Twitter Sentiment Analysis using Nature Inspired Algorithm

Step-1: Preprocessing

The raw tweets, collected from Twitter, have noise in terms of unwanted and fuzzy words, URLs, stopwords etc., which are needed to be reduced before feature extraction. The preprocessing method uses following two phases before extracting the features:

Phase 1

This phase eliminates unwanted noise elements from the Twitter data set using the following steps:

1. Eliminate all the URLs via regular expression matching.

2. Replace "@Username" with "usr" using regular expression matching.

3. Since "hash-tag(#)" provides some useful information, therefore remove only #, keeping the word as it is.

4. Remove parenthesis, forward slash (/), backward slash (\), and dash from tweets.

5. Replace multiple white spaces with single white space.

Phase 2

In this phase, two dictionaries namely; stop word and acronyms have been used to improve the precision of resultant Twitter dataset of Phase 1. The steps of Phase 2 are as follows:

1. Convert all the words of tweets into lowercase.

2. Remove all the stop words such as, a, is, the, etc. by comparing them with stop word dictionary.

3. Replace sequence of repeated characters (three or more) in a word by one character viz., "hellooooo" is converted to "Hello".

4. Eliminate words which do not start with an alphabet.

Step-2 Feature extraction

After applying the preprocessing, tweets are converted into the feature vector by calculating the following 11 features from the Twitter dataset.

1. **Total Characteristics:** It represents the total number words available in the tweets.

2. **Positive Emoji:** Positive emoji, such as :), ;), : D, etc., are the symbols used to express happy moments. This feature uses a positive emoticon dictionary to count the total number of positive emojis in the tweets.

3. **Negative Emoji:** The special symbols used to express sad/ negative feelings, such as : (, :, (, > : (, etc., are known as negative emoji. To get the total counts of negative emoji in tweets a negative emoticon dictionary is used.

4. **Neutral Emoji:** Neutral emoji (straight-faced emoji) do not provide any particular emotion. Total neutral emoji is counted by comparing tweets with neutral emoticon dictionary.

5. **Positive Exclamation:** Exclamatory words, such as hurrah! wow! etc., can be used to convey a very strong feeling/ opinion about the topic. For the same, positive exclamation dictionary is used to count the positive exclamation.

6. **Negative Exclamation:** Negative exclamations are counted by comparing the tweet with negative exclamation dictionary.

7. Negation: To express the negative opinion, negations words like no, not, etc., are generally used. Therefore, this feature counts the negation words in the tweet by comparing it with negation words.

8. **Positive Words:** This feature counts the number of positive words like achieve, confidence, etc., using positive word dictionary. If there are two negative words (double negation) then these words are counted as single positive word.

9. **Negative Words:** This feature represents the total counts of negative words such as bad, lost, etc., in tweets.

10. **Neutral Words:** Neutral words (okay, rarely) do not provide any particular emotion/feeling. Total counts of neutral words are obtained by comparing the tweets with neutral word dictionary.

11. **Intense Words:** Intense words, like very, much etc. are used in a sentence to make it more effective/intense. Total counts of intense words are determined by using intense word dictionary.

Generated feature vector after step 2

Total	Positive_E	Negative_	Epos	Eneg	Eneut	Negation	Positive	Negative	Neutral	Intensity
15	0	0	0	0	0	0	0	0	0	0
12	1	0	0	0	0	0	0	0	0	0
11	0	0	1	0	0	1	0	0	0	0
27	0	1	0	0	1	0	0	4	0	0
12	0	0	0	0	0	0	0	0	1	0
12	1	0	0	1	0	0	0	0	0	0
19	0	0	0	0	0	0	1	0	0	0
22	1	0	1	0	0	1	1	0	0	1
26	0	0	0	0	0	0	2	0	0	0
6	0	0	0	0	1	0	0	0	1	0
20	0	0	0	0	0	0	1	0	0	0
11	0	0	0	0	0	0	0	0	1	0
22	1	1	0	1	0	0	2	0	0	0
8	0	0	0	0	0	1	0	0	0	0
8	0	0	1	0	1	0	0	0	2	0
26	0	0	0	0	0	0	1	0	0	0
4	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	1	0	0	0	0
20	1	0	0	1	0	0	1	0	0	0
14	0	0	0	0	1	0	0	0	0	0
17	0	0	1	0	0	0	1	0	0	0
12	0	0	0	0	0	1	0	0	1	0
20	0	1	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	1	0
8	0	0	0	1	0	0	0	0	1	0
13	1	0	0	0	1	1	0	0	0	0
19	0	0	0	1	0	0	1	0	0	0
26	0	0	0	0	0	0	2	0	0	0
22	0	0	0	0	0	0	1	0	0	0

Step-3 Feature normalization

Total	Positive_E	Negative_	Epos	Eneg	Eneut	Negation	Positive	Negative	Neutral	Intensity
0.580645	1	0	0	0	0	0	0	0	0	0
0.516129	1	0	0	0	0	0	0	0	0	0
0.354839	1	0	0	0	0	0	0	0	0	0
0.451613	1	0	0.5	0	0	0	0	0	0	0
0.387097	1	0	0	0	0	0	0	0	0	0
0.387097	1	0	0	1	0	0	0	0	0	0
0.516129	1	0	0	0	0	0	0	0	0	0
0.516129	1	0	0	0	0	0	0	0	0	0
0.354839	1	0	0	0	0	0	0	0	0	0.5
0.225806	1	0	0	0	0	0	0	0	0	0
0.064516	1	0	0	0	0	0	0	0	0	0
0.225806	1	0	0	0	0	0	0	0	0	0
0.548387	1	0	0	0	0	0	0	0	0	0
0.516129	1	0	0	0	0	0	0.333333	0	0	0
0.580645	1	0	0	0	0	0	0	0	0	0
0.516129	1	0	0	0	0	0	0	0	0	0
0.354839	1	0	0	0	0	0	0	0	0	0
0.451613	1	0	0.5	0	0	0	0	0	0	0
0.387097	1	0	0	0	0	0	0	0	0	0
0.387097	1	0	0	1	0	0	0	0	0	0
0.516129	1	0	0	0	0	0	0	0	0	0
0.516129	1	0	0	0	0	0	0	0	0	0
0.354839	1	0	0	0	0	0	0	0	0	0.5
0.225806	1	0	0	0	0	0	0	0	0	0

Step-4 clustering using nature inspired algorithms

Choose any clustering algorithm and form the desired number of clusters (k=2 or 3) depending upon the datasets.

Step-4 Evaluate the performance

Compare the actual class label with predicted class label and determine the accuracy of classifier.