Machine Learning Engineer Nanodegree Capstone Proposal

Topic: Using Twitter Data for NLP and Sentiment Analysis

Siddhesh Khedekar

October 29, 2018

1. Introduction

Below I have explained the projects domain background, in essence, the field of research from where the project is derived.

In today's time, microblogging has become the most widely used communication tool among users of the web, especially through social media sites. Daily millions of messages and posts appear on popular social media networks and websites such as Facebook, Twitter, Reddit, Quorra that provide microblogging services. People on such sites write about their life, discuss current issues and share their opinions on a variety of topics. More and more Internet users are shifting towards microblogging platforms from blogs or mailing lists which are the most common traditional communication tools as they give more easy accessibility and free format of communication. Every day an increasing number of users post their reviews about the products they use along with their religious and political views. Social media and microblogging websites have become valuable sources for analysis of opinions and sentiments such data can be efficiently used for digital marketing and social studies.

2. Problem description

Here I have stated the problem I am investigating and for which I will be defining a solution.

The problem I will be working upon is detecting the sentiment of a text string being positive or negative. Through proper implementation of Natural Language Processing it is possible to learn a lot from simple data streams such as Twitter posts. A number of domains can benefit from Sentiment Analysis and its applications in today's world are ever increasing. Some of the domains Sentiment Analysis finds its uses are listed below.

A. E-Commerce Sites

E-commerce activities involve the most general use of sentiment analysis. Sites allow their users to submit their reviews about product qualities and shopping experiences. For every product, a summary is provided along with its different features and user-assigned rating and scores. New users can view recommendations and opinions regarding a product and its individual features. Visualization using graphical models of overall features and products help users choose. Amazon, Flipkart and other popular merchant websites provide reviews and ratings from users. Trivago and Tripadvisor are popular websites that provide reviews on travel destinations and hotels. they contain millions of opinions from across the world. Sentiment Analysis helps in converting the huge volume of opinions of users into promoting business.

B. Government Organizations

Governments can assess their strengths and weaknesses through Sentiment analysis of the opinions of the citizens. A Twitter poll of who is most likely to win an election is the most basic example of sentiment analysis at work. Be it tracking of peoples opinions on a newly proposed law or some newly introduced system or identifying improvements in campaigns and many other similar areas, we can see the potential for sentiment analysis.

C. BRM, VOM, VOC

The concern about managing a companies reputation in the market is described as Brand Reputation Management(BRM). BRM is focused more on the company rather than the customer. Analysis of Sentiment proves to be helpful in determining how a company's brand, service or product is being perceived by community online.

Determining what customers are feeling about services or products of competitors is termed as Voice of the Market(VOM). With sentiment analysis, it is possible to get customer opinion in real-time. This accumulated data can help companies can predict the chances of product failure to design new marketing strategies and improve product features.

The concern about what individual customers are saying about services and products is defined as Voice of the Customer(VOC). VOC is the analyzing of feedback and reviews of the users. Obtaining customer opinions helps non-functional requirements like cost and performance and some identify the functional requirements of the products.

3. Datasets used

I will be using the following datasets and inputs for solving the problem.

In order to train my twitter sentiment classifier, I needed a dataset which meets conditions below.

- big enough to train a model
- preferably tweets text data with annotated sentiment label
- with at least 2 sentiment classes: negative and positive

While googling to find a good data source, I learned about renowned NLP competition called Sentiment140 dataset with 1.6 million tweets on Kaggle. The dataset for training the model is from "Sentiment140", a dataset originated from Stanford University. More info on the dataset can be found on its official website. The dataset can be downloaded from the provided link. First look at the description of the dataset, the following information on each field can be found.

1. sentiment of the tweet (0 = negative, 4 = positive)
2. id of the tweet (3223)
3. date of the tweet (Sun Mar 12 13:38:45 UTC 2006)
4. query_string (qwe). NO_QUERY is set as value if there is no query
5. user that tweeted (dcrules)
6. text of the tweet (Marvel is cool)

Here I will first need to be dropping some of the columns that I don't need for the specific purpose of sentiment analysis.

4. Solution

The solution I would like to propose for the purpose of sentiment analysis is given below.

