LONDON'S GLOBAL UNIVERSITY

Social Media Sentiment Analysis on Cryptocurrencies

David Nentwig[1]

BSc Computer Science

Paolo Tasca

09/05/2024

[1]**Disclaimer:** This report is submitted as part requirement for the BSc Computer Science at UCL. It is substantially the result of my own work except where explicitly indicated in the text. *Either:* The report may be freely copied and distributed provided the source is explicitly acknowledged
Or:
The report will be distributed to the internal and external examiners, but thereafter may not be copied or distributed except with permission from the author.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisors, Paolo Tasca, Kamil Tylinski and Darwin Jahja, for their invaluable guidance, support, and encouragement throughout this project. Their expertise, insights, and dedication have been instrumental in shaping my research and ensuring that I stay on track to produce the best possible work. I am incredibly fortunate to have had the opportunity to learn from and work with such knowledgeable and supportive mentors.

I would also like to extend my heartfelt thanks to my family and friends for their unwavering support and encouragement throughout my studies.

Abstract

Cryptocurrency has experienced significant growth and attention in recent years, with sentiment and topic trends playing a crucial role in understanding market dynamics and community discussions. This project focuses on analysing sentiment and topic trends within the cryptocurrency community using data collected from the Mastodon social media platform.

The project employs advanced data analysis techniques, including sentiment analysis and topic modeling, to investigate the relationship between public sentiment and market trends, as well as to identify the dominant topics and themes discussed within the community. The data collection process involves scraping relevant posts from Mastodon using custom scripts, while sentiment analysis is performed using the VADER (Valence Aware Dictionary and sEntiment Reasoner) model. Topic modeling is conducted using Latent Dirichlet Allocation (LDA) to extract prevalent topics from the collected data. The results of the sentiment analysis reveal notable correlations between sentiment and market capitalisation for several major cryptocurrencies, such as BNB, BTC, ETH, MATIC, XMR, and ICP. The topic modeling analysis uncovers the evolving nature of discussions within the cryptocurrency community, with topics such as Bitcoin, Ethereum, and other cryptocurrency-focused themes exhibiting varying levels of engagement over time.

The insights gained from this project contribute to a deeper understanding of the complex dynamics within the cryptocurrency market and community. The findings highlight the potential of sentiment analysis as a valuable tool for gauging market sentiment and predicting trends, while topic modeling provides a comprehensive view of the shifting interests and concerns of the cryptocurrency community. This project lays the foundation for further research and analysis in the field of cryptocurrency sentiment analysis and topic modeling, offering meaningful insights for researchers, investors, and enthusiasts seeking to navigate the rapidly evolving landscape of cryptocurrencies.

Furthermore, this project explores the potential of fine-tuning a large language model, GPT-neo-x-20B, on the collected cryptocurrency-related social media data. By leveraging advanced techniques such as 4-bit quantisation and parameter-efficient fine-tuning, the model is trained to generate relevant responses to questions about the cryptocurrency domain, demonstrating the feasibility of extracting valuable insights from the data using state-of-the-art natural language processing approaches.

Contents

Chapter 1

Introduction

Cryptocurrency has emerged as a transformative financial technology, attracting significant attention from investors, researchers, and the general public. As the cryptocurrency market continues to evolve and mature, understanding the sentiment and prevalent topics within the community becomes increasingly important for making informed decisions and gaining valuable insights.

This project aims to explore the sentiment and topic trends within the cryptocurrency community by analysing data from the Mastodon social media platform. By leveraging advanced data analysis techniques, such as sentiment analysis and topic modeling, we seek to uncover the relationship between public sentiment and market dynamics, as well as identify the dominant themes and discussions surrounding various cryptocurrencies.

The report is structured as follows: Chapter 2 provides the context and background of the project, including an overview of cryptocurrency and blockchain technology, the role of sentiment analysis in financial markets, the significance of topic modeling in cryptocurrency analysis and the potential of using large language models. Chapter 3 outlines the project requirements and analysis, detailing the project plan, goals, constraints, and challenges. Chapter 4 describes the design and implementation of the data collection, sentiment analysis, and topic modeling components. Chapter 5 covers the model training and inference aspects of the project, addressing the model architecture, dataset preparation, training process, and model saving and loading. Chapter 6 presents the results and discussion, including an empirical analysis of the correlation between sentiment and market trends, an in-depth examination of the topic modeling outcomes as well as analysis of model outputs. Finally, Chapter 7 concludes the report, summarising the key findings, implications, limitations and potential future work. The appendices include a user manual for running the data collection, analysis scripts and model training and use, code snippets, and additional figures.

Through this project, we aim to contribute to the growing body of research on cryptocurrency sentiment analysis and topic modeling, providing useful insights for researchers, investors, and enthusiasts in the field.

Chapter 2

Context

2.1 Background

2.1.1 Cryptocurrency and Blockchain Technology

In the recent years, cryptocurrencies have become a major financial innovation, with Bitcoin being the first and most well-known example [1]. These digital or virtual currencies rely on cryptographic techniques to secure and verify transactions, operating independently of central banks [2]. Cryptocurrencies are built upon blockchain technology, a decentralised and distributed ledger system that records transactions across a network of computers [3]. The rise of cryptocurrencies has been fueled by various factors, including the desire for financial autonomy, the need for faster and cheaper cross-border transactions, and the potential for greater financial inclusion [4].As of 2023, there were over 9,000 cryptocurrencies in existence[5], with a total market capitalisation exceeding $1.68 trillion[6] . This rapid growth has attracted significant attention from investors, regulators, and the general public.

