Team 24 DataMinions: Storms 'n' Crypto

Chen Xi	Huang Weiqi	Wang Qiang	Yang Lujia
E1123393	*E1327928*	*E1327918*	*E0983277*
A0084768L	*A0290897H*	*A0290887J*	*A0261883W*

***Abstract*—This report explores the application of text mining methodologies to assess how social media, especially tweets from prominent individuals, influences cryptocurrency valuations. Considering the round-the-clock trading characteristic of cryptocurrencies and their responsiveness to public discourse, this study leverages real-time analysis to support investor decision-making. It employs a dual-model strategy, consisting of a classifier to determine crypto relevance and a sentiment analysis model. The analysis utilized machine learning techniques such as keyword extraction, logistic regression, and LSTM. The findings demonstrate considerable potential for practical application.**

***Index Terms*—Cryptocurrency, Influential Tweets, Sentiment Analysis, Real-time Decision Making, Crypto-relatedness Classifier, keyword extraction, Logistic Regression, LSTM**

I. Introduction

A. Motivation

The rapid ascent of cryptocurrencies in recent years has not only disrupted the traditional financial landscape but also attracted the attention of investors worldwide. With their inherent volatility, cryptocurrencies such as Bitcoin have demonstrated remarkable price fluctuations, often influenced by social events and public discourse. This characteristic volatility, while presenting significant risks, also offers unique opportunities for investors who can navigate through the rapidly changing market landscape effectively.

Given the decentralized nature of cryptocurrencies, they are traded on a 24-hour basis, distinguishing them from traditional financial markets that adhere to specific trading hours. This round-the-clock trading environment ensures that the impact of social events, including tweets from influential figures or celebrities, can have an immediate and pronounced effect on cryptocurrency prices. Such immediacy necessitates real-time analysis and swift decision-making from investors aiming to capitalize on or hedge against these sudden market movements.

Building on the aforementioned facts, this goal of our project is to leverage Natural Language Processing (NLP) techniques for the real-time analysis of social media discourse, particularly tweets from influential personalities and celebrities. By analyzing the sentiment and content of such tweets, the project aims to offer immediate insights into their potential impact on cryptocurrency prices, enabling investors to discern and react to market sentiments with unprecedented speed and precision.

B. Initial Experiments

In the early stages of our project, we embarked on an ambitious endeavor to directly correlate individual tweets with fluctuations in Bitcoin prices. Our methodology involved annotating a dataset from Kaggle [1], which contained 22 million tweets related to cryptocurrency, with subsequent Bitcoin price movements. The goal was to discern whether a given tweet could have a positive or negative impact on Bitcoin prices.

We conducted preprocessing on both numerical and qualitative features to prepare the data for prediction. We categorized the price changes as "positive" if the increase was over 1%, "negative" if the decrease was over 1%, and "no impact" otherwise. Tweet text is cleaned, lemmatized and tokenized to reduce words to their base forms. We also incorporated emojis and hashtags as separate features derived from the raw text, which were converted to cleaned and tokenized texts.

We experimented with the dataset using traditional methods like regression classifiers with `TF-IDF` and deep learning models. While conceptually promising, this approach led us down a complex and ultimately unproductive path.

Challenges Encountered

- Dataset Limitations: The primary dataset was compiled using a series of hashtags (e.g., #crypto, #bitcoin, #cryptocurrency), a method that inadvertently skewed the data towards promotional content. We observed that influential figures in the cryptocurrency space, such as Elon Musk, rarely utilize these hashtags. Their omission is likely due to their already substantial visibility, which negates the need for hashtags to attract additional attention. This realization highlighted a significant flaw in our dataset: it was disproportionately filled with advertisements rather than impactful discourse.
- Mis-attribution of Price Movements: Our initial model aimed to classify tweets as having a positive or negative impact based on the direction of Bitcoin's price movement following the tweet. However, this approach did not account for the inherent volatility of cryptocurrency prices, which can fluctuate independently of social media discourse. Consequently, we found ourselves mislabeling tweets that coincided with price changes as causative, despite the absence of a direct relationship. This misattribution led to a model trained on inaccurately tagged data, undermining its validity.

