Introduction

The data set used that was used in this study includes 1.6 million tweets that were extracted using the Twitter API in 2009. Each data contains text - text of the tweet, target - sentiment of the tweet, user, flag, ids, and date. The emoticon in the text was stripped meaning it would be impossible to determine the sentiment of some of the tweets [3].

The aim of this study is to build machine learning classifiers that determine the sentimental status of a tweet by observing the text inside a Twitter post. We've investigated different methods of preprocessing data and hyper-parameter tuning of the classifier model while considering the computational time.

Sentimental analysis remains a difficult task for machine learning as texts can be inflected, meaningless, and mixed with non-letters. This study will look at how we can deal with some of these problems using different data-processing methods. Also, we will examine which classifier model within the support vector machine, k-nearest neighbour and logistic regression can be used to achieve a high level of sentimental classification accuracy.

Data Pre-Processing

Data preprocessing is an essential part when dealing with text-based classification or in other words natural language processing. Because text data frequently contains specific formats such as number formats, date formats, and the most common terms that are unlikely to help Text mining such as prepositions, articles, and pronouns, they can be removed[5].

1. **Remove Punctuation:**
   Upon analysing the dataset it is evident that the tweets data contains a lot of non-letters and special characters therefore the tweet data is cleaned out by removing these special characters and non-letters.

2. **Lower Case:**
   The entire data is lower-cased to ensure uniformity in the text and help the algorithms learn the words with ease, by making the data more standardised.

3. **Tokenization:**
   Tokenization is the process of converting a continuous stream of text into words, phrases, symbols, or other meaningful items known as tokens. The goal of tokenization is to explore the words in a sentence. The token list is used as input for subsequent processing such as parsing or text mining. Tokenization is useful in both linguistics (as a type of text segmentation) and computer science (as part of the lexical analysis). Textual data is merely a string of characters at first. The words in the data set are required for all procedures in information retrieval[5].Tokenization is mostly used to identify significant keywords. Different number and time formats might cause inconsistencies. Another issue is the transformation of abbreviations and acronyms into a standard form[5].

4. **Stopwords Removal:**

Many words in papers appear frequently yet are essentially meaningless because they are employed to connect words in a phrase. Stop words, as is widely assumed, do not contribute to the context or content of written writings. Because of their frequent occurrence, their existence in text mining creates a barrier to interpreting the content of the texts.
Stop words are often used as common words like 'and,' 'are,' 'this,' and so on.
They are ineffective for document classification. As a result, they must be deleted[5].

5. **Stemming:**
   Stemming is the process of combining a word's different forms into a single representation, the stem. For instance, the words "presentation," "presented," and "presenting" might all be simplified to a single representation of "present." This is a commonly used text processing approach for information retrieval (IR) based on the notion that posing a query with the phrase presenting implies an interest in documents containing the words presentation and presented[5].

Classifier Methods

Logistic Regression

Logistic regression is a supervised machine learning algorithm that is used to classify the dependent variable. In our sentimental data, the only possible dependent variable (that identifies the sentiment) that is available is 0 – Negative or 4- Positive, so we have decided to use the binary logistic regression model. After data processing, our input no longer contains non-meaningful words. This is consistent with the logistic regression assumption, which assumes there is a non-meaningful variable [1].

When training the logistic regression model for this study, a grid search with five-cross validation was used to identify the most optimal hyper-parameter value for our dataset. The hyper-parameter includes to – “tolerance for stopping criteria”, C – “inverse of regularization strength”, and solver – “algorithm (that are used) in the optimisation problem” [8]. The result of the grid search revealed that the C value of 0.1, to the value of 1e-4, and the lbfgs algorithm were most optimal.

Support Vector Machine

The SVM uses a maximum large hyperplane to separate data into classes that are used for the classification [8]. SVM is especially robust to noises and changes in data which will help 3 to minimise the overfitting of the model. As our dataset only had large numbers of features due to the nature of linguistics. SVM classifier is robust for training a dataset with multi-feature by utilising kernel tricks such as linear, poly, rbf, and sigmoid [6]. Our study involves finding linear problems for categorising binary sentiment, so we have considered linear kernel function and rbf kernel function. Rbf function was considered as it “creates non-linear combinations of your features to uplift your samples onto a higher-dimensional feature space” allowing to make linear decisions [4].

