Sentiment Analysis of Twitter using Semantic Analysis.

Divya Seshani- N-Batch- 2018103531
Vignesh Kumar Murugavel- N-Batch- 2018103079
Jerrick Gerald- N-Batch- 2018103031

November 8, 2020

Abstract. Twitter is a free social networking microblogging service that allows registered members to broadcast short posts called tweets.Twitter members can broadcast tweets and follow other users' tweets by using multiple platforms and devices. Tweets are permanent, they are searchable and they are public. Anyone can search tweets on Twitter, whether they are a member or not. The publishing tweets may be a good tweet or a bad tweet. In this project,we classify the tweets into positive and negative tweeets. We first preprocess the tweets to remove unnecessary content in tweet; we then extract the adjectives which forms the feature vector, then we use the concept of sentiment analysis by classifying the tweets with the help of algorithms like Logistic Regression.

1 Introduction

Twitter is an American micro-blogging and social networking service on which users post and interact with messages known as "tweets". The growing popularity of twitter has been unexpected augmentation in the opinion-forming. But the opinion so formed remains incomplete in the absence of adequate strategical analysis. The sentiment analysis is at present the most important field in the development of an organization and thus many Machine Learning technologies have been applied to automate the work. A wide range of features and methods for training sentiment classifier have been researched in the recent years with varying results. This project aims to add semantics as additional feature into the dataset for sentiment analysis, for each extracted entites we add the corresponding semantic concept as additional feature. This approach could yield better accuracy from the classification model.

2 Dataset

The dataset contains 1,600,000 tweets extracted using the twitter api which is an open source data. The tweets have been annotated (0 = negative, 1 = positive) and they can be used to detect sentiment. It contains the following 6 fields (Refer Fig1).

Target	ID	User
Flag	Query	Tweets

Table 1: Features

- Target: The polarity of the tweet (0 = Negative,1 = Positive).
- ID: The id of the tweet (2087).
- Date: The date of the tweet (Sat May 16 23:58:44 UTC 2009).
- flag: The query (lyx).
- user: The user that tweeted (@vickykumar2k).
- text: The text of the tweet (Vicky is cool).

3 Architecture

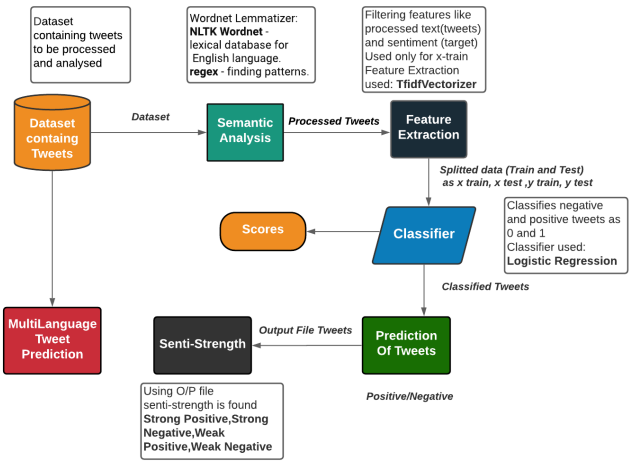

Figure 1: Architecture

4 Data Preprocessing

Data preprocessing is a technique that allows transformation of raw/vague data into an understandable format. Data we obtain from different platforms is often

unstructured, incomplete, inconsistent, and/or lacking in certain traits and is likely to have ambiguities. Data preprocessing helps resolve these problems. It prepares raw data for further processing.

	target	id	time	query	user	tweet
0	0	1467810369	Mon Apr 06 22:19:45 PDT 2009	NO_QUERY	_TheSpecialOne_	@switchfoot http://twitpic.com/2y1zl - Awww, t...
1	0	1467810672	Mon Apr 06 22:19:49 PDT 2009	NO_QUERY	scotthamilton	is upset that he can't update his Facebook by ...
2	0	1467810917	Mon Apr 06 22:19:53 PDT 2009	NO_QUERY	mattycus	@Kenichan I dived many times for the ball. Man...
3	0	1467811184	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	ElleCTF	my whole body feels itchy and like its on fire
4	0	1467811193	Mon Apr 06 22:19:57 PDT 2009	NO_QUERY	Karoli	@nationwideclass no, it's not behaving at all....

