Project on
WEB CRAWLER AND TEXT CLASSIFICATION

Submitted by
SHUBHAM BALASAHEB PATIL

Internship at
ECKOVATION CAREERS

Course Name
PYTHON PROGRAMMING

Index

Abstract

Web Crawling also known as Opinion Mining refers to the use of natural language processing, text analysis to systematically identify, extract, quantify, and study affective states and subjective information. Web Crawling is widely applied to reviews and survey responses, online and social media, and healthcare materials for applications that range from marketing to customer service to clinical medicine.

In this project, we aim to perform Web Crawling of product based reviews. Data used in this project are online product reviews collected from “amazon.com”. We expect to do review-level categorization of review data with promising outcomes.

Automated text classification has been considered as a vital method to manage and process a vast amount of documents in digital forms that are widespread and continuously increasing. In general, text classification plays an important role in information extraction and summarization, text retrieval, and question-answering. This paper illustrates the text classification process using machine learning techniques. The references cited cover the major theoretical issues and guide the researcher to interesting research directions.

Introduction

Sentiment analysis, which is also known as opinion mining, studies people's sentiments towards certain entities. From a user's perspective, people are able to post their own content through various social media, such as forums, micro-blogs, or online social networking sites. From a researcher's perspective, many social media sites release their application programming interfaces (APIs), prompting data collection and analysis by researchers and developers. However, those types of online data have several flaws that potentially hinder the process of sentiment analysis. The first flaw is that since people can freely post their own content, the quality of their opinions cannot be guaranteed. The second flaw is that ground truth of such online data is not always available. A ground truth is more like a tag of a certain opinion, indicating whether the opinion is positive, negative, or neutral.

"It is quite a boring movie…….. but the scenes were good enough. " The given line is a movie review that states that "it" (the movie) is quite boring but the scenes were good. Understanding such sentiments requires multiple tasks. Hence, SENTIMENTAL ANALYSIS is a kind of text classification based on Sentimental Orientation (SO) of opinion they contain. Sentiment analysis of product reviews has recently become very popular in text mining and computational linguistics research.

- Firstly, evaluative terms expressing opinions must be extracted from the review.
- Secondly, the SO, or the polarity, of the opinions must be determined.
- Thirdly, the opinion strength, or the intensity, of an opinion should also be determined.
- Finally, the review is classified with respect to sentiment classes, such as Positive and Negative, based on the SO of the opinions it contains.

Objective of the Project

- Scrapping product reviews on various websites featuring various products specifically amazon.com.
- Analyze and categorize review data.
- Analyze sentiment on dataset from document level (review level).
- Categorization or classification of opinion sentiment into- Positive Negative.

Figure 1: A typical sentiment analysis model.

System Design

Hardware Requirements:

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

Software Requirements:

- Python 3.x
- Anaconda Distribution
- NLTK Toolkit
- UNIX/LINUX Operating System

Data Information:

- The Amazon reviews dataset consists of reviews from amazon. The data span a period of 18 years, including ~35 million reviews up to March 2013. Reviews include product and user information, ratings, and a plaintext review. For more information, please refer to the following paper: J. McAuley and J. Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. RecSys, 2013.

- The Amazon reviews full score dataset is constructed by Xiang Zhang (xiang.zhang@nyu.edu) from the above dataset. It is used as a text classification benchmark in the following paper: Xiang Zhang, Junbo Zhao, Yann LeCun. Character-level Convolutional Networks for Text Classification. Advances in Neural Information Processing Systems 28 (NIPS 2015).

- The Amazon reviews full score dataset is constructed by randomly taking 200,000 samples for each review score from 1 to 5. In total there are 1,000,000 samples.

Methodology for Implementation

DATA COLLECTION:

Data which means product reviews collected from amazon.com from May 1996 to July 2014. Each review includes the following information:
1) reviewer ID
2) product ID
3) rating
4) time of the review
5) helpfulness
6) review text.
Every rating is based on a 5-star scale, resulting all the ratings to be ranged from 1-star to 5-star with no existence of a half-star or a quarter-star.