2.1.2 Sentiment Analysis in Financial Markets

Sentiment analysis, also known as opinion mining, is the process of computationally identifying and categorising opinions expressed in a piece of text [7]. In the context of financial markets, sentiment analysis is used to gauge market sentiment by analysing news articles, social media posts, and other forms of textual data [8]. The goal is to extract subjective information and determine whether the sentiment towards a particular financial asset or market is positive, negative, or neutral. Numerous studies have explored the relationship between market sentiment and financial market dynamics. Bollen et al. [9] demonstrated that public mood, as measured by Twitter sentiment, can be used to predict stock market movements. Similarly, Li et al. [10] found that sentiment extracted from financial news articles can effectively predict stock price volatility. These findings highlight the potential of sentiment analysis as a valuable tool for investors and financial analysts.

2.1.3 Topic Modeling in Cryptocurrency Analysis

Topic modeling is a machine learning technique that automatically discovers the hidden themes or topics within a large collection of documents [11]. It helps in understanding the main ideas discussed across a corpus of text data, without the need for manual annotation. Latent Dirichlet Allocation (LDA) is one of the most widely used topic modeling algorithms [12]. In the realm of cryptocurrency analysis, topic modeling has been applied to identify the key topics of discussion within cryptocurrency communities. For example, Kim et al. [13] used LDA to analyse user comments and replies on cryptocurrency forums, uncovering topics such as market trends, blockchain technology, and investment strategies. Topic modeling can provide valuable insights into the evolving narratives and concerns surrounding cryptocurrencies, helping stakeholders stay informed about market sentiments and emerging trends.

2.1.4 Large Language Models and Fine-tuning

Large language models, such as GPT-neo-x-20B, have demonstrated remarkable performance in various natural language processing tasks [15]. These models are pre-trained on vast amounts of text data and can be fine-tuned on domain-specific datasets to adapt to specific tasks [16]. Fine-tuning large language models on cryptocurrency-related social media data has the potential to complement sentiment analysis and topic modeling approaches in understanding the nuances and dynamics of the cryptocurrency discourse.

2.1.5 DLT Science and Cryptocurrency Analysis

DLT Science is a data science company specialising in the analysis of distributed ledger technologies and cryptocurrencies [14]. Their mission is to provide data-driven insights and solutions to help individuals, businesses, and institutions navigate the complex landscape of cryptocurrencies and blockchain technology. By leveraging advanced data science techniques, including sentiment analysis and topic modeling, DLT Science aims to extract meaningful insights from vast amounts of cryptocurrency-related data. These insights can aid in understanding market trends, investor behavior, and the overall sentiment surrounding specific cryptocurrencies. Through their work, DLT Science seeks to contribute to the development and adoption of cryptocurrencies and blockchain technology, while also providing valuable intelligence to stakeholders in the industry.

2.2 Literature Review

2.2.1 Sentiment Analysis

Sentiment analysis in social media has become an essential tool for understanding public opinions, emotions, and trends. It involves the computational study of people's opinions, sentiments, attitudes, and emotions expressed in written language [7]. Various sentiment analysis techniques have been applied to diverse domains, including finance, where market sentiment can significantly influence stock prices and trading volumes [9]. In the context of cryptocurrencies, sentiment analysis can provide insights into the market's emotional response to different currencies, potentially predicting price movements based on public sentiment [13].

2.2.2 Topic Modeling

Topic modeling is a type of statistical model for discovering the abstract "topics" that occur in a collection of documents. Latent Dirichlet Allocation (LDA) is one of the most popular topic modeling techniques, which assumes that each document is a mixture of a small number of topics and that each word's presence is attributable to one of the document's topics [12]. In social media analysis, topic modeling helps identify prevalent themes or subjects in users' posts, aiding in the categorisation and summarisation of large datasets [17].

2.2.3 Data Pipeline Approaches

Data pipelines are crucial for efficiently processing and analysing large volumes of data. Real-time data streaming is particularly relevant for social media data, enabling immediate analysis and response to emerging trends. Apache Kafka and Apache Flink are notable technologies that facilitate real-time data processing, offering robust, scalable solutions for streaming data pipelines [18][19].

2.2.4 Temporal Analysis in Sentiment and Market Trend Correlation

Temporal analysis is a critical component in the study of market trends, particularly in the context of sentiment analysis. It involves examining and interpreting data over time to identify patterns, trends, and potential causative relationships. In financial markets, and specifically in the domain of cryptocurrencies, temporal analysis aids in understanding how market sentiments expressed on social media platforms correlate with fluctuations in market prices and trends. For instance, Bollen, Mao, and Zeng (2011)[9] demonstrated that Twitter mood could predict the stock market, highlighting the value of temporal analysis in correlating sentiment with market dynamics.