C. Revised Strategy

Faced with these challenges, we explored various adjustments, such as refining our dataset by the number of followers, in hopes of mitigating the issues. Despite these efforts, the fundamental problems persisted, primarily due to the biased nature of our dataset, which was heavily laden with advertisements and hashtag-driven content.

This realization prompted a strategic pivot toward a dual-model approach. We developed one model to classify whether a tweet is related to cryptocurrency and another to assess the sentiment of the tweet. Instead of relying on the hashtag-filtered tweets, we chose to analyze the broader spectrum of tweets from influential figures. This revised methodology, focusing on the content and sentiment of tweets without the bias of hashtags, proved to be significantly more effective. Our evaluation results confirmed the practicality and relevance of our approach, demonstrating its potential to accurately gauge the impact of social media on cryptocurrency markets.

II. Data Collection/Preparation

Different datasets were used by our two models, in this section will discuss them one by one.

A. Crypto Relatedness Classifier

Our first dataset comprises historical tweets from 101 celebrities (Twitter influentials dataset) known for their influence in the cryptocurrency domain, such as Vitalik Buterin. To construct a representative sample of the discourse surrounding cryptocurrencies on social media, we extracted 500 tweets from each identified influencer, culminating in an initial dataset of 50,500 tweets. Recognizing the necessity of distinguishing between cryptocurrency-related content and unrelated discourse, we employed `ChatGPT` to label each tweet according to its relevance to the cryptocurrency sphere. GPT failed to label some of the tweets and finally, we have a labelled dataset of 37,000 tweets. The final GPT-tagged dataset has the following columns:

- **text**: The content of the tweet. (String)
- **is_crypto_related**: If the tweet is related to crypto. (Boolean)

B. Sentiment Analysis

For our sentiment analysis task, we opted for the Sentiment140 dataset available on Kaggle [2]. This dataset comprises 1,600,000 tweets, gathered via the Twitter API. These tweets are annotated with sentiment labels, evenly distributed between two classes. We chose this dataset over others due to its widespread usage and direct relevance to tweets, ensuring alignment with our analysis objectives. Moreover, its substantial size provides enough data for training a neural network-based sentiment analysis model. Within the dataset, two specific columns out of the total six are relevant to our task:

- **text**: the text of the tweet ("Lyx is cool")
- **target**: the polarity of the tweet (0 = negative, 4 = positive)

With the help of libraries like `langdetect`, we determined that the majority of tweets in our dataset are in English. For the ones tagged as non-English, we manually sampled some and found them to be in English as well. Consequently, we can confidently proceed with the assumption that our subsequent steps, including preprocessing, will predominantly involve handling English text.

III. Data Pre-processing and Feature Engineering

Slightly different approaches were used for our two models, there will be discussed in detail in this section.

A. Crypto Relatedness Classifier

The Twitter influential dataset only has two simple columns, the pre-processing is mainly focused on the text column.

Text Column Pre-Processing: For the text column, we removed URLs and mentions from the tweets to eliminate noise and focus solely on the textual content relevant to our analysis. We didn't do stopwords removal and case folding, because those are necessary for the latter steps like POS tagging and keyword extraction. By maintaining the original case of the text, we ensured that our NLP models could accurately recognize and interpret these entities, thereby enhancing the precision of our analysis.

B. Sentiment Analysis

In the preprocessing phase of Sentiment 140 dataset [2], we employed a comprehensive approach dedicated to each dataset to transform raw text and numerical features into structured data suitable for data mining.

target: We normalized the numerical representation for labels (0 = negative, 1 = positive)

text: The actual text for tweet, is cleaned with the following steps:

- Cleaned tweets using the `tweet-preprocessor` [4] package, including URL removal, hashtag handling, mention elimination (e.g., @username), and reserved word exclusion (e.g., RT for retweets, FAV for favorites).
- Lowercased text to standardize formatting and remove case sensitivity biases.
- Removed punctuation marks and digits using regular expressions to focus on textual content relevant to sentiment analysis.
- Applied lemmatization(using `WordNetLemmatizer` from nltk) to reduce words to their base forms, enhancing text normalization and reducing feature dimensionality.
- Converted emojis and emoticons to words for consistent textual representation across the dataset.
- Tokenized(using `TweetTokenizer` from nltk) the text into individual words or tokens for further processing and feature extraction.