SENTIMENT SENTENCE EXTRACTION & POS TAGGING:

Tokenization of reviews after removal of STOP words which mean nothing related to sentiment is the basic requirement for POS tagging. After proper removal of STOP words like “am, is, are, the, but” and so on the remaining sentences are converted into tokens. These tokens take part in POS tagging. In natural language processing, part-of-speech (POS) taggers have been developed to classify words based on their parts of speech. For sentiment analysis, a POS tagger is very useful because of the following two reasons:

1) Words like nouns and pronouns usually do not contain any sentiment. It is able to filter out such words with the help of a POS tagger;

2) A POS tagger can also be used to distinguish words that can be used in different parts of speech.

NEGATIVE PHRASE IDENTIFICATION:

Words such as adjectives and verbs are able to convey opposite sentiment with the help of negative prefixes. For instance, consider the following sentence that was found in an electronic device's review: "The built in speaker also has its uses but so far nothing revolutionary." The word, "revolutionary" is a positive word according to the list in. However, the phrase "nothing revolutionary" gives more or less negative feelings. Therefore, it is crucial to identify such phrases. In this work, there are two types of phrases that have been identified, namely negation-of-adjective (NOA) and negation-of-verb (NOV).

SENTIMENT CLASSIFICATION ALGORITHMS

Support Vector Classifier

Support vector classifier (SVC) is a method for the classification of both linear and nonlinear data. If the data is linearly separable, the SVC searches for the linear optimal separating hyperplane (the linear kernel), which is a decision boundary that separates data of one class from another. Mathematically, a separating hyperplane can be written as: WX+b=0, where W is a weight vector and W=w1,w2,...,w n. X is a training tuple. b is a scalar. In order to optimize the hyperplane, the problem essentially transforms to the minimization of $\|W\|$, which is eventually computed as:

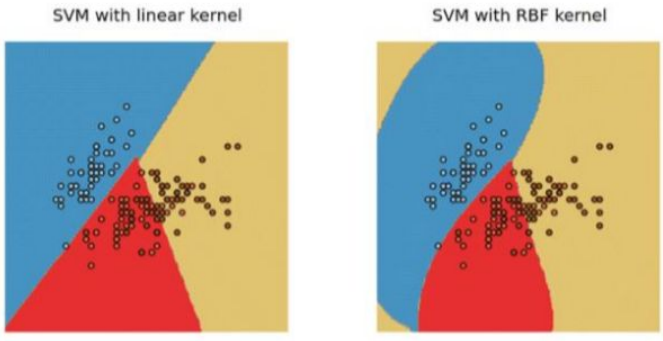
where αi are numeric parameters, and yi are labels based on support vectors, Xi .

That is: if yi =1 then

if $y_i = -1$ then

If the data is linearly inseparable, the SVC uses nonlinear mapping to transform the data into a higher dimension. It then solves the problem by finding a linear hyperplane. Functions to perform such transformations are called kernel functions. The kernel function selected for our experiment is the Gaussian Radial Basis Function (RBF):

where Xi are support vectors, X j are testing tuples, and γ is a free parameter that uses the default value from scikit-learn in our experiment. Figure shows a classification example of SVC based on the linear kernel and the RBF kernel below.



Implementation Details

The training of dataset consists of the following steps:

1. Unpacking of data:

The huge dataset of reviews obtained from amazon.com comes in a .json file format. A small python code has been implemented in order to read the dataset from those files and dump them into a pickle file for easier and fastaccess and object serialization.

```
In [ ]: import requests
        from bs4 import BeautifulSoup

        class Content:
            def __init__(self, url, title, body):
                self.url = url
                self.title = title
                self.body = body

        def getPage(url):
            req = requests.get(url)
            return BeautifulSoup(req.text, 'html.parser')
```

2. Preparing Data for Sentiment Analysis:

i) The pickle file is hence loaded in this step and the data besides the one used for sentiment analysis is removed. As shown in our sample dataset, there are a lot of columns in the data out of which only rating and text review is what we require. So, the column, “reviewSummary” is dropped from the data file.
ii) After that, the review ratings which are 3 out of 5 are removed as they signify neutral review, and all we are concerned of is positive and negative reviews.
iii) The entire task of preprocessing the review data is handled by this
iv) The time required to prepare the following data is hence displayed.

```
In [ ]: def scrapeFiles(url):
            bs = getPage(url)
            title = bs.find('h1').text
            body = bs.find({'class','body'}).text
            return Content(url, title, body)

        url = 'https://ndownloader.figshare.com/files/5975967'
        content = scrapeFiles(url)
        print('Title: {}'.format(content.title))
        print('URL: {}\n'.format(content.url))
        print(content.body)
```