The inclusion of temporal analysis in sentiment assessment allows researchers to track sentiment changes over time, providing insights into how public perception evolves in response to market events or external stimuli [20]. For instance, in cryptocurrency markets, a sudden spike in negative sentiment on social media could precede a drop in market prices, offering predictive insights if analysed correctly.

2.2.5 Libraries

TextBlob

TextBlob is a Python library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more. The sentiment analysis feature of TextBlob is particularly noteworthy, as it utilises a pre-trained classifier, making it accessible for rapid prototyping and analysis [21].

Plotly

Plotly is a multi-language data visualisation library that enables interactive, publication-quality graphs online. It supports over 40 unique chart types, encompassing a wide range of statistical, financial, geographic, scientific, and 3D visualisations. Plotly is integrated with analytical and web-based programming languages and frameworks, making it a versatile tool for creating comprehensive visual representations of data [22].

Asynchronous Data Collection

Asynchronous data collection, particularly in Python's asyncio library, allows for efficient data fetching and processing without blocking the execution flow. This approach is beneficial when dealing with high-volume data sources like social media, where simultaneous requests and non-blocking operations can significantly enhance performance and responsiveness [23].

VADER Sentiment Analysis

The VADER (Valence Aware Dictionary and sEntiment Reasoner) library is a lexicon and rule-based sentiment analysis tool specifically attuned to sentiments expressed in social media. Developed by Hutto and Gilbert (2014)[24], VADER is effective in handling text with a mix of polarity (positive/negative), intensity (strength of emotion), and context, including idiomatic expressions, slang, and emoji. It evaluates text to return a compound score that aggregates the cumulative sentiment conveyed, making it particularly useful for real-time sentiment monitoring. Its high accuracy and ease of use, without the need for training on labeled data, make VADER an excellent choice for projects that require rapid and robust sentiment analysis, particularly in the context of social media content.

2.2.6 Recent Advancements in LLM's and their Applications

Recent advancements in large language models have showcased their effectiveness in various domain-specific tasks. The use of parameter-efficient fine-tuning (PEFT) techniques, such as LoRA (Low-Rank Adaptation) [25], has enabled efficient adaptation of these models to specific domains while reducing the memory footprint. Additionally, quantisation techniques, such as 4-bit quantisation [26], have further reduced the computational requirements, making fine-tuning more accessible and democratising the use of large language models.

Chapter 3

Requirements and Analysis

3.1 Project Plan: Cryptocurrency Sentiment and Topic Analysis on Mastodon

3.1.1 Objective

The objective of this project is to develop a system that collects data from the Mastodon social media platform, conducts sentiment analysis and topic modeling on the discourse surrounding various cryptocurrencies, and visualises the results to uncover insights about public perception and the prevalence of different topics within the cryptocurrency community.

3.1.2 Goals

Data Collection:

- Brief: Extract textual data from Mastodon posts (toots) related to cryptocurrency discussions.
- How: Utilise the Mastodon API to extract relevant posts using specified tags or keywords related to cryptocurrencies. Develop a custom data collection script to automate the process.

Sentiment Analysis:

- Brief: Assess the general sentiment (positive, negative, neutral) towards different cryptocurrencies from the collected data.
- How: Implement sentiment analysis techniques using NLP libraries (such as NLTK or VADER) and custom algorithms to calculate sentiment scores for each post.

Topic Modeling:

- Brief: Identify the main topics and themes discussed within the cryptocurrency-related posts on Mastodon.

- How: Apply topic modeling techniques, such as Latent Dirichlet Allocation (LDA), to discover latent topics in the collected data. Utilise libraries like scikit-learn for implementation.

Fine-tuning a Large Language Model:

- Brief: Explore the feasibility of fine-tuning GPT-neo-x-20B on the collected cryptocurrency-related social media data.
- How: Utilise advanced techniques such as 4-bit quantisation [26] and parameter-efficient fine-tuning (PEFT) [25] to train the model to generate relevant responses to questions about the cryptocurrency domain.
- Goal: Demonstrate the potential of leveraging state-of-the-art natural language processing approaches to extract valuable insights from the collected data.

Data Visualisation:

- Brief: Visually represent the sentiment distribution and topic prevalence related to various cryptocurrencies over time.
- How: Create informative and interactive visualisations using libraries like Matplotlib, Seaborn, or Plotly in Python. Display sentiment trends and topic evolution across different time periods.

Scope and Constraints

- The project will focus on analysing sentiment and topics related to the top 50 cryptocurrencies based on market capitalisation.
- The data collection will be limited to English-language posts on Mastodon.
- The project will not involve real-time data collection and analysis due to the limitations of the Mastodon API and the processing requirements.

Data Preprocessing

- Perform data cleaning and preprocessing techniques to remove noise, irrelevant information, and handle missing data.
- Implement NLP techniques such as tokenisation, stemming, and stop-word removal to prepare the data for sentiment analysis and topic modeling.

Model Evaluation and Validation

- Evaluate the performance of the sentiment analysis and topic modeling algorithms using appropriate metrics and validation techniques.
- Assess the performance of the fine-tuned GPT-neo-x-20B model in generating relevant and coherent responses to cryptocurrency-related questions.