Firstly even before Data Exploration, there will be a lot of cleaning work to be done on the data set. I will have to clean up all the HTML elements, byte order marks, URL address, Twitter mentions, numbers, and special characters. I will convert all text to lower case, handle negation along with tokenizing and joining.

Then comes the Feature engineering part where I will need to experiment with the maximum number of features to see how the results differ. I will also have to experiment with different n-grams (unigram, unigram+bigram, bigram only, etc.), test both with and without stemming or lemmatization, test different vectorizing models (count vectorizer, tfidf vectorizer, word2vec/doc2vec), test with and without lexicon use before creating my model. I will have to try to understand different models namely Naive Bayes, Random Forest, Logistic Regression, etc along with the above features. Then I need to create my own model of Convolutional Neural Network with the same features. Also, I need to compare their performance to the benchmark as well as each other.

For the purpose of implementing the dataset is a very huge one to be training and for the purpose of being able to go through the training of the model on my system I will be forced to be using only a small portion of the original dataset. The risk I face here is relying on training data's vocabularies, if the training data is not big enough, it has a risk of performing not very well with the validation and test data. I will be overcoming this by splitting my data in some ratio, the majority of data will be used as the training set, and the minority equally divided for the validation and the test set. This way there will be more than enough values to refine the parameters and evaluate the model.

5. Model

The below-mentioned benchmark model and their results will be useful to compare the solution I have defined.

All the model performance will be compared to fit the same training and validation set. Baseline or benchmark provides a point of reference to compare when comparing various machine learning algorithms. Zero Rule (ZeroR) is one of the most popular baselines. It simply predicts the majority category (class). There is no predictability power in ZeroR but it is useful for determining a baseline performance as a benchmark for other classification methods.

Another benchmark I wanted to compare the validation results with is TextBlob. This python library helps in processing textual data. Apart from other useful tools such as POS tagging, n-gram, The package has built-in sentiment classification. It is an out-of-the-box sentiment analysis tool. I will keep in mind of the accuracy I get from TextBlob sentiment analysis along with the null accuracy to see how my model is performing.

how my model is performing.

6. Evaluations

The functional representations for how the solution can be measured will be the two evaluation metrics stated below.

I will be splitting the dataset into training, validation and testing set. The main evaluation metric for the model will be the validation set accuracy, the testing set accuracy along with their respective comparisons to the benchmarks accuracy.

7. Overview

These are the steps of project design I will be following for developing the solution and obtaining results.

- Step 0: Import Datasets and Data Preparation
- Step 1: Data Cleaning and Saving Cleaned Data as CSV
- Step 2: Explanatory Data Analysis and Visualisation of Zipf's Law
- Step 3: Data Split and Benchmark Selection
- Step 4: Feature Investigation and Extraction
- Step 5: Model Comparison and Creating a CNN
- Step 6: Training Model and Measuring Validation Accuracy
- Step 7: Testing Model and Getting the Final Result

8. Conclusion

The conclusion of the project and the results are the following.

I am successfully able to create a sentiment classifier using a labeled twitter dataset. The CNN Classifier I implement will be able to meet a certain accuracy level of approximately higher than the benchmarks I have used.

9. References

Following are the references that helped me make this project a success.

[1] - Steven Bird, Ewan Klein, and Edward Loper. Natural language processing with Python. "O'Reilly Media, Inc.", 2009.

[2] - A Survey on Sentiment Analysis Algorithms for Opinion Mining

[3] - Mullen Sentiment Course Slides

[4] - Stanford Sentiment Slides

[5] - Sentiment Analysis and Opinion Mining

[6] - Sentiment Analysis Methodology of Twitter Data

[7] - Pak, Alexander, and Patrick Paroubek. "Twitter as a Corpus for Sentiment Analysis and Opinion Mining." Vol. 10. 2010.

[8] - Build a Sentiment Analysis Tool for Twitter with this Simple Python Script - AYLIEN

[9] - Text Classification & Sentiment Analysis tutorial / blog - Data Science Central

[10] - Sentiment Analysis on News Articles using Python for traders

[11] - Sentiment140 dataset with 1.6 million tweets

[12] - Sentiment Analysis Wikipedia