After the preprocessing steps described above, certain text entries become empty. These entries are subsequently removed from the dataset before it is passed to the training models.

Although stopwords(commonly occurring words with little semantic value) removal helps to reduce noise and improve

feature interpretability, we opted not to remove stop words to preserve context information since stopwords express negation like 'not' might have a significant impact on sentiment analysis.

During preprocessing, we carefully considered feature selection and relevance to optimize model performance. We excluded features like "date" to focus on sentiment and thematic features extracted from text, rather than temporal sequences. Additionally, we avoided using "user_name" or "user_id" due to normalization challenges and potential biases, as these features can introduce noise in sentiment analysis tasks. Our emphasis on feature selection aimed to enhance the dataset's quality, ensuring that noise, irrelevant information, and redundant features were minimized.

IV. Data Mining

We pivoted towards a more robust strategy leveraging ensemble techniques and specialized models. Instead of relying solely on a single model for cryptocurrency price trend prediction, we adopted ensemble techniques. This involved training separate models for topic classification (whether crypto-related) and sentiment classification (positive, negative), combining their outputs to enhance trend prediction accuracy.

A. Crypto Relatedness Classifier

In developing a classifier to accurately determine whether a tweet is related to cryptocurrency, our approach was twofold, combining rule-based filtering with a logistic regression model to ensure both precision and scalability.

Rule Based Approach:

- Initially, we implemented a rule-based system that scrutinized each tweet for the presence of specific keywords associated with the cryptocurrency domain, such as "Bitcoin." If any of these predefined keywords were detected within a tweet, the classifier would immediately categorize it as crypto-related. This step served as an efficient filter to quickly identify clear-cut cases of cryptocurrency-related content without the need for more complex processing.
- POS Tagging: Each tweet underwent Part-Of-Speech (POS) tagging to identify the grammatical components, focusing specifically on nouns. Our rule-based classification was particularly attentive to nouns, as they often hold the most significant in indicating the subject matter of a tweet.

For tweets not immediately classified through keyword detection, we employed a more nuanced analysis using a logistic regression model. The preparation for this model involved several preprocessing steps to optimize the text for classification:

Logistic Regression Approach:

- POS Tagging: Similar to rule-based approach, POS tagging was done first to identify the grammatical components.
- Lemmatization: Following POS tagging, we applied lemmatization to the text. This process involved reducing words to their base or dictionary form (lemma), which is crucial to avoid a word's different forms appearing as multiple keywords in the next step.
- Keyword Extraction with YAKE: We then utilized Yet Another Keyword Extractor (YAKE) to identify key terms within the tweets. This tool helped in pinpointing relevant keywords that might not have been explicitly listed in our initial rule-based screening.
- TF-IDF Matrix Creation: With the extracted keywords, we constructed a TF-IDF (Term Frequency-Inverse Document Frequency) matrix.

Why we **only used keywords to construct the TF-IDF matrix** mainly because our dataset is small. We have experimented with using all non-stopwords in the sentences to create TF-IDF matrix for training, although the training accuracy was higher (92% vs 86%), it yielded very poor results in the actual application. We suspect this is due to the regression model overfitting to the training dataset. This will be further discussed in the evaluation section.

The logistic regression model then processed this TF-IDF matrix to classify tweets, outputting a binary result: '0' for tweets not related to cryptocurrency and '1' for those that are. This blend of rule-based and logistic regression methodologies allowed us to create a robust classifier that is both sensitive to explicit mentions of cryptocurrency terms and capable of discerning subtler indicators of crypto-related content.

Figure 1 shows the overall flow of the classifier.