- Conduct manual review of a sample of the results to assess the accuracy and coherence of the sentiment labels, identified topics, and generated responses.
- Compare the insights derived from the fine-tuned language model with those obtained from sentiment analysis and topic modeling to validate the consistency and complementarity of the different approaches.

Challenges and Considerations

- Address the limited availability of data for less popular cryptocurrencies and implement strategies to handle data imbalance.
- Develop mechanisms to filter out bot-generated posts and spam to improve data quality.
- Ensure compliance with Mastodon's API terms of service and respect user privacy throughout the data collection and analysis process.
- Address the computational requirements and resource constraints associated with fine-tuning a large language model like GPT-neo-x-20B.
- Explore the potential of integrating the fine-tuned language model with the sentiment analysis and topic modeling components to provide a comprehensive understanding of the cryptocurrency community's sentiments and discussions.

3.2 Constraints and Challenges

3.2.1 Limited Data Availability

One of the main constraints encountered during the project was the limited availability of posts related to some of the top 50 coins based on market capitalisation. While popular coins like Ethereum (ETH) and Bitcoin (BTC) had a substantial number of posts, other coins had comparatively fewer mentions.

3.2.2 Presence of Bot-Generated Posts

The presence of bot-generated posts on Mastodon posed a challenge, as these posts were not suitable for sentiment analysis. To address this issue, a trust score check was implemented to filter out bot posts, which will be discussed in detail later in the report.

3.3 Data Collection

3.3.1 Data Source

The primary data source for this project was Mastodon, a decentralised social media platform. A custom data collection script was developed to scrape posts containing specific hashtags related to cryptocurrencies. The script included preprocessing steps to clean and prepare the data for analysis. The code for the data collection process can be found in the accompanying script.

3.3.2 Exploratory Data Analysis

To gain a better understanding of the cryptocurrency discourse on Mastodon, exploratory data analysis was conducted. Scripts were created to identify the most popular hashtags within the crypto community by analysing posts tagged with #crypto. This analysis also revealed the significant presence of bot users, highlighting the need for filtering mechanisms. Additionally, a script was used to determine the number of posts collected per hashtag within a specific timeframe, helping to estimate the required data collection duration to obtain a sufficiently large dataset for meaningful insights.

3.4 Challenges and Considerations

3.4.1 Multi-Platform Data Collection

During the project, several challenges and considerations were identified. Initially, the project aimed to collect data from multiple platforms, including Mastodon, GETTR, and Discord. However, upon analysing the GETTR data, it was discovered that a substantial portion of the posts were generated by bot accounts or contained spam and irrelevant content, rendering the data unsuitable for analysis.

3.4.2 Ethical Constraints

Collecting data from Discord servers was found to be against the platform's terms of service, preventing the inclusion of Discord data in the project. While manual exploration of crypto-related Discord servers suggested the presence of valuable data, ethical constraints prevented its collection and utilisation.

3.5 Outcome

Despite these challenges, the project successfully obtained a sufficient volume of high-quality data from Mastodon to perform sentiment analysis and topic modeling. The data collection process was optimised to ensure the dataset was representative and suitable for deriving meaningful insights into the cryptocurrency community's sentiments and discussions.

Chapter 4

Design and Implementation

4.1 Data Collection

4.1.1 Mastodon Data Collection (`mastodonScraper.py`)

To gather data for sentiment analysis and topic modeling, we collected posts from Mastodon, a decentralised social media platform [33]. The `mastodonScraper.py` script was developed to fetch posts containing specific hashtags related to cryptocurrencies within a given time duration.

The `fetch_toots` function in the script handles rate limiting and error scenarios using adaptive retry intervals. It checks for specific error codes, such as 429 (Too Many Requests), and retries the request after a calculated interval based on the error response [34]. This ensures that the script adheres to the rate limits imposed by the Mastodon API and gracefully handles any temporary server issues.

The `fetch_hashtag_toots` function is responsible for fetching posts that contain the specified hashtags. It makes use of the `fetch_toots` function to retrieve the posts and applies additional filtering to remove posts from bots and clean the HTML content. The script utilises asynchronous programming with `aiohttp` and `asyncio` to make concurrent requests, improving the overall performance of the data collection process [35, 36].

The collected posts are saved as JSON files in the `mastodon_posts_top50` folder, which serves as the dataset for further analysis. The JSON files are named according to the corresponding hashtag, allowing for easy organisation and retrieval of the collected data. While the script initially focused on collecting posts based on the top cryptocurrency hashtags found on Mastodon (as described in the next subsection), the analysis primarily concentrated on the top 50 coins based on market capitalisation data from Yahoo Finance [37]. This approach enables a more comprehensive examination of the relationship between sentiment changes and market cap fluctuations for the most prominent cryptocurrencies.

4.1.2 Hashtag Analysis (hashTagFinder.py)

To identify the most frequently used cryptocurrency hashtags on Mastodon, we developed the `hashTagFinder.py` script. This script analyses the collected Mastodon data and determines the prevalence of various hashtags within the posts.

The `find_hashtags_in_text` function utilises regular expressions to extract hashtags from the content of each post [38]. It updates a frequency dictionary, incrementing the count for each hashtag found. The `find_hashtags_in_json` function iterates over the JSON files in the specified directory (`mastodon_posts`) and applies the `find_hashtags_in_text` function to each post's content.

