

# IRSW Project Report

## Twitter Emoticon Analysis



Department of CSE/IT  
Jaypee Institute of Information Technology, Noida

**Submitted to:**  
Dr. Neetu Sardana

**Submitted By:**

<b>Abhinav Modi</b>	<b>19104035</b>	<b>B12</b>
<b>Rishabh Kaushik</b>	<b>19104036</b>	<b>B12</b>
<b>Harsh Jain</b>	<b>19104037</b>	<b>B12</b>
<b>Govind Yadav</b>	<b>19104038</b>	<b>B12</b>

## Introduction

Twitter is a powerful social media where people share their opinion on various topics. Sentiment Analysis on twitter data gives the classification of opinion on a topic as positive, negative or neutral. Twitter messages are written informally and tweets are short. Hence, the classification of tweets by only considering the text part of the message does not give accurate results. To improve the classification accuracy we use Emotion Tokens like Emoticons or Emojis. Emotion Tokens are independent of language, grammar or size of the tweet. Considering Emotion tokens while classifying tweets will improve the accuracy of classification. In this project, we focus mainly on twitter sentiment analysis considering emoji & emoticons as well to improve the efficiency and classification accuracy.

## Problem Statement

Microblogging websites such as Twitter ([www.twitter.com](http://www.twitter.com)) have evolved to become a great source of various kinds of information. This is due to the nature of microblogs on which people post real time messages regarding their opinions on a variety of topics, discuss current issues, complain, and express positive or negative sentiment for products they use in daily life. As the audience of microblogging platforms and social networks grows every day, data from these sources can be used in opinion mining and sentiment analysis tasks. For example, manufacturing companies may be interested in the following questions:

- What do people think about our product (service, company, etc.)?
- How positive (or negative) are people about our product?
- What would people prefer our product to be like? Political parties may be interested to know if people support their program or not. Social organizations may ask people's opinion on current debates. All this information can be obtained from social networks, as their users post everyday what they like/dislike, and their opinions on many aspects of their life.

Opinions and its related concepts such as sentiments and emotions are the subjects of study of sentiment analysis and opinion mining. The inception and rapid growth of the field coincide with those of social media on the Web. e.g., reviews, forum discussions, blogs, microblogs, Twitter, and social networks. Most NLP based

methods perform without particular success in social media. Almost all forms of social media are very noisy and full of all kinds of spelling, grammatical, and punctuation errors.

## Dataset Used

Using the emoticon dataset we simulated, replace, and added utf-8 emoticons the dataset

- **1.6m Tweet Dataset**

<https://www.kaggle.com/kazanov/sentiment140>

We downsized the 1.6m tweet dataset to 10k. The name tags and links were also removed since most would not contribute to the value of the sentiment and converted the sentiment values to 0 and 1.

- **Emoticon Dataset**

[https://www.kaggle.com/thomasseleck/emoji-sentiment-data?select=Emoji\\_Sentiment\\_Data\\_v1.0.csv](https://www.kaggle.com/thomasseleck/emoji-sentiment-data?select=Emoji_Sentiment_Data_v1.0.csv)

The emoticon dataset although contains sentiment, the sentiment is formatted by voting value of the data acquirer. It also contains 3 polarities (positive, neutral, negative) to set the sentiment of the emoticon, we compare the values of the positive and negative, whichever is higher is set as the sentiment polarity. We normalize the value to 0 and 1.

- **3 part tweet dataset**

<https://www.kaggle.com/shashank1558/preprocessed-twitter-tweets>

The formatting of the dataset is listed by column and contains 3 files, positive, negative, and neutral tweet list. We create a new data frame and list the dataset by row and the sentiment polarity depends on what file the tweets were taken from.











The tweets contain tags and links which need to be removed. Since the raw data is already a cleaned tweet the emoticons were transformed into plain words or words with symbols. ASCII emoticons and word emoticons were converted to utf-8 emoticons. Emoticons were also randomly added by sentiment to balance the data.