B. Sentimental Analysis

Having learned from our previous failed attempts, we opted to conduct sentiment analysis using the `Sentiment140` dataset [2] for a broader scope of sentiment analysis, rather than focusing solely on crypto-related tweets.

Given that a single tweet can contain multiple sentences, each sentence within the same tweet may express contrasting sentiments. To address this issue, we debated the level of granularity required for sentiment analysis. After examining the `Sentiment140` dataset (refer to Figure 2), we concluded that document-level sentiment analysis would be most suitable. Our analysis indicates that the majority of tweets, regardless of their positive or negative sentiment, consist of fewer than 20 words. The average English sentence length falls between 15–20 words. This suggests that most tweets contain only one sentence, likely expressing a single sentiment. Therefore, document-level sentiment analysis should prove to be efficient.

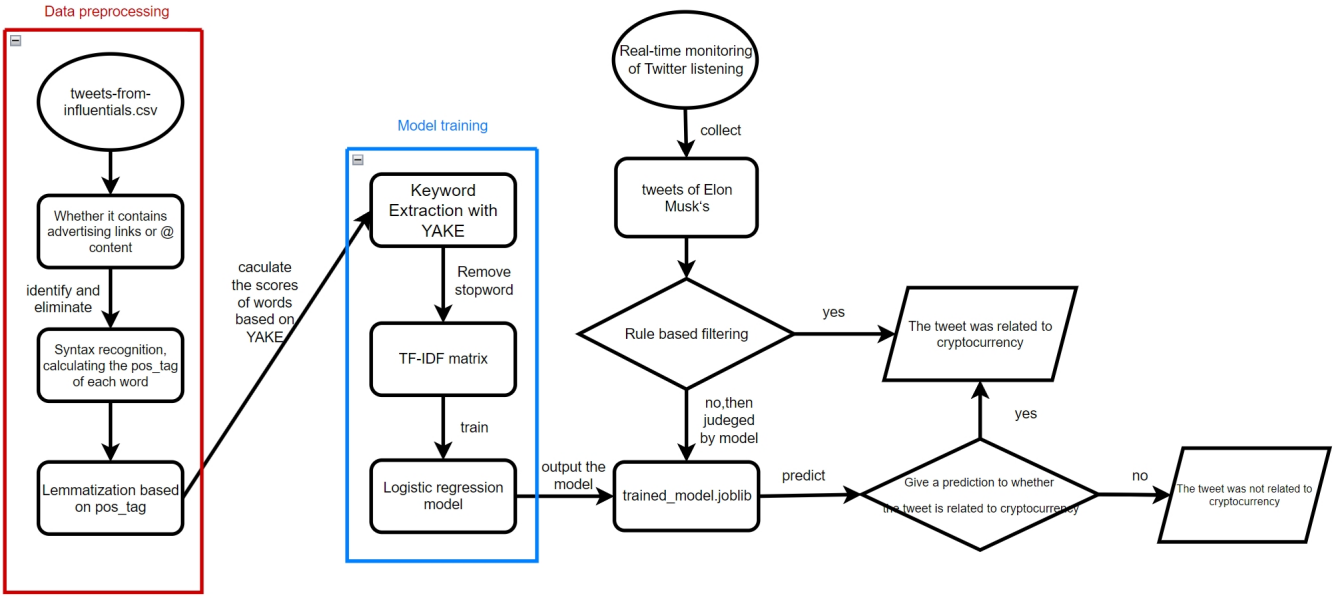

Fig. 1. Crypto-Relatedness Classifier

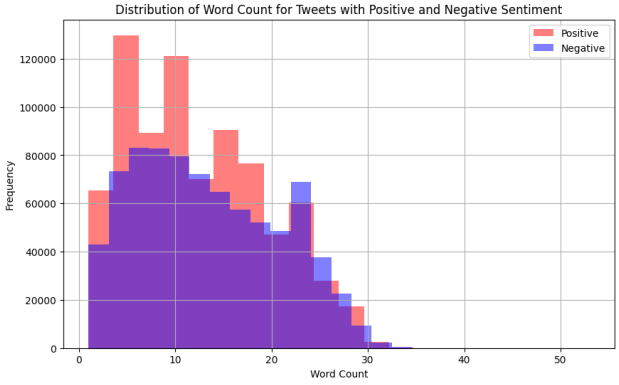

Fig. 2. Word Count Distribution

We explored sentiment analysis using both traditional machine learning and NN-based approaches. Our first step involved dividing the preprocessed data into training and test sets, adhering to an 80:20 split ratio. This partitioning facilitated model training and evaluation, while also providing sufficient validation for performance assessment. To refine our models further, we subdivided the training sets into training and validation sets, following an 80:20 split ratio. This additional step allowed us to leverage the validation set for hyperparameter tuning.

Traditional ML Approach:

We conducted experiments using a Logistic Regression Model, and use TF-IDF (Term Frequency-Inverse Document Frequency) as features. For implementation, we employed the LogisticRegression classifier from sklearn. In the process of hyperparameter tuning, we explored various configurations with the help of `GridSearchCV`, including five different maximum n-gram sizes ranging from 1 to 5, along with two options for the number of features: 20,000 and all features available. Figure 3 and Figure 4 below showcase the F1 Score of the model across various hyperparameter configurations on the same validation dataset.