After processing all the JSON files, the `save_hashtags_to_csv` function is used to save the found hashtags and their corresponding frequencies to a CSV file named `hashtags.csv`. The hashtags are sorted in descending order based on their frequency, providing an overview of the most popular cryptocurrency-related hashtags on Mastodon.

Although the hashtag analysis provided insights into the prevalent topics of discussion on Mastodon, the primary focus of the sentiment analysis was on the top 50 coins based on market capitalisation. This approach allows for a more targeted examination of the relationship between sentiment and market performance for the most significant cryptocurrencies.

However, for the aggregate sentiment analysis over time, the entire dataset, including posts collected based on the top Mastodon cryptocurrency hashtags, was utilised. This comprehensive dataset provides a broader perspective on the overall sentiment trends within the cryptocurrency community on Mastodon.

By collecting data from Mastodon and analysing the top cryptocurrency hashtags, we established a robust dataset for sentiment analysis and topic modeling. The focused analysis on the top 50 coins based on market capitalisation enables a deeper understanding of the interplay between sentiment and market dynamics for the most prominent cryptocurrencies.

4.2 Sentiment Analysis and Topic Modeling

4.2.1 Introduction

Sentiment analysis and topic modeling have emerged as powerful techniques for understanding market dynamics and investor sentiment in the cryptocurrency domain. This report aims to leverage these techniques to extract valuable insights from a textual dataset, offering a deeper understanding of the sentiment patterns and thematic structures within the cryptocurrency community. The findings of this analysis have the potential to inform investment decisions, market forecasting, and overall comprehension of the cryptocurrency market landscape.[39] [40]

4.2.2 Methodology

The analysis code employs the VADER (Valence Aware Dictionary and sEntiment Reasoner) [24] model for sentiment analysis and Latent Dirichlet Allocation (LDA) [12] for topic modeling. VADER is a lexicon and rule-based sentiment analysis tool specifically attuned to sentiments expressed in social media, making it suitable for analysing cryptocurrency-related discussions. LDA, on the other hand, is a probabilistic topic modeling technique that discovers latent topics in a collection of documents, which allows for the uncovering of the main themes and topics within the cryptocurrency community.

The data preprocessing steps, including text cleaning, tokenisation, and removal of stop words, are crucial in preparing the data for analysis. These steps ensure that the text is standardised and free from noise, allowing for more accurate sentiment scores and topic extraction.

Sentiment Analysis

VADER calculates sentiment scores by assigning a sentiment intensity to each word in the text, ranging from -4 (extremely negative) to +4 (extremely positive). The compound score [24], which is the normalised sum of the sentiment scores of each word in the text, provides a unified measure of sentiment. The compound score ranges from -1 (most extreme negative) to +1 (most extreme positive), with 0 indicating a neutral sentiment.

Mathematically, the compound score is calculated as follows:

$$\text{Compound Score} = \frac{\sum_{i=1}^{n} \text{sentiment\_score}(w_i)}{n} \tag{4.1}$$

where w_i represents each word in the text, $\text{sentiment\_score}(w_i)$ is the sentiment intensity of word w_i, and n is the total number of words in the text.

In addition to the sentiment score, we introduced a trust score [41] to assess the reliability of the sentiment expressed in each post. The trust score takes into account factors such as user engagement, follower-to-following ratio, and post repetition. It is calculated using the following formula:

$$\text{Trust Score} = \left(\frac{\text{Engagement Index} \times 0.7 + \log(1 + \min(\frac{\text{Followers}}{\text{Following}}, 10)) \times 30}{100} \right) \times \text{Penalty} \tag{4.2}$$

where the Engagement Index is a normalised measure of user engagement based on replies, reblogs, and favorites; the Followers/Following Ratio is capped at 10 to mitigate the impact of outliers; and the Penalty is a factor applied to accounts with repetitive posts to reduce the influence of potential spam.

The sentiment scores and trust scores provide a quantitative measure of sentiment expressed in the cryptocurrency-related discussions, which enables analysis of sentiment trends over time and across different cryptocurrencies.

Sentiment analysis is conducted through two scripts:

- Aggregated Sentiment Over Time (AggrSentOverTime.py): This script aggregates sentiment scores over specified time intervals, calculating average sentiment values to reveal temporal sentiment trends within the dataset.

- Comparative Sentiment Analysis (allCoinSentGraph.py): This script generates comparative visualisations of sentiment across different categories or entities, highlighting how sentiment varies over time among various subjects.

Aggregated Sentiment Over Time with `AggrSentOverTime.py`

The `AggrSentOverTime.py` script plays a pivotal role in the sentiment analysis component of the project. It processes the textual data to extract sentiment scores over time, facilitating a temporal analysis of sentiment trends.

