# CS5621 - Machine Learning - Twitter Irony Classification

# git: <https://github.com/dilinigunasena/CS5621-ML-Twitter-Irony-Classification.git>

# I used the numpy & pandas for data reading from the given training data set and did data preprocessing by removing the at mark & emoji & other special characters(ex: ! mark) from training tweets. Then did the further cleaning the data set by removing non-interested Index coloumn from the data set.

# This is a binary classification problem hence defined the independent(X) & dependent(Y) variables by analyzing the cleaned data set. Performed the CountVectorizing by using the sklearn`s CountVectorizer in order to extract the features fro the Tweets(X). Then did label encording for the available labels and converted to arry as we have to pass an array to the model.

Then built the first model by using sklearns RandomForestClassifier and splitted the data set in to two parts one part for training & another part for testing. Split 70% of the data as training data and 30% of the rest as testing data. Then trained the model and created a prediction label value for predicting the label for given input tweet from training data. The same approach I followed with the SVM model too as this is a binary classification model. Tried to do fine tuning of the RandomForestClassifier model but default and combination Random Forest Classifier, (n\_estimators=3,min\_samples\_leaf=1) gave the maximum accuracy for the model.

|  |  |
| --- | --- |
| Model Name | Accuracy |
| RandomForestClassifier | 0.8766519823788547 |
| SVM | 0.8766519823788547 |

Hence the both models gave the same accuracy , tested the testing data set by using the Random Forest Classification Model. Preprocessed the given testing data set same as the training data set and predicted the label of the given text by using the Random Forest Classfier model. Added new predicted data to the available test data frame “char\_filtered”. Wrote the predicted data to a csv file , had to modify it adding heading and starting index by 1 and submitted to kaggle.com as a late subbmission.

