0.61	0.68	251
accuracy			0.85	7532
macro avg	0.85	0.85	0.85	7532

The Accuracy, Precision, Recall, and Evaluation time is calculated and displayed.

Accuracy of 0.85 is obtained by Support Vector Classifier being the highest.

This project focuses on a analysis model consisting of three core steps, namely data preparation, review analysis and classification, and describes representative technique involved in those steps.

Thank You!

SHUBHAM PATIL
ECKOVATION
PYTHON PROGRAMMING