Figure 2: Sample dataset

5 Semantic part - Module 1

In this phase, we clean the data on the following aspects: stop words removal, stemming, punctuation marks, lowercasing the data, removing repeated words/characters, lastly, after removing these unusable words,remaining data will be tokenized in tokens.

This process is of utmost importance as it is here we filter the text and keep the one that is valuable for us .i.e. just the one that has the actual and relevant content. Removing Stopwords and Lemmatizing: Lemmatization is the process of converting a word to its base form. (e.g: "Great"to "Good") are done by NLP concept.

```
of info already knew ',
 'Home really wana sleep but due to wasting my free line in town have an assignment to finish ',
 'AT USER also send some update in plurk but upload photo on twitter you didnt see any of my update
on plurk Zero ',
 'omg quot The Reader quot is making me ',
 'AT USER oh At least you re getting decent exchange rate at the moment sterling is still getting f
logged ',
 'tried to download tweetdeck but it wont download ',
 'There an inch of snow on the ground and counting worried about the poor flower ',
 'AT USER why aren you happy camper ',
 'AT USER thanks man so very grateful feel unworthy of such attention though because in this becaus
e of myself ',
 'AT USER miss too totally comin back tho Lastnight wa soo much fun ',
 'AT USER ohh love it p sad we didn get to hang out ',
 'And somehow still end up in this place ',
 'AT USER oh that is very sad poor boy ',
```

Figure 3: Bar plot

```
wordLemm = WordNetLemmatizer()
def preprocess_tweet(data):
    preprocessed=[]
    for tweet in data:
        tweet = re.sub('((www\.[^\s]+)|(https?://[^\s]+))','URL',tweet)
        tweet = re.sub('@[^\s]+','AT_USER', tweet)
        tweet = re.sub('[\s]+', ' ', tweet)
        tweet = re.sub(r'#([^\s]+)', r'\1', tweet)
        tweet = re.sub("[^a-zA-Z0-9]"," ",tweet)
        sequencePattern   = r"(.)\1\1+"
        seqReplacePattern = r"\1\1"
        tweet = re.sub(sequencePattern,seqReplacePattern,tweet)
        tweetwords = ''
        for word in tweet.split():
            if len(word)>1:
                word = wordLemm.lemmatize(word)
                tweetwords += (word+' ')
        preprocessed.append(tweetwords)
    return preprocessed
```

Figure 4: Code

5.1 Data Distribution

The distributed data is visually shown in fig4. This is used to check whether the dataset is balanced or not. From the fig we can conclude that the dataset is equally balanced where is equal number of positive tweets and negative tweets in the dataset.

```
Out[111]: <matplotlib.axes._subplots.AxesSubplot at 0x26fb7be5c88>
```

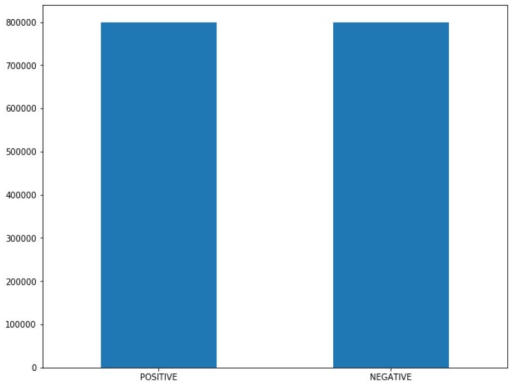

Figure 5: Code

6 Feature Selection - Module 2

The main purpose of feature selection is to,

- Reduce complexity of the model.
- Improves accuracy of the model.
- Make the machine learning algorithm to train faster.

For sentiment classification, the features are mostly the terms or phrases that influence the sentiment of text. The filtered dataset post preprocessing has a lot of distinctive properties. The feature extraction method extracts the part-of-speech from the dataset. Tfid Vectorize feature is used. It Transforms text to feature vectors that can be used as input to estimator. vocabulary Is a dictionary that converts each token (word) to feature index in the matrix form.

6.1 Train Test Split:

The selected features are splitted into Training and Testing Sets.

- Training-80%:
  - The actual dataset that we use to train the model.The model observes and learn from this data.
- Testing-20%:
  - The sample of data used to evaluation of a final model fit on the training dataset.

```
Train Values
(1120000,)
(1120000,)
-----------------
Test values
(480000,)
(480000,)
```

Figure 6: Code

7 Classification - Module 3

The purpose of classification is to find whether the tweet is positive or negative. The classifier used is Logistic Regression.

