# Sentiment Analysis of Twitter Post

Yuming Chang
Georgia Institute of Technology
Atlanta, USA
ychang394@gatech.edu

## 1 INTRODUCTION

In recent years, social media has become increasingly critical to people's lives. People share their life and comments about the movie, services, news, and feelings on social media. It's essential for companies and policymakers to understand the users' sentiments on specific events to enhance the service and provide customized content to serve users and make more profit. So it is essential to do sentiment analysis to achieve those goals. Twitter is one of the largest companies with many active users on their platform, providing sufficient amounts of data to do sentiment analysis. In recent years, many researchers have put effort into using machine learning models such as Support Vector Machine (SVM), Decision Trees, and CNN to analyze the sentiment of the text. In this project, serval machine learning and deep learning models will be implemented for sentiment analysis and prediction.

## 2 PROBLEM DEFINITION

Using the Natural Language Process(NLP) method, sentiment analysis can automatically extract the information from the text and classify the sentiment from those text[1] . In order to use the machine learning model to do the sentiment analysis, we need to convert the text into numeric values using Natural Language Process methods. So the problem in this project will be to convert the text information into a numeric value and train and test models using those data.

## 3 RELATED WORK

In recent years, plenty of researchers have focused on using different methods to analyze the text's sentiment. Dey[2] has developed a Naive Bayes classifier to do the sentiment analysis. Assuming that each word is independent, the Naive Bayes classifier they developed is based on the Bayes theorem by using the frequency of the words and the text category as the feature to build the joint probability. The strength of the naive Bayes classifier is compared with other machine learning models, and the naive Bayes classifier requires fewer data to train. However, the weakness of the naive bayes model they proposed is that in this model, they assume each word appears independently, which in some cases is not valid, and each word may have a different meaning when they appear in the word combination. Astri[**?** ] uses Multinomial Logistic Regression which uses the softmax activation function to do the logistic regression. The strength of this method is that it requires a small amount of data to train the model and does well even though the input features has a relationship of multi-linear. Moreover, the weakness of those methods is that this model assumes the input variable has a linear relationship with each other, which may not be accurate in the text classification. Tripathi[3] develop the random forest model to do the sentiment analysis on the Twitter text. The random forest model produces the output by combining the results of the different decision trees in the random forest model. Furthermore, in their research, they picked up some features from the dataset as the input feature, then based on the input features selected to determine the best number of nodes in each decision tree and built multiple decision trees. The strength of the random forest model is that this model can often generate good classification results and deal with missing data. However, the weakness of the random forest model is that sometimes they need lots of features to fit the model, but in the text classification task, we need to avoid the overfitting issue during the training period by carefully choosing several features. Also, we need to address that the training time of the random forest model may be longer compared with the primary decision tree model or other machine learning models. Rahman[4] developed the Support Vector Machine (SVM) model to do the sentiment analysis. In their research, they use the Principle Component Analysis method to find the essential features using the matrix build by calculating the weights of different features; then, they use grid research to find the best parameter in the SVM models. The advantage of the SVM model is that when the dataset is highly dimensional, the performance of the SVM model will perform well. However, the weakness of their work is that they use a small number of data to train their model in this research, but the problem is that the SVM model may perform worse when the dataset is too large, so we need to see how the performance of SVM model when dataset become large. Malarvizhi[5] has developed the Convolutional Neural Network (CNN) model to analyze sentiment. They first convert the different lengths of text data into the matrix with the same dimension and pad those matrices together. After that, they put the padded matrix into the embedding layer of the CNN model. After generating the embedding results, the embedding results will go through the convolutional layer to get the classification results. The strength of this method is that the CNN model has multiple layers, which enable the model to capture some vital information from the text. However, the weakness of the CNN model is that train the CNN model needs a large amount of data, and it is time-consuming to train the CNN models. Swathi[6] proposed the Long-short Term Memory (LSTM) networks to investigate the sentiment of the tweet data. After cleaning the tweet text data by removing the special characters, usernames, and hashtags, they put the processed text dataset into the tokenization module and fed those data into the LSTM model they built. The strength of the LSTM model is that the LSTM model can handle the long sequence of text well compared with other models, but the weakness of the LSTM model is that it needs more data to train.

