Analysis on Twitter Sentiment with Spark

Authors: CHENG Yu Hong, WONG Kwan Mei, HO Cheuk Bong

1. Introduction

Twitter has been a popular social media site for recent years. Users have been using the sites to express their personal opinions on various topics and issues that they are interested in with different “tweets”, which are short messages that can be up to 280 characters long. With tweets, Twitter users can communicate and engage with each other around the world. The use of Twitter has been widespread and become influential public figures’ favorite platform to share their thoughts. The way of how the general public get the latest news of Tesla, SpaceX, and even the US government has changed from the news to Elon Musk’s and Donald Trump’s twitter accounts.

In October 2022, Elon Musk acquired Twitter and took over the position of Chief Executive Officer. One of the reasons of his action is that he claimed Twitter has too many hate speeches, spambots accounts, and false accusations towards political figures and celebrity. He would like to take the company private, making Twitter a more open space with free speeches.

Therefore, we would like to examinate tweets by analysing their sentiment. This enables us to have an insight on people’s emotion when they are sending these tweets. Sentiment analysis can also let us validate Elon Musk’s claim if it is true or not. To achieve that, we used natural language processing (NLP) to predict each tweet sentiment. We first trained our model using supervised learning with some pre-labelled training data. The model then was used to predict the test data. These results were compared with the actual label to find out the accuracy. Finally, we could draw conclusion from the analysis results made by the model.

2. Exploration

2.1 Dataset Characteristics

The dataset we used is Sentiment140, collected by the Stanford University. There are 1.6 million tweets included in this dataset, collected between April to May in 2009. It contains 6 fields, including “target”, “ids”, “date”, “flag”, “user”, “text”. The label in this dataset is “target”, which stands for the polarity, or the sentiment, of each text. It is classified into two classes: 0 (negative), 4 (positive).

```
1  df.show(n=5)
```

▸ (1) Spark Jobs

```
+------+----------+----+--------+---------------+--------------------+
|target|       ids|date|    flag|           user|                text|
+------+----------+----+--------+---------------+--------------------+
|     0|1467810369|null|NO_QUERY|_TheSpecialOne_|@switchfoot http:...|
|     0|1467810672|null|NO_QUERY|  scotthamilton|is upset that he ...|
|     0|1467810917|null|NO_QUERY|       mattycus|@Kenichan I dived...|
|     0|1467811184|null|NO_QUERY|        ElleCTF|my whole body fee...|
|     0|1467811193|null|NO_QUERY|         Karoli|@nationwideclass ...|
+------+----------+----+--------+---------------+--------------------+
only showing top 5 rows
```

Figure 2.1 Data Characteristic

2.2 Choice of Machine Learning Model

The model we have adopted is the Logistic Regression Model imported from PySpark library. It is chosen as it is a simple but efficient algorithm which can handle high-dimensional feature spaces. It is suitable for handling text data of tweets. Logistic Regression can model the probability of the tweet belonging to sentiment categories (Negative, Neutral or Positive).

Once the features have been extracted, a model can be trained using machine learning. The model is trained on a labelled dataset, which is used to classify tweets as either positive or negative. The model we have adopted is Logistic Regression, which is a linear model for binary classification tasks. The model learns a set of weights for each feature that contribute to the classification decision, then combines these weights with the feature values to produce a score that indicates the probability of the document or sentence belonging to a particular sentiment class [5]. Due to its simplicity and scalability, it can be easily parallelized and distributed using Spark, making it a good choice for handling large volumes of data.

3. Implementation

3.1 Preprocessing

Before training the model, we preprocess the text data to ensure that it is in a format that can be easily understood and processed by the model. We did three steps of preprocessing, including removing special characters and stopwords, converting the text to lowercase, and stemming the words.

```
def preprocessDef(text, stem=False):
    # Remove link,user and special characters
    text = re.sub(TEXT_CLEANING_RE, ' ',
str(text).lower()).strip()
    tokens = []
    for token in text.split():
        if token not in stop_words:
            if stem:
                tokens.append(stemmer.stem(token))
            else:
                tokens.append(token)
    return " ".join(tokens)

spark.udf.register("preprocess", preprocessDef)
preprocess = udf(preprocessDef)
```

Figure 3.1 Preprocessing

3.2 Word2Vec Model

We randomly split the data into the training set and testing set in a ratio of 8:2. By using the training data, we trained a Word2Vec model which can be used to transform each list of words in the text column into word embeddings. The word embeddings capture the semantic meaning of the words. It can later be used as the input to the logistic regression model for sentiment analysis.

```
documents = df_train.select("target", split("text", "
").alias("text"))
w2v_model = Word2Vec(vectorSize=W2V_SIZE, windowSize=W2V_WINDOW,
minCount=W2V_MIN_COUNT, inputCol="text", outputCol="features")
```

Figure 3.2 Word2Vec Model

3.3 Logistic Regression Model

A logistic regression model is used in this project. We made use of the PySpark LogisticRegression class in building the logistic regression model. This model is trained with random training data. It is used to understand the relationship between the Twitter text and the sentiment. The model can be used in Twitter sentiment analysis to predict the sentiment of tweets based on the text content of the tweets.

```python
# Train a logistic regression model
MAX_ITERATIONS = 10
lr = LogisticRegression(featuresCol="features", labelCol="target",
maxIter=MAX_ITERATIONS)
lr_model = lr.fit(train_word2vec_df)
```

Figure 3.3 Logistic Regression Model

4. Evaluation

After applying the model, we evaluate the performance of the model by accuracy, false positive rate, true positive rate, F-score, precision and recall. Table 4.1 shows the performance results.