Grid search with five-cross validation was used to tune the hyper-parameter of the SVM which includes C - a regularisation parameter to manage to overfit, and kernel – functions. The result of the search showed that the most optimal hyper-parameter was 0.01 for C and rbf kernel function.

Random Forest

Random forest is an ensemble method in which it combines a number of decision trees (in Sci-kit learn) on the subset of the training data, which tends to increase the accuracy and reduces the overfitting of the trained model [1]. From our previous models, we’ve found that there was a discrepancy between accuracy in the training set and accuracy in the test set, where the accuracy of the test set was significantly lower. To deal with this, we wanted to use a model that is less prone to data overfitting.

To find the hyper-parameter that gives the highest accuracy on the testing set, a grid search with 5 cross-validations has been used. From the grid search, it was observed the n-estimator of 256, the criterion of Gini, and max_depth of16 gave the highest accuracy.

Experiment and Results

We have trained three different classifiers algorithm using Sci-kit learn models which include Logistic Regression, Linear Support Vector Machine, and Random forest. Each classifier was trained with preprocessed data and using the best hyper-parameters that were found using 5 cross-validation grid searches. The table below shows the accuracy of each classifier including the training and testing set.

Table 1: Accuracy of Each Classifier 10% of Total Data - Training Set and the Testing Set

	Logistic Regression	Linear SVM	Random Forest
Training Set	81.37%	81.38%	77.79%
Test Set	77.75%	77.9%	75.0%

Table 2: Accuracy of Each Classifier 5% of Total Data - Training Set and the Testing Set

	Logistic Regression	Linear SVM	Random Forest
Training Set	81.28%	81.24%	76.84
Test Set	78.43%	78.46%	74.71%

Training all 1.6 million data was not feasible to be trained with just the large number of data as well as the size of the sparse matrix that will be generated from large word sets. To account for this 5% of the stratified data was used for training and test sets, where 4% was the training set and 1% was the testing set.

Logistic Regression Result

The logistic regression showed very similar accuracy to the SVM only having a 0.15% difference. While the accuracy of the training set is 81.37% which is relatively higher, the test set's accuracy was lower by 3.62%. At this difference, it is hard to assert that there is overfitting of data.

The potential reason for the low accuracy rate could be the limitation of the machine learning model which is not able to distinguish sarcasm in each text and is unable to deal with multipolarity words.

Linear Support Vector Machine Result

Linear SVM showed similar accuracy to the logistic regression and much higher accuracy than a random forest classifier. The difference in accuracy between the training and test set was 3.48% which is only 0.14% different. Comparable to the logistic regression result, it is hard to determine that our model overfits the training set.

The linear SVM has the highest accuracy among other classifiers and the potential reason for this is that SVM is more robust to noises in the data, able to handle the changes in the data, and effective in high dimensional data [3] [13]. This is especially important for sentimental analysis where the size sparse matrix (i.e., training input for the model) is dependent on the number of words in the training set. With the large dataset that we are using, there exists high dimensionality and a number of noises within the data. This can be speculated by the higher accuracy when training a larger number of data and the accuracy of the training set has been improved.

The potential drawback of this algorithm is the same as logistic regression, SVM is unable to handle multipolarity words and sarcasm.

Random Forest Result

The random forest classifier had the lowest accuracy among other classifiers. However, it has the lowest difference in accuracy (i.e., 2.79%) between the training and testing sets. The possible reason for this is that the random forest is not susceptible to noises within the data as it utilises a number of decision trees.

Despite the increase in the training dataset, accuracy has decreased despite that the random forest is robust to noises.

Conclusion

The linear support vector machine classifier produced the highest accuracy on both the training and testing sets. While our classifier was not overfitted to our training model, it did not have strong accuracy within both the test and training set. We speculated that the reason for this was from the different nature of linguistics such as sarcasm and multipolarity words where we require the context of the Tweet. For future work on developing an algorithm for sentimental analysis, we suggest utilising the LSTM deep learning algorithm which is a type of RNN network that is “capable of learning order dependence in sequence prediction problems” [1].