# 4 MODELS

## 4.1 SVM

*4.1.1 Description.* SVM is one of the supervised learning methods to do classification. By mapping the data to high dimensional space, the algorithm draws the hyperplane to separate different class data. In this project, the SVM model will be used as the baseline model.

*4.1.2 Implementation Details.*

(1) Covert text data into the matrix using the TF-IDF method
(2) Separates the total data into training, validation, and test data with the ratio: 7:1:2
(3) Built SVM model using Scikit-learn
(4) Train the SVM model using training data
(5) Use 5 fold cross-validation to evaluate the model
(6) Using test data to evaluate the performance of the SVM model
(7) Generate the classification report and confusion matrix to evaluate the performance of the model

## 4.2 CNN model

*4.2.1 Description.* CNN is the deep learning method model, it has been wildly used in the image classification area, but due to the convolutional layer of the CNN model, we can also use the CNN model to do text classification in the natural language process area. In order to feed data into the CNN model, we need to transform the text data into the matrix. The detailed structure of the CNN model shows in Figure 1. In this project, I am going to use the term frequency-inverse document frequency method to build the vectorized data, then feed the vectorized data into the CNN model. The classification output will be produced after the fully connected layer.

*4.2.2 Implementation Details.*

(1) Covert text data into the matrix using the TF-IDF method.
(2) Separates the total data into training, validation, and test data with a ratio: 7:1:2.
(3) Built CNN model using TensorFlow.
(4) Trains the CNN model using training data.
(5) Using test data to evaluate the performance of the CNN model.
(6) Changes the number of convolutional layers to investigate the model's performance with different layers and get the training time of each CNN model.
(7) Tuning the hyperparameters of the CNN model, such as the learning rate and optimization function, to find the optimal hyperparameters.
(8) Based on the training time and accuracy to find the best combination of the hyperparameters and the number of convolutional layers.
(9) Generate the classification report and confusion matrix to evaluate the model's performance.
(10) Error case analysis.

## 4.3 Graph Convolutional Network Model

*4.3.1 Description.* In recent years, more and more graph or network-based data have been published, such as road network, profile, and

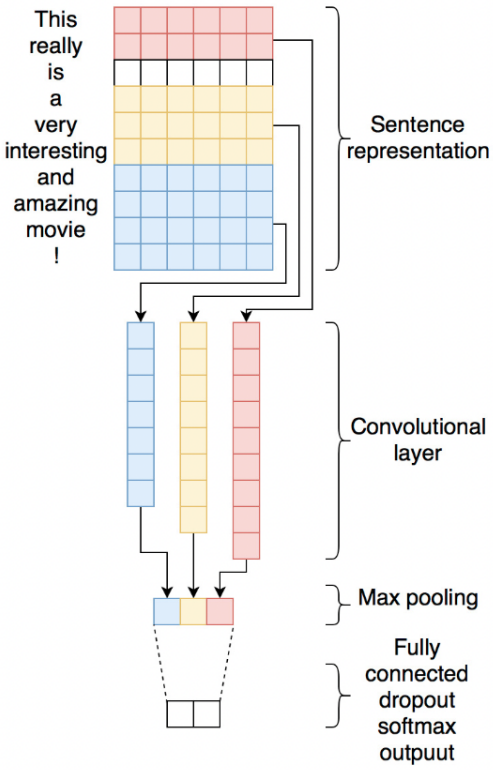


**Figure 1: structure of proposed CNN model[7]**

recommend systems data. The traditional deep learning method have good test performance on the traditional dataset. However the performance of those traditional methods on graph-based or network-based dataset may not be promising. So Graph Convolutional Network(GCN) can help us to overcome the shortcomings of CNN or RNN. GCN model can help us learn the features of graph and network data through its hidden layers. The GCN model structure in this project shows in Figure 2. We also need to construct the text-graph from the original Twitter post. The detail on how to construct the text-graph is in section 4.3.2

*4.3.2 Implementation Details.*

(1) Covert text data into the matrix using the TF-IDF method.
(2) Compute the cosine similarity between each pair of the vectorized data
(3) Using the networkx to build a graph based on the cosine similarity. In this graph, each post is a node.
(4) Built GCN model.
(5) trains the GCN model using training data.
(6) using test data to evaluate the performance of the GCN model.
(7) based on the training time and accuracy to find the best combination of the hyperparameters and the number of hidden layers.
(8) Generate the classification report and confusion matrix to evaluate the model's performance.