Data Loading and Sentiment Analysis

The script begins by loading the dataset, iterating through each JSON entry using the `json` library. It extracts essential information such as the `created_at` timestamp and the `content` for sentiment analysis. The `SentimentIntensityAnalyzer` from the `vaderSentiment` library is utilised to calculate the sentiment scores, particularly focusing on the compound score which reflects the overall sentiment of the text. In addition to sentiment scoring, a trust score is calculated for each post to assess its reliability based on engagement metrics and the account's followers-to-following ratio. *Listing 1 demonstrates the calculation of sentiment and trust scores using the VADER SentimentIntensityAnalyzer.*

Time-based Aggregation

The script aggregates sentiment scores by the `created_at` timestamp, categorising the data into specific time intervals (e.g., days, weeks or months). This organisation allows for the computation of average or cumulative sentiment scores for each interval. *Listing 2 shows the aggregation logic used to categorise sentiment scores based on the `created_at` timestamp.*

Smoothing and Clipping Sentiment Data

To mitigate the impact of extreme sentiment values and reduce noise in the data, the script applies smoothing and clipping techniques. A simple moving average (SMA) is used to smooth the sentiment scores over a specified window size. Additionally, sentiment scores are clipped based on quantiles to limit the influence of outliers.

Resampling Data

The script resamples the sentiment data to a monthly frequency using the `resample('M')` function from pandas. This step helps in reducing the number of data points and smoothing out short-term fluctuations, providing a clearer view of the overall sentiment trend.

Visualisation of Sentiment Trends

Finally, the script employs `plotly` to visualise the sentiment trends over time, creating a time series graph that demonstrates the evolution of sentiment. The graph includes both the actual sentiment data points and a trend line to highlight the overall sentiment trajectory. *Listing 3 demonstrates the visualisation of sentiment trends using the `plotly` library.* This section of the analysis provides invaluable insights into how the sentiment associated with the dataset's subject matter evolves, offering a nuanced understanding of public perception or sentiment trends over time.

Comparative Sentiment and Market Analysis with `allCoinSentGraph.py`

The `allCoinSentGraph.py` script is instrumental in correlating sentiment data with market performance across different cryptocurrencies. It provides a dual perspective by overlaying sentiment trends with market data to discern potential correlations or patterns.

Data Loading and Sentiment Analysis

Initially, the script parses through JSON files to extract cryptocurrency-related posts, focusing on their `content` and `created_at` fields for sentiment and time series analysis, respectively. The sentiment scores are calculated using the VADER SentimentIntensityAnalyzer, emphasising the compound score for a holistic sentiment perspective.

Market Data Integration

To complement sentiment analysis, the script fetches market data for corresponding cryptocurrencies using `yfinance`. It aligns this market data with the sentiment data temporally, ensuring a coherent and synchronised comparative analysis.

Handling Specific Cases (ICP)

For the cryptocurrency ICP, the script incorporates a specific adjustment to handle the large drop in market capitalisation that affects the scale of the graph. By setting a specific start date for ICP, the script focuses on the period after the significant market cap drop, allowing for a more meaningful comparison between sentiment and price.

Visualisation of Sentiment and Market Trends

Using `plotly`, the script generates a graph that concurrently displays sentiment scores and market data over time. This visualisation facilitates an intuitive understanding of how public sentiment may align or diverge from market trends.

Listing 4 illustrates the process of fetching market data using the `yfinance` library.

Listing 5 demonstrates the alignment of sentiment and market data and their visualisation using the `plotly` library.

Analysis

Through the `allCoinSentGraph.py` script, we gain a multifaceted view of the interplay between public sentiment and market dynamics in the cryptocurrency sector. This analysis aids in uncovering patterns that may forecast market movements, demonstrating the script's utility in providing predictive insights and a more comprehensive market understanding.

Topic Extraction

Topic extraction is executed through a series of scripts, each addressing a unique facet of the topic modeling process:

- LDA Topic Modeling (ldaProcessing.py): This script is the cornerstone of topic extraction, applying Latent Dirichlet Allocation to distill the primary themes from the dataset. The journey to this script's final form involved overcoming challenges in data preprocessing, such as tokenisation and stop-word removal, to ensure the LDA algorithm could effectively interpret the text. Fine-tuning the LDA parameters to optimise topic coherence was a significant focus, requiring iterative testing and validation.
- Topic Trends Visualisation (plotLda.py): Visualising the topics over time presented unique challenges, particularly in effectively conveying the dynamic nature of topics within a static visual format. This script represents a synthesis of data transformation and visual storytelling, converting LDA output into a temporal topic representation that elucidates the evolution of themes over time.
- LDA Optimisation (topicGridSearch.py): The grid search implemented in this script is a testament to the iterative process of refining the LDA model. The challenge here was not just computational efficiency but also determining the optimal range and granularity of parameters to explore. This script underscores the balance between computational resources and the pursuit of model precision.

Topic Extraction with `ldaProcessing.py`

The `ldaProcessing.py` script is integral to the analysis framework, focusing on extracting prevalent topics from the textual data using Latent Dirichlet Allocation (LDA). This process illuminates the dominant themes within the Mastodon posts related to cryptocurrencies.

Data Loading and Preprocessing

The script commences by loading the posts' content from JSON files, preparing it for topic modeling. It incorporates a series of text preprocessing steps, including tokenisation, stop word removal, and lemmatisation, to refine the text for subsequent analysis.