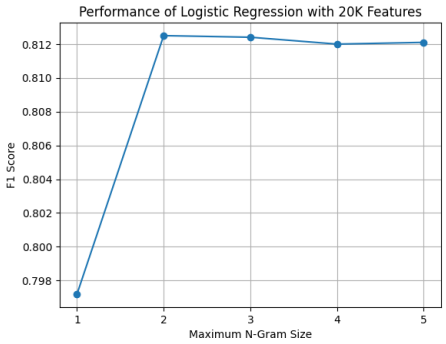

Fig. 3. Logistic Regression Performance with 20K Features

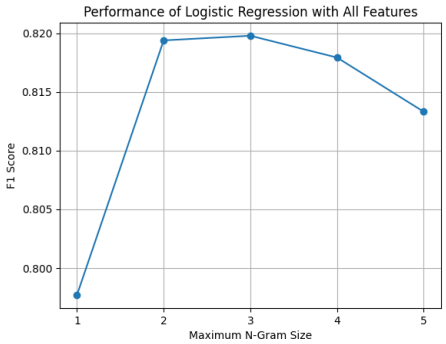

Fig. 4. Logistic Regression Performance with All Features

The Logistic Regression model achieved its highest F1 score of 81.98% on the validation dataset, utilizing the following hyperparameters:

- max_features=NONE,
- ngram_range=(1, 3),
- sublinear_tf=True,
- smooth_idf=True,
- max_iter=300, this ensures that the model has converged effectively.

The model trained with the optimal parameter values achieved an accuracy of 82.01% on the unseen test dataset.

NN-based ML Approach:

- Vocabulary Construction: The vocabulary was constructed from the training data, with special handling for unique and padding tokens. We set a minimum token frequency threshold of 10 to help manage vocabulary size and avoid model training overhead. Text tokens are converted to index vectors, with padding to provide a constant length for all records according to the maximum length. The original length of the tokens is also preserved for pad_packed_sequence() operation in model forwarding.
- Model Architecture: Our sentiment analysis model was based on Long Short-Term Memory (LSTM) networks, known for their ability to capture temporal dependencies in sequential data. Pretrained embeddings from NLPL [3] is utilized to enhance the model's semantic understanding.

```
LSTM(
  (embedding): Embedding(26193, 300,
      padding_idx=1)
  (lstm): LSTM(300, 64, num_layers=2,
      batch_first=True, dropout=0.5,
      bidirectional=True)
  (fc): Linear(in_features=128,
      out_features=2, bias=True)
  (dropout): Dropout(p=0.5, inplace=False)
)
```

Despite challenges such as processing overhead in RNN-based models, we optimized training parameters and employed early stopping techniques to prevent overfitting and reduce computational burdens. The model achieved a test accuracy of 83%, see Figure 5 for an example of training results.

