CS 410 Course Project Progress Report

11.15.2021

—

Ratan Bajpai
MCS-DS Student
University of Illinois at Urbana-Champaign
Champaign, IL

1. Overview

This document provides a progress report for the course project for CS 410.

2. Team Captain / Member (Team: Sentient)

1. Ratan Bajpai - rbajpai2@illinois.edu

3. Challenges Faced

In the project proposal we proposed to do sentiment analysis on the “Movie Review” dataset (http://www.cs.cornell.edu/people/pabo/movie-review-data/) using two different approaches / tools. One approach is Amazon Comprehend and the other is python based NLTK toolkit. The idea was to see how well Amazon Comprehend performs based on it’s comparison with NLTK. However there were the following challenges faced and so we are planning to use the “Twitter Samples” dataset which is available in the NLTK Corpus.

1. Text Limit with Amazon Comprehend

When invoking Amazon Comprehend “DetectSentiment” API, it has a size limit of 5000 UTF-8 encoded bytes for the input text for which sentiment analysis has to be done. However the Movies Review Dataset contains several reviews that are well above this limit. Hence, we have decided to use the “Twitter Samples” dataset, which does not violate this limit.

2. Parameter Tuning

Although NLTK provides a low level API to try different ML models to use while doing sentiment analysis, Amazon Comprehend has a very high level API in which there are no models or parameters available for tuning. Amazon Comprehend uses a pre-trained model for doing sentiment analysis. It has the following request / response structure. Because of this limitation, all the model selection / parameter tuning would be done on the NLTK side to try to achieve good performance for this task. The Amazon Comprehend API will be used as it is available.

3. DetectSentiment API

This API takes the following request / response structure (source: Amazon Comprehend website):

- Request

```
{
   "LanguageCode": "string",
   "Text": "string"
}
```

Here the “LanguageCode” denotes what language the text string is in, i.e. “en”, “de”, “es” etc. A list of these codes along with description is available on Amazon Comprehend website. The “Text” is the actual text string for which the sentiment analysis has to be done.

- Response

```
{
   "Sentiment": "string",
   "SentimentScore": {
      "Mixed": number,
      "Negative": number,
      "Neutral": number,
      "Positive": number
   }
}
```

The response contains the “Sentiment” result, which could be “POSITIVE”, “NEGATIVE”, “NEUTRAL” or “MIXED”. Also it contains a score for each of those sentient values. Below is an example of such a response.

- Example Response

```
{
    "SentimentScore": {
        "Mixed": 0.0033542951568961143,
        "Positive": 0.9869875907897949,
        "Neutral": 0.008563132025301456,
        "Negative": 0.0010949420975521207
    },
    "Sentiment": "POSITIVE",
 }
}
```


4. Tasks Completed

1. Analysis of Movie Review and Twitter Samples datasets.
2. Understanding how Amazon Comprehend works, it's implementation details and it's limitations.
3. Understanding how NLTK works and it's implementation details.
4. Design and implementation of sentiment analysis using NLTK on "Twitter Samples" dataset (40% complete).
5. Project proposal and project progress report.

5. Tasks Remaining

1. Complete the remaining sentiment analysis implementation using NLTK.
2. Complete sentiment analysis implementation using Amazon Comprehend.
3. Do analysis and comparison of both results.
4. Complete the project report.

6. Changes Based on Review / Feedback

There were no specific changes mentioned in the review of the project proposal. A comment was made regarding the effort to implement the project. We are pretty confident that the required amount to finish the total project will take well beyond the 20 hours suggested effort for a single person team.