LDA Topic Modeling

LDA is a generative probabilistic model that assumes each document in a corpus is a mixture of latent topics, and each topic is characterised by a distribution over words. The goal of LDA is to discover these latent topics and their associated word distributions.

Mathematically, LDA can be represented as follows:

$$P(w_i|d) = \sum_{j=1}^{K} P(w_i|z_j)P(z_j|d) \tag{4.3}$$

where w_i is the i-th word in the document d, z_j is the j-th topic, $P(w_i|z_j)$ is the probability of word w_i given topic z_j, $P(z_j|d)$ is the probability of topic z_j given document d, and K is the total number of topics.

LDA assigns each word in a document to a topic based on the probability distribution of words for each topic and the distribution of topics for each document. By applying LDA to the cryptocurrency-related discussions, we can uncover the prevalent topics and themes within the community.

Vectorisation

Using the `CountVectorizer` from scikit-learn, the script transforms the preprocessed text into a document-term matrix, essential for LDA processing.

Listing 6 shows the text preprocessing and vectorisation steps using the `CountVectorizer` from scikit-learn.

LDA Application

The script employs `LatentDirichletAllocation` from scikit-learn to perform topic modeling, extracting significant patterns and topics from the document-term matrix.

Listing 7 demonstrates the application of Latent Dirichlet Allocation (LDA) using the `LatentDirichletAllocation` from scikit-learn.

Topics Extraction and Display

Post-LDA, the script identifies and presents the topics, showcasing the top words associated with each to elucidate the theme.

Listing 8 shows the extraction and display of topics and their associated top words.

4.3 Analysis

`ldaProcessing.py` is pivotal for delineating the thematic landscape within the Mastodon cryptocurrency discussions, offering a granular view of the discourse. This section elucidates the script's contribution to the overarching analysis, highlighting its role in distilling complex textual data into discernible topics.

4.3.1 Visualising LDA Results with `plotLda.py`

The `plotLda.py` script is integral to the post-modeling phase, transforming the abstract topics generated by LDA into tangible and interpretable visualisations. These visual representations are crucial for understanding the thematic structure extracted from the data.

Data Loading and Transformation

Initially, the script retrieves the topic modeling output, typically stored in a CSV file, which includes the topics, their top words, and associated metrics like word frequencies or relevance.

Visualisation

The script leverages a visualisation library, such as plotly, to create dynamic and informative charts that illustrate the topic distribution and their constituent words.

Creating Topic Visualisation

Using plotly, the script generates visualisations like bar charts or heatmaps that display the topic-word relationships or the distribution of words across topics.

Listing 9 demonstrates the reading and preparation of data for visualisation.

Listing 10 shows the generation of visualisations using the `plotly` library.

Analysis

By converting the LDA results into visual formats, `plotLda.py` enables researchers and analysts to intuitively grasp the extracted topics' nature and distribution. The script's output serves as a bridge between complex data analysis and actionable insights, facilitating a deeper understanding of the underlying themes in the dataset.

4.3.2 Optimising LDA Parameters with `topicGridSearch.py`

The `topicGridSearch.py` script is fundamental to enhancing the LDA model's performance by fine-tuning its parameters. This optimisation is crucial for improving the relevance and distinction of the topics extracted from the text data.

Data Loading and Preprocessing

Initially, the script processes the textual data, preparing it for the LDA model by cleaning, tokenising, and transforming it into a document-term matrix using CountVectorizer, similar to the preprocessing in `ldaProcessing.py`.

Grid Search for LDA Optimisation

Setting Parameters

The script sets up a parameter grid that includes various options for the number of topics, learning decay rates, and other relevant LDA parameters, preparing for an exhaustive search to find the optimal configuration.

Listing 11 demonstrates the setup of the parameter grid and the grid search using the `GridSearchCV` from scikit-learn.

Grid Search Execution and Model Evaluation

The script executes the grid search over the defined parameter space, assessing each model's performance to determine the best set of parameters based on predefined metrics like coherence or perplexity.

Listing 12 shows the execution of the grid search and the evaluation of the results.

Analysis

By identifying the optimal parameters, `topicGridSearch.py` significantly contributes to the robustness and interpretability of the LDA model's topic extraction. This script ensures that the model is well-tuned to capture the nuances of the dataset, thereby enhancing the overall quality and utility of the topic modeling process.

Chapter 5

Model Training and Inference

5.1 Model Architecture

For this project, the GPT-neo-x-20B model architecture, developed by EleutherAI[27], was utilised. This large-scale language model has 20 billion parameters, making it a powerful tool for natural language processing tasks. To optimise the model's performance and reduce memory footprint, 4-bit quantisation was applied using the bitsandbytes library [28]. The model was loaded with specific configurations, such as using double quant, nf4 quant type, and bfloat16 compute dtype, to ensure compatibility with the quantisation scheme and efficient computation. The choice of GPT-neo-x-20B was motivated by its strong performance on various language tasks and its ability to generate coherent and contextually relevant text. By applying 4-bit quantisation, we were able to significantly reduce the model's memory requirements while maintaining its performance, making it feasible to fine-tune the model on the custom dataset.