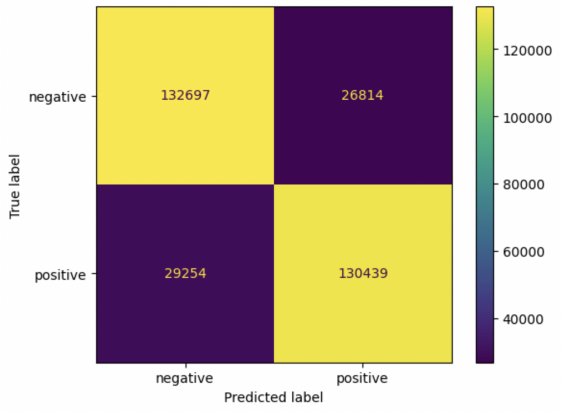

Fig. 5. Confusion Matrix for Sentiment Analysis Model

V. Evaluation and Interpretation

A. Crypto Relatedness Classifier

In this section. we want to discuss and compare the performance two different TF-IDF matrices we used for logistic regression. Specifically, the keywords (YAKE) based TF-IDF and all word-based TF-IDF matrix. We used a twitter dataset of Elon Musk's historical tweets as a verification dataset, which has 5459 tweets and about 110 crypto-related tweets.

TABLE I
Performance comparison of two training approaches

	Tweet Inf. Test Acc.	Elon Musk Tweets Classified as "Crypto"
Keywords TF-IDF	86%	69
Non-stopwords TF-IDF	92%	763

Initially, we trained our logistic regression model using all words from a tweet, excluding stopwords. This approach achieved a test accuracy of 92% on our dataset. However, it incorrectly identified 763 of Elon Musk's tweets as related to cryptocurrency, significantly overestimating the actual figure of just over 100 tweets. This discrepancy highlighted a tendency of the model to erroneously classify many unrelated tweets as being associated with cryptocurrency.

To address this issue, we shifted our strategy to utilize keywords generated by YAKE to construct the TF-IDF matrix for training the logistic regression model. Although this method resulted in a lower training accuracy, its performance in practical applications was markedly improved. For instance, it accurately classified only 69 of Elon Musk's tweets as related to cryptocurrency, closely aligning with the actual count of approximately 110 tweets.

Upon comparing these outcomes, we surmised that the initial overclassification was partly due to the dataset's size (only 37,000 tweets), which led to common, non-cryptocurrency-related words being mislabeled as pertinent to cryptocurrency when they appeared alongside other cryptocurrency-related terms. The implementation of keyword extraction effectively eliminated these "common" words, thereby enhancing the model's accuracy in real-world applications. This adjustment

underscores the importance of refining input data to improve the relevance and precision of predictive models.

B. Sentiment Analysis

TABLE II
PERFORMANCE COMPARISON OF TWO SENTIMENT ANALYSIS APPROACHES

	F1 Score
Logistic Regression with TF-IDF	82.01%
LSTM with NLPL embedding (after 8 epochs)	83.00%

With just 8 epochs, the LSTM approach has already demonstrated superior performance compared to Logistic Regression. This can be attributed to the inherent design of LSTM as a type of recurrent neural network (RNN) tailored to effectively capture sequential dependencies and long-term context in textual data, and tweets frequently consist of word sequences intricately tied to the expressed sentiment. LSTM excels in understanding the nuanced relationships between words, allowing them to discern complex sentiment patterns and sentiments expressed across tweets. In addition, LSTM can automatically learn relevant features from the text, reducing the need for manual feature engineering. However, training LSTM model is pretty time-consuming, which limits our capability to optimize numerous hyperparameters and tune the architecture locally.

Conversely, despite employing trigram features, Logistic Regression struggles to grasp extensive sequential patterns due to its inherent limitations in capturing only a narrow window of sequential information. The model assumes a linear decision boundary, which may not adequately capture nuanced sentiment expressions and patterns. Logistic regression also relies heavily on manual feature engineering, and we did see a significant performance increase after properly processing the text data using domain-specific tool (eg. `tweet-preprocessor` [4]). On the other hand, logistic regression offers advantages in terms of interpretability and computational efficiency, providing a straightforward interpretation of the impact of the features on sentiment classification.