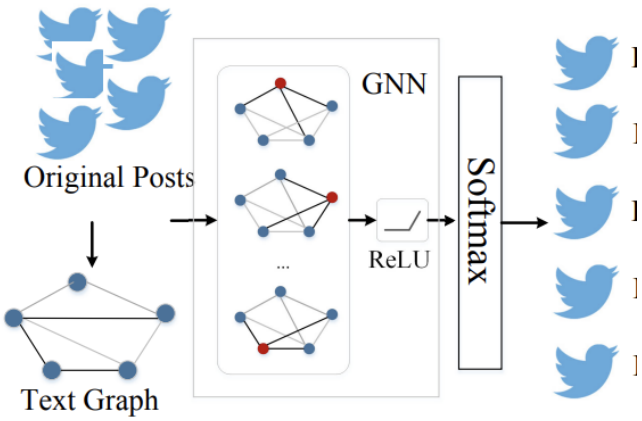


**Figure 2: structure of proposed GCN model[8]**

(9) Error case analysis.

## 5 PLATFORM

In this project, each model will be implemented and tested in Google Colab using Python, with Pandas, Numpy, Sciti-learn, and Tensorflow, Pytorch with the Google Colab default environments.

## 6 DATA

In this project, Sentiment140 dataset[9] will be used to train, validate and test the models built. This dataset was stored in a CSV file—the detail of the dataset provided in Figure 3. From Figure 3, we can know there are 1600000 data in this dataset. Moreover, have 6 features in this dataset, which are sentiment, ids, date, flag, user, and text. In this dataset, the sentiment information is already be stored as numeric values (0-negative, 4-positive). The distribution of data based on the sentiment information is shown in Figure. We can see this dataset contains 80000 negative data and 80000 positive data. From Figure 4, some space characters, website information, and hashtags need to be removed from the text to improve the performance of models. We also need to convert all the Uppercase characters to lowercase characters. In this project, the whole dataset will be split into training, validation, and test dataset using the ratio 7:2:1.

| | sentiment | ids | date | flag | user | text |
|---|---|---|---|---|---|---|
| 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.... |
| ... | ... | ... | ... | ... | ... | ... |
| 1599995 | 4 | 2193601966 | Tue Jun 16 08:40:49 PDT 2009 | NO_QUERY | AmandaMarie1028 | Just woke up. Having no school is the best fee... |
| 1599996 | 4 | 2193601969 | Tue Jun 16 08:40:49 PDT 2009 | NO_QUERY | TheWDBoards | TheWDB.com - Very cool to hear old Walt interv... |
| 1599997 | 4 | 2193601991 | Tue Jun 16 08:40:49 PDT 2009 | NO_QUERY | bpbabe | Are you ready for your MoJo Makeover? Ask me f... |
| 1599998 | 4 | 2193602064 | Tue Jun 16 08:40:49 PDT 2009 | NO_QUERY | tinydiamondz | Happy 38th Birthday to my boo of alll time!!! ... |
| 1599999 | 4 | 2193602129 | Tue Jun 16 08:40:50 PDT 2009 | NO_QUERY | RyanTrevMorris | happy #charitytuesday @theNSPCC @SparksCharity... |

1600000 rows × 6 columns

**Figure 3: screenshot of dataset**

```
data['sentiment'].value_counts()

0    800000
4    800000
Name: sentiment, dtype: int64
```