7.1 Classification Metrics:

Accuracy,Precision, Recall and F1-score were selected as classification metrics for comparing the performance results of different models with respect to two classes.

- Class 0-Negative Tweet.
- Class 1-positive Tweet.

Accuracy:
Accuracy is one metric for evaluating classification models

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \quad (1)$$

Precision:
Precision is defined as the proportion of correctly predicted positive observations of the total predicted positive observations.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2)$$

Recall:
Recall or Sensitivity is defined as the fraction of correctly identified positives.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

F1-Score:
F1-Score is the harmonic mean (HM) of Precision and Recall.

$$\text{F}_1 - \text{Score} = \frac{2 \times (Precision) \times (Recall)}{Recall + Precision} \quad (4)$$

- TN-True Negative.
- TP-True Positive.
- FP-False Positive.
- FN-False Negative.

8 Prediction of tweets Module-4

The tweets are seperated as positive or negative tweets after classfication. A sample dataset containing 15000 tweets is tested without any sentiment label (0 or 1) and it is found that the model correctly predicts the output and saved in output file.

	Tweet	sentiment
0	@user when a father is dysfunctional and is s...	NEGATIVE
1	@user @user thanks for #lyft credit i can't us...	NEGATIVE
2	bihday your majesty	POSITIVE
3	#model i love u take with u all the time in ...	POSITIVE
4	factsguide: society now #motivation	POSITIVE

Figure 7: Scores

9 Sentiment Strenght Module-5

Sentiment strength estimates the strength of positive and negative sentiment in texts, even for informal language. Polarity scores are used to find senti-strenght. Polarity values ranges from -1 (strongly negative) to +1 (strongly positive). Based on polarity of each tweets, they are classified as below. We calculate polarity scores

Senti-Strenght	Polarity
Strongly Positive	Polarity = +1
Weakly Positive	Polarity > 0
Weakly Negative	Polarity > -0.9
Strongly Negative	Polarity = -1

Table 2: Polarity Scores

using TextBlob which is a python library for processing textual data.

	Tweet	polarity
0	@user when a father is dysfunctional and is s...	-0.500000
1	@user @user thanks for #lyft credit i can't us...	0.200000
2	bihday your majesty	0.000000
3	#model i love u take with u all the time in ...	0.976562
4	factsguide: society now #motivation	0.000000

Figure 8: Scores

10 Multi-language Testing Module-6

We also included a new appoarch of finding the sentiment analysis of foreign languages. Some of the language we used are Tamil, French, Spanish, Italian, Arabic, Chinese , Hindi, and many more.

We used Multilingual tweets dataset for this module, which is an open source dataset. The foreign language is changed into English and allowed the tweets to performs semantic and classification task and finally it predicts the sentiment and senti-strenght of the tweets.Example,

- ARABIC:
  الجميع من فضلك لا تطير مع. كان لديه تجارب قليلة رهيبة. لديهم خدمة سيئة للغاية ورحلاتهم الجوية تتأخر دائمًا فدب سفر
  رجاء الجميع لا يطير مع الخطوط الجوية. كان لديه تجارب قليلة رهيبة. لديهم خدمة سيئة للغاية ورحلاتهم الجوية تتأخر دائمًا
- THAI:
  เจนไน ทำได้ดีมากในวันนี้โ แซม ดยเ
  เล่นเตะที่ยอดเยี่ยมเพื่อเอาชนะทีม เอ้ย
  ทุกคนโปรดอย่าบินกั สายการบิน มีประสบการณ์ที่แย่-มาก พวกเขามีบริการที่แย่มากและเที่ยวบินมักจะล่าช้าเสมอ โอ้พระเจ้า
- FRENCH:
  Csk se débrouillait très bien aujourd'hui, surtout que Samcurran a joué un excellent coup de pied pour gagner l'équipe. Oh mon Dieu

  Tout le monde, s'il vous plaît, ne volez pas avec @airways. J'ai eu quelques terribles expériences. Ils ont un service très médiocre et leurs vols sont toujours retardés fedup travelwoes
- GERMAN:
  Csk ging es heute großartig, besonders Samcurran spielte einen großartigen Kick, um das Team zu gewinnen. Oh mein Gott

  Jeder bitte nicht mit @airways fliegen. Hatte ein paar schreckliche Erfahrungen. Sie haben einen sehr schlechten Service und ihre Flüge sind immer verspätet satt Travelwoes
- CHINESE:
  @csk 今天的表現□棒，尤其是 @samcurran 發揮了驚人的敲門聲，□球隊□得了勝利。 # 連勝