C. Combining Two Models

In this section, we evaluate the effectiveness of our two models in classifying Elon Musk's tweets related to cryptocurrency and determining their sentiment. We researched the Internet and handpicked a series of historical tweets [5] by Elon Musk that significantly influenced the prices of Bitcoin and Dogecoin. By subjecting Musk's tweets to our classification models, we aim to demonstrate the models' ability to identify relevant tweets and gauge their sentiment, thereby providing insights into potential market trends.

The results, as depicted in the table, show that the combined models have successfully classified the relatedness and the sentiment of those historical tweets. The sentiment analysis model has labeled all sentiments correctly. One thing to note is that our crypto-relatedness classifier model also accurately categorized all but one tweet. The exception was a tweet simply stating "ur welcome," which, based on the text alone, does not straightforwardly pertain to cryptocurrency. Upon further examination, we discovered that the tweet included an attached image, a detail our text-based analysis failed to capture. This aspect will be explored in greater depth in the future work section.

TABLE III
CRYPTO RELATEDNESS CLASSIFICATION ON ELON MUSK TWEETS

tweet	Crypto Related	LSTM sentiment label
You can now buy a Tesla with bitcoin	True	Positive
Tesla would trial run accepting DOGE for merchandise	True	Positive
Tesla buys $1.5 billion in bitcoin	True	Positive
Tesla would no longer accept BTC as payment	True	Negative
Tesla will make some merch buyable with Doge & see how it goes	True	Positive
Doge	True	Positive
Ur welcome	False	Positive
Working with Doge devs to improve system transaction efficiency. Potentially promising	True	Positive
Important to support	True	Positive
Tesla has suspended vehicle purchases using Bitcoin..... We are also looking at other cryptocurrencies that use $<$ 1 of Bitcoin's energy/transaction.	True	Negative

VI. FUTURE WORK AND CONCLUSION

A. Future Work

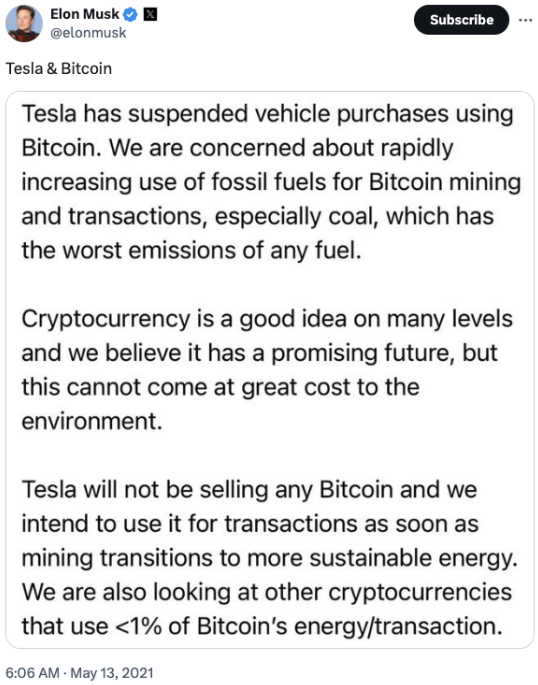

Fig. 6. Tweet with Text as Image

As part of our project's future work, we plan to extend the capabilities of our model to analyze tweets that contain

text within images—a format not currently supported. An illustrative example is a tweet by Elon Musk as shown in Figure 6, which contains significant information affecting Bitcoin's valuation but is not detectable by our current text-only analysis model. This is one of the tweets which has a substantial impact to Bitcoin's price in history.