## Implementation

- **Data Preprocessing**

We will be importing emoticons and tweets dataset which are available from the internet. The tweet dataset we downloaded is a processed data that converted the emoticons into words and contains sentiment polarity, most of the available dataset with sentiment polarity are already processed. We will be manipulating and simulating the tweet data set and adding the emoticons while retaining the sentiment polarity. This will ensure that the emoji icons will have the corresponding sentiment polarity we need.

The provided data was processed and downsized from the original 1.6m tweet dataset. We will remove words containing ('@', 'http://', '&', '#'). We will remove them since hyperlinks and tags don't add much to the sentiment of the post. Some tweets only contain tags. So post pre-processing, tweets might now contain empty cells, we will fill it up with '...'.

Emoticon	Emoji
:)	
:P	
:D	
:	
:'(	
:O	
:*	
<3	
:(	
;)	

xD	
:/	
=D	

We used two methods to find the sentiment of tweets.

- **First Method**

This method extracts and separates the emoticon and texts from the input tweet.

The emoticon dataset already contains sentiment polarity which we will use to analyze the extracted emoticons.

We train our processed tweet dataset using Naive Bayes to set our sentiments.

We use our trained model to predict the extracted text from our input tweet.

To get the final sentiment polarity, We average the sentiment value of the extracted emoticon and tweets.

- **Second Method**

This method trains our processed dataset as is, where it contains both the emoticons and text.

The tweets contain their respective sentiments which will be used to train our model.

We predict our input tweet & emoji using our model.

This method will also set the sentiment of each emoticon, and depending on the dataset might skew the real sentiment of the emoticon.

## Results

```
print_senti_status("I hate dancing 🙄 🙄 🙄")
```

```
=====
```

```
Your input is "I hate dancing 🙄 🙄 🙄"
```

```
    Extracted: "I hate dancing" , 🙄 🙄 🙄
```

```
    Text value: 0
```

```
    Emoji average value: 0.0
```

```
    Average value: 0.0
```

```
YOUR INPUT IS OF "NEGATIVE" SENTIMENT
```

```
=====
```

```
def get_sentiment(s_input = '😄 I love sentiment analysis 😄'):
    # turn input into array
    input_array = np.array([s_input])
    # vectorize the input
    input_vector = vectorizer.transform(input_array)
    # predict the score of vector
    pred_senti = clf.predict(input_vector)

    return pred_senti[0]
```

```
get_sentiment("I am happy 😄 😄 😄")
```

1

## Conclusion

Microblogging like twitter nowadays has become one of the major types of communication. People share their opinion in the form of tweets. These opinions are useful when other people can use the analyzed results with the help of Sentiment analysis. Tweets are written informally and have size restrictions. Hence, Emoticons play a vital role in classification of tweets along with the text.

The large amount of information contained in these web-sites makes them an attractive source of data for opinion mining and sentiment analysis. Most text based methods of analysis may not be useful for sentiment analysis in these domains. To make significant progress, we still need novel ideas. Using twitter names and hashtags to collect training data can provide better results. Also adding symbol analysis using emoticons and emoji characters can significantly increase the precision of recognizing emotions. The most successful algorithms will probably be integration of natural language processing methods and symbol analysis.

## References

- Yuki Yamamoto, Tadahiko Kumamoto, and Akiyo Nadamoto. 2014. **Role of Emoticons for Multidimensional Sentiment Analysis of Twitter**. In Proceedings of the 16th International Conference on Information Integration and Web-based Applications & Services (iiWAS '14). Association for Computing Machinery, New York, NY, USA, 107–115. <https://doi.org/10.1145/2684200.2684283>
- Wegrzyn-Wolska, Katarzyna & Bougueroua, Lamine & Yu, Haichao & Zhong, Jing. (2016). **Explore the Effects of Emoticons on Twitter Sentiment Analysis**. 65-77. 10.5121/csit.2016.61006.
- Wolny, Wieslaw. (2016). **Twitter Sentiment Analysis Using Emoticons And Emoji Ideograms**.
- S. Hiremath, S. H. Manjula and V. K. R, "Unsupervised Sentiment Classification of Twitter Data using Emoticons," 2021 International Conference on Emerging Smart Computing and Informatics, 2021, pp. 444-448, DOI: 10.1109/ESCI50559.2021.9397026.