  每個人都不要與 @airways 一起飛翔。經歷了幾次可□的經歷。他們的服務質量□差，而且航班總是延誤# fedup # travelwoes
- KOREAN:
  @csk 의오늘공연은훌륭했습니다. 특히 @samcurran 는팀의승리를위해놀라운노크를했습니다. 연승

  모두 @dvjairways 와함께비행하지마십시오. 끔찍한경험을했습니다. 그들은서비스가매우열악하고항공편이항상지연됩니다. 먹고싶다
- ENGLISH:
  Today's Performance by @csk was great especially @samcurran played an amazing knock to get the win for the team . winningstreak.

  Everyone please dont fly with @airways. Had a terrible few experiences. They have very poor service and their flights are always delayed.

The above tweets convey the same meaning in different languages. Here, in the example, we showed one positive and one negative tweet where it undergoes all the semantic, classifier part and gives us Sentiment.

	Tweet	sentiment
0	Bugün bulusmami lazimdiii	POSITIVE
1	Volkan konak adami tribe sokar yemin ederim :D	POSITIVE
2	Bed	POSITIVE
3	I felt my first flash of violence at some fool...	NEGATIVE
4	Ladies drink and get in free till 10:30	POSITIVE

Figure 9: Scores

11 Results

In this section, we discuss the findings obtained through Logistic Regression and XGBoost and we compare their relative performances based on three parameters: Accuracy, Precision, and Recall

LogisticRegression:

	precision	recall	f1-score	support
NEGATIVE	0.83	0.81	0.82	239877
POSITIVE	0.82	0.83	0.83	240123
accuracy			0.82	480000
macro avg	0.82	0.82	0.82	480000
weighted avg	0.82	0.82	0.82	480000

Figure 10: Report

Logistic Regression classifier was found to be the best performing classifier because it had a good Accuracy of %82. Hence it is a balanced dataset we concentrate more on accuracy. A confusion matrix is represented below to show TN,TP,FP,FN.

12 Conclusion

In this paper, we proposed a set of machine learning techniques with semantic analysis. Based on tweet sentiment score, the words from each sentence will get score by which intention he/she have written that tweet. Then we segregated the tweets in different classes, positive

```
]: pred1=XGB1.predict(X_test)
   print("XGBClassifier:\n",metrics.classification_report(y_test,pred1))
```

```
XGBClassifier:
              precision    recall  f1-score   support

    NEGATIVE       0.79      0.70      0.74    239877
    POSITIVE       0.73      0.81      0.77    240123

    accuracy                           0.75    480000
   macro avg       0.76      0.75      0.75    480000
weighted avg       0.76      0.75      0.75    480000
```

Figure 11: Scores

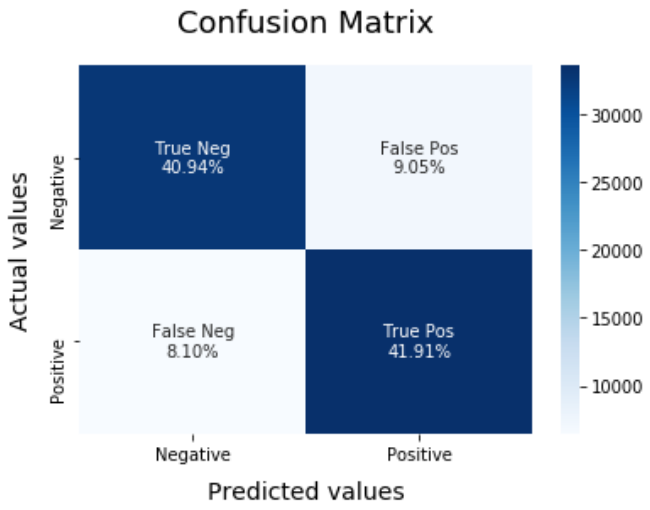

Figure 12: Confusion matrix

or negative, based on the sentiments identified in the tweets. Sentiment strenght is also defined for the tweets using polarity scores. FInally we also included foreign languages for sentiment analysis and therefore it gave us good results.

References

[1] Sentiment analysis of twitter tweets using semantic analysis. Snehal Kale, Vijaya Padmadas, Mumbai.

[2] Semantic analysis of twitter post
Malika Acharya, Shilpi Sharma,Noida.