**Figure 4: number of data of each sentiment**

## 7 EVALUATION METHOD

In this project, the models will be evaluated based on the accuracy and F1 score. And based on the accuracy and F1 score to get the best performance model through the project. The formula of the accuracy and F1 score is shown in Figure 5

| Evaluation Method | Equation |
|---|---|
| Accuracy | $\frac{Total\ Number\ of\ Correct\ Predicton}{All\ Number\ of\ prediction}$ |
| Recall | $\frac{True\ Positive}{True\ Positive + False\ Negitive}$ |
| Precision | $\frac{True\ Positive}{True\ Positive + False\ Postitive}$ |
| F1-score | $2 * \frac{Precision * Recall}{Precision + Recall}$ |

**Figure 5: number of data of each sentiment**

Also in the model modification process, the confusion matrix of each model will also be used to evaluate performance of each model

## 8 CONTRIBUTION

| | |
|---|---|
| Data Cleaning | Yuming Chang |
| Data Processing | Yuming Chang |
| Build Models | Yuming Chang |
| Evaluate Models | Yuming Chang |
| Modify Models | Yuming Chang |

## 9 TIMELINE

| | |
|---|---|
| 02/23/2023-02/28/2023 | Data Cleaning and processing |
| 03/01/2023-03/15/2023 | Build models |
| 03/16/2023-03/23/2023 | Model training and evaluation |
| 03/24/2023-04/02/2023 | Model modification and final report |

## 10 CONCLUSION

In the project, I am going to implement three machine learning and deep learning models: SVM, CNN, and GCN to do the sentiment analysis and compare their performance and improve the performance of each model by tuning hyperparameters of each model based on the evaluation results to understand each models' weakness and strength better. By the end of this project, I am supposed to find the best model to do the sentiment analysis on the given Twitter text dataset.

## REFERENCES

[1] Doaa Mohey Hussein. A survey on sentiment analysis challenges. *Journal of King Saud University - Engineering Sciences*, 30(4):330–338, 2018. doi: 10.1016/j.jksues.2016.04.002.

[2] Lopamudra Dey, Sanjay Chakraborty, Anuraag Biswas, Beepa Bose, and Sweta Tiwari. Sentiment analysis of review datasets using naïve bayes' and k-nn classifier. *International Journal of Information Engineering and Electronic Business*, 8(4):54–62, 2016. doi: 10.5815/ijieeb.2016.04.07.

[3] Jyotsna Singh and Pradeep Tripathi. Sentiment analysis of twitter data by making use of svm, random forest and decision tree algorithm. *2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT)*, 2021. doi: 10.1109/csnt51715.2021.9509679.

[4] Mohammad Rezwanul, Ahmad Ali, and Anika Rahman. Sentiment analysis on twitter data using knn and svm. *International Journal of Advanced Computer Science and Applications*, 8(6), 2017. doi: 10.14569/ijacsa.2017.080603.

[5] Usha Devi Gandhi, Priyan Malarvizhi Kumar, Gokulnath Chandra Babu, and Gayathri Karthick. Sentiment analysis on twitter data by using convolutional neural network (cnn) and long short term memory (lstm). *Wireless Personal Communications*, 2021. doi: 10.1007/s11277-021-08580-3.

[6] T. Swathi, N. Kasiviswanath, and A. Ananda Rao. An optimal deep learning-based lstm for stock price prediction using twitter sentiment analysis. *Applied Intelligence*, 52(12):13675–13688, 2022. doi: 10.1007/s10489-022-03175-2.

[7] Shiyang Liao, Junbo Wang, Ruiyun Yu, Koichi Sato, and Zixue Cheng. Cnn for situations understanding based on sentiment analysis of twitter data. *Procedia Computer Science*, 111:376–381, 2017. doi: 10.1016/j.procs.2017.06.037.

[8] Yuqing Yang. Covid-19 fake news detection via graph neural networks in social media. *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2021. doi: 10.1109/bibm52615.2021.9669662.

[9] KazAnova. Sentiment140 dataset with 1.6 million tweets, Sep 2017. URL https://www.kaggle.com/datasets/kazanova/sentiment140?resource=download.