3. Preprocessing Data:

This is a vital part of training the dataset. Here Words present in the file are accessed both as a solo word and also as a pair of words. Because, for example the word “bad” means negative but when someone writes “not bad” it refers to as positive. In such cases considering single word for training data will work otherwise. So words in pairs are checked to find the occurrence to modifiers before any adjective which if present which might provide a different meaning to the outlook.

```
In [25]: newsgroups_train = fetch_20newsgroups(subset='train')
         newsgroups_test = fetch_20newsgroups(subset='test')
         X_train = newsgroups_train.data
         X_test = newsgroups_test.data
         y_train = newsgroups_train.target
         y_test = newsgroups_test.target

         Downloading 20news dataset. This may take a few minutes.
         Downloading dataset from https://ndownloader.figshare.com/files/5975967 (14 MB)
```

4. Training Data/ Evaluation:

The main chunk of code that does the whole evaluation of sentimental analysis based on the preprocessed data is a part of this. The following are the steps followed:

i) The Accuracy, Precision, Recall, and Evaluation time is calculated and displayed.
ii) Naive Bayes, Logistic Regression, Linear SVC and Random forest classifiers are applied on the dataset for evaluation of sentiments.
iii) Prediction of test data is done and Confusion Matrix of prediction isdisplayed.
iv) Total positive and negative reviews are counted.
v) A review like sentence is taken as input on the console and if positive the console gives 1 as output and 0 for negative input.

```
In [26]: 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))
```

Conclusion

Sentiment analysis deals with the classification of texts based on the sentiments they contain. This article focuses on a typical sentiment analysis model consisting of three core steps, namely data preparation, review analysis and sentiment classification, and describes representative techniques involved in those steps.

Sentiment analysis is an emerging research area in text mining and computational linguistics, and has attracted considerable research attention in the past few years. Future research shall explore sophisticated methods for opinion and product feature extraction, as well as new classification models that can address the ordered labels property in rating inference. Applications that utilize results from sentiment analysis are also expected to emerge in the near future.

References

- S. ChandraKala1 and C. Sindhu2, “OPINION MINING AND SENTIMENT CLASSIFICATION: A SURVEY,”.Vol .3(1),Oct 2012,420-427

- G.Angulakshmi , Dr.R.ManickaChezian ,”An Analysis on Opinion Mining: Techniques and Tools”. Vol 3(7), 2014 www.iarcce.com. Callen Rain,”Sentiment Analysis in Amazon Reviews Using Probabilistic Machine Learning” Swarthmore College, Department of Computer Science.

- Padmani P .Tribhuvan,S.G. Bhirud,Amrapali P. Tribhuvan,” A Peer Review of Feature Based Opinion Mining and Summarization”(IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (1), 2014, 247-250 ,www.ijcsit.com.

- Carenini, G., Ng, R. and Zwart, E. Extracting Knowledge from Evaluative Text. Proceedings of the Third International Conference on Knowledge Capture (K-CAP’05), 2005. Dave, D., Lawrence, A., and Pennock, D. Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews. Proceedings of International World Wide Web Conference (WWW’03), 2003.

- Zhu, Jingbo, et al. "Aspect-based opinion polling from customer reviews." IEEE Transactions on Affective Computing, Volume 2.1,pp.37-49, 2011. Na, Jin-Cheon, Haiyang Sui, Christopher Khoo, Syin Chan, and Yunyun Zhou. "Effectiveness of simple linguistic processing in automatic sentiment classification of product reviews." Advances in Knowledge Organization Volume9, pp. 49-54, 2004.

- Nasukawa, Tetsuya, and Jeonghee Yi. "Sentiment analysis: Capturing favorability using natural language processing." In Proceedings of the 2nd international conference on Knowledge capture, ACM, pp. 70-77, 2003.

- Li, Shoushan, Zhongqing Wang, Sophia Yat Mei Lee, and Chu-Ren Huang. "Sentiment Classification with Polarity Shifting Detection." In Asian Language Processing (IALP), 2013 International Conference on, pp. 129-132. IEEE, 2013.