Fig. 7. Tweet of Image with Significant Indication

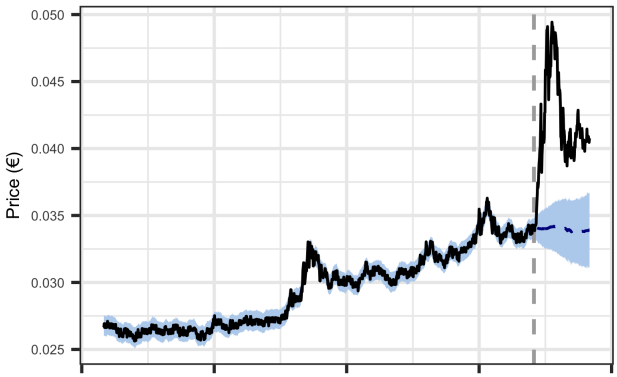

Fig. 8. Dogecoin Price Surge after the Tweet

In Figure 7, the visual component is crucial for understanding the context, as the text accompanying the image is minimal, simply stating "ur welcome". Without visual analysis, the connection to cryptocurrency is not immediately apparent. However, the image depicts a figure presenting a Shiba Inu dog, an unmistakable reference to Dogecoin. This particular tweet had a substantial effect on the cryptocurrency's value, with Dogecoin's price soaring by 50% following the post, as detailed in Figure 8. Accurate interpretation of such tweets necessitates the use of image-to-text conversion methods to extract and analyze the embedded text and visual cues.

B. Conclusion

In this study, we analyzed the impact of social media, particularly tweets from influential individuals, on cryptocurrency valuations using text mining methodologies. By developing a dual-model strategy with a crypto-relatedness classifier and sentiment analysis model, we provided insights for investor decision-making in the cryptocurrency market.

Our refined approach, incorporating a rule-based filtering and logistic regression model in the crypto-relatedness classifier, improved the accuracy of classifying tweets related to cryptocurrency. Similarly, our sentiment analysis models, including logistic regression and LSTM, showed promising results in discerning sentiment from tweets.

Combining the outputs of both models, we successfully classified historical tweets by Elon Musk related to cryptocurrency and determined their sentiment, offering valuable insights into potential market trends.

Looking ahead, our future work explores enhancing the model's capabilities by analyzing tweets with text within images to improve accuracy and provide more context for analysis. While our approach is not perfect, our study demonstrates the potential of text mining and machine learning techniques in real-time decision-making in the cryptocurrency market.

REFERENCES

[1] Bitcoin tweets. (2023, March 10). Kaggle. https://www.kaggle.com/datasets/kaushiksuresh147/bitcoin-tweets.

[2] Sentiment140 dataset with 1.6 million tweets. (2017, September 13). Kaggle. https://www.kaggle.com/datasets/kazanova/sentiment140/data.

[3] NLPL word embeddings repository. (n.d.). http://vectors.nlpl.eu/repository/.

[4] S. (n.d.). GitHub - s/preprocessor: Elegant and Easy Tweet Preprocessing in Python. GitHub. https://github.com/s/preprocessor.

[5] Jain, S., Johari, S., & Delhibabu, R. (2023). Analyzing Cryptocurrency trends using Tweet Sentiment Data and User Meta-Data. arXiv. https://arxiv.org/abs/2307.15956.

[6] Dablander, F. (n.d.). Causal effect of Elon Musk tweets on Dogecoin price. Retrieved [Insert retrieval date here], from https://fabiandablander.com/r/Causal-Doge.html.

VII. APPENDIX

A. Workload Breakdown

TABLE IV
WORKLOAD BREAKDOWN

Team Member	**Main Responsibilities**
Chen Xi	sentiment analysis using logistic regression model, elon musk tweet reasearch, sentiment140 dataset analysis, reports
Huang Weiqi	tweets data preprocessing, programming of crypto relatedness classifier architecture, model designing/training and testing,
Wang Qiang	data collection (tweets, crypto price data), data prepration and tagging (crypto price impact tagging + gpt relatedness tagging), classification model design
Yang Lujia	experiment on Bitcoin price prediction, sentiment analysis using NN models, progress/ final report crafting

B. Source Code

You can find the source code for this study on GitHub at https://github.com/kotorinsky/storms-n-crypto