Metric	**Result**		
	Total	**Label 0 (when target = 0)**	**Label 1 (when target = 4)**
Accuracy	0.7239683372529179	/	/
False Positive Rate	0.2760294485282593	0.2781280825082096	0.2739330287671318
True Positive Rate	0.7239683372529178	0.7260669712328681	0.7218719174917904
F-score	0.723967275072013	0.7260669712328681	0.7218719174917904
Precision	0.7239734990500657	0.7228223589963743	0.7251234245632519
Recall	0.7239683372529178	0.7244410321564836	0.7234940178373799

Table 4.1 Performance Evaluation of the Model

From the metrics shown in Table 4.1, it shows that the modal gets a high accuracy (roughly 0.724) in sentiment analysis.

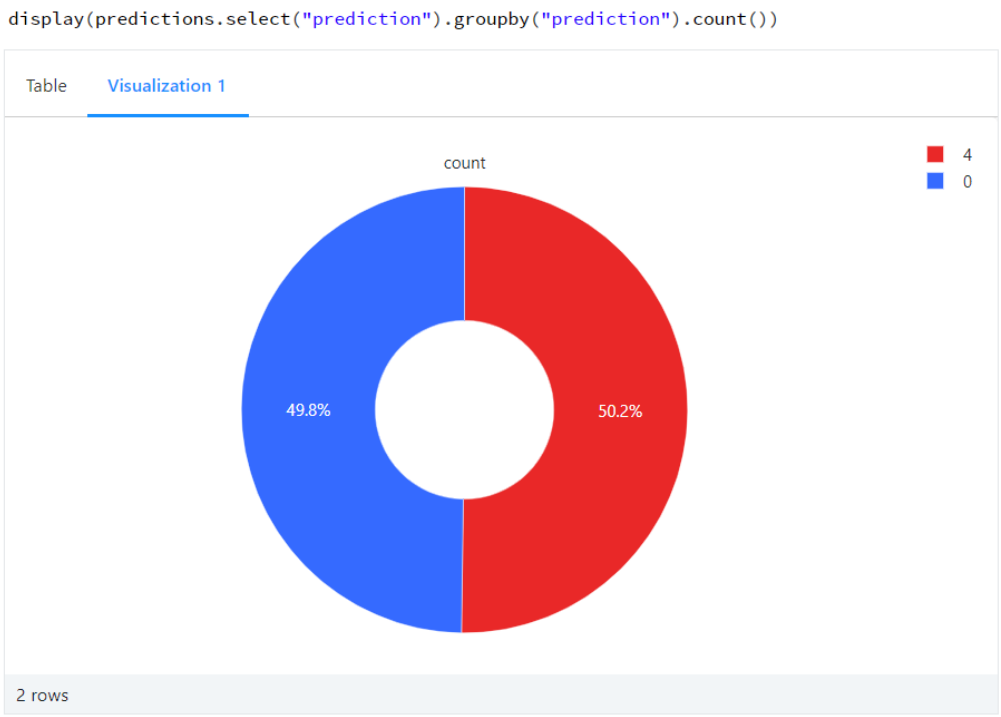

Figure 4.1 Prediction Result Distribution

In Figure 4.1, it shows that 49.8% of the prediction made for the test data is 0 (negative), while 50.2% is 4 (positive). The proportion of positive and negative tweets are roughly similar without a dominant.

5. Conclusion

In conclusion, the analysis of Twitter sentiment using Spark has provided valuable insights into the opinions and emotions expressed by users on the platform. Through the use of various techniques such as text preprocessing, feature extraction, and machine learning algorithms, we were able to accurately classify tweets into positive and negative.

Overall, this analysis has demonstrated the power of using big data tools such as Spark to analyze large volumes of text data and extract meaningful insights. While some individuals believe Twitter is filled with negativity, our result does not align with these claims. These insights can be used by businesses, governments, and individuals to better understand public opinion and sentiment on various topics, and to inform decision-making processes. As the use of social media continues to grow, the analysis of Twitter sentiment is likely to become an increasingly important tool for understanding and shaping public discourse.

6. References

[1] Go, A., Bhayani, R. and Huang, L., 2009. Twitter sentiment classification using distant supervision. CS224N Project Report, Stanford, 1(2009), p.12, https://cs.stanford.edu/people/alecmgo/papers/TwitterDistantSupervision09.pdf

[2] Μαριος Μιχαηλιδης KazAnova, “Sentiment140 dataset with 1.6 million tweets,” Kaggle, https://www.kaggle.com/datasets/kazanova/sentiment140

[3] Apache Software Foundation, "Apache Spark - MLlib - Evaluation Metrics," 2017. Available: https://spark.apache.org/docs/2.2.0/mllib-evaluation-metrics.html

[4] P. Ripamonti, "Twitter Sentiment Analysis," Kaggle, 2020. Available: https://www.kaggle.com/code/paoloripamonti/twitter-sentiment-analysis/notebook

[5] D. Jurafsky and J. H. Martin, “Speech and Language Processing (3rd ed.),” Speech and Language Processing, https://web.stanford.edu/~jurafsky/slp3/