5.2 Dataset Preparation

The training dataset consisted of two sources: a dataset of social media posts related to cryptocurrencies collected as part of the project, and data from the Mastodon platform. The datasets were loaded using the Hugging Face Datasets library [29] and concatenated to form a single dataset. Preprocessing steps were applied to the dataset, including tokenisation, truncation, and padding, to prepare the text data for training. The maximum sequence length was set to 128 tokens to ensure efficient processing and to capture the most relevant information within each post. By combining data from multiple sources, we aimed to create a diverse and representative dataset that captures various aspects of cryptocurrency discussions on social media. The preprocessing steps were essential to ensure that the data was in a suitable format for training the language model.

5.3 Training Process

The model training process was based on the notebook 'transformers meets bitsandbytes for democratizing Large Language Models (LLMs) through 4bit quantization' found on Google Colab [30]. The notebook was adapted to train the GPT-neo-x-20B model on the custom social media dataset using 4-bit quantisation and the PEFT (Parameter-Efficient Fine-Tuning) library from Hugging Face [31].

The hyperparameters used in this project were selected based on careful consideration and experimentation to optimise the model's performance on the specific task and dataset. The `per_device_train_batch_size` was set to 4, and `gradient_accumulation_steps` to 8, to ensure efficient utilisation of available hardware resources while maintaining a reasonable effective batch size. The `warmup_steps` and `max_steps` were adjusted to 100 and 500, respectively, to match the size and complexity of the dataset and to prevent overfitting. The `learning_rate` of 1e-4 was chosen as a conservative value to ensure stable training, and the use of `fp16` (half-precision floating-point format) helped to reduce memory usage and speed up computations. The `logging_steps` were set to 50 to provide regular updates on the training progress. Finally, the `"paged_adamw_8bit"` optimiser was used to enable efficient 8-bit optimisation. These hyperparameters differ from the original notebook, which used different values for `per_device_train_batch_size`, `gradient_accumulation_steps`, `warmup_steps`, `max_steps`, and `learning_rate`, as they were tuned specifically for this project's requirements and dataset.

The training was performed using the Hugging Face Transformers Trainer class [32]. The training hyperparameters were set as follows:

- per_device_train_batch_size: 4
- gradient_accumulation_steps: 8
- warmup_steps: 100
- max_steps: 500
- learning_rate: 1e-4
- fp16: True
- logging_steps: 50
- optim: "paged_adamw_8bit"

The model was prepared for efficient fine-tuning using the prepare_model_for_kbit_training function from the PEFT library. Gradient checkpointing was enabled to reduce memory usage during training, allowing for more efficient utilisation of computational resources.

The PEFT library and LoRA (Low-Rank Adaptation) configuration offer several benefits for efficient fine-tuning and model storage. PEFT provides a set of techniques to reduce the number of trainable parameters in the model, making the fine-tuning process more memory-efficient and faster. By using methods like adapters, prefix tuning, and LoRA, PEFT allows for effective fine-tuning of large language models like GPT-neo-x-20B without requiring the entire model to be trained.

LoRA, in particular, is a parameter-efficient fine-tuning technique that adds a low-rank matrix to the model's weights, reducing the number of trainable parameters while still allowing the model to adapt to the specific task. This means that only the LoRA configuration needs to be saved and loaded during inference, rather than the entire fine-tuned model, resulting in significant storage savings. Moreover, the LoRA configuration can be easily applied to the pre-trained model, enabling quick and efficient deployment of the fine-tuned model.

By leveraging the PEFT library and LoRA configuration, this project was able to efficiently fine-tune the large GPT-neo-x-20B model on a custom dataset, while minimising memory usage and storage requirements. This approach makes it more feasible to work with large language models and adapt them to specific tasks, even with limited computational resources.

5.4 Model Saving and Loading

After training, the fine-tuned model was saved using the save_pretrained method from the Transformers library. The LoRA configuration was saved alongside the model to enable efficient loading and inference. To load the fine-tuned model for inference, the LoraConfig was loaded from the saved directory, and the get_peft_model function was used to apply the LoRA configuration to the pre-trained GPT-neo-x-20B model. This approach allows for efficient storage and deployment of the fine-tuned model, as only the LoRA configuration needs to be saved and loaded, rather than the entire model.

Chapter 6

Results and Discussion

6.1 Empirical Analysis of Sentiment Analysis vs Market Trends

Aggregated Sentiment Over Time

The aggregate sentiment over time shows significant fluctuations in overall sentiment towards cryptocurrencies within various coin communities. The sentiment starts off relatively low in early 2022, but then rises sharply to a peak around 0.3 in mid-2022. This is followed by an equally sharp decline, with sentiment turning slightly negative by late 2022. Sentiment recovers in early 2023, reaching another peak above 0.3, before declining again and stabilising around 0.2 for much of mid to late 2023. The start of 2024 sees another uptick in sentiment. Overall, the data suggests that crypto sentiment is quite volatile, with alternating waves of optimism and pessimism over the 2-year period analysed.

The sharp decline in 2022 coincides with the collapse of FTX, which occurred in November. 'This was a major cryptocurrency exchange which handled around $1 billion transactions each day. Its collapse had a knock-on effect on other crypto exchanges'. [42] This could be a possible factor in the sharp decline that took place as crypto investors in many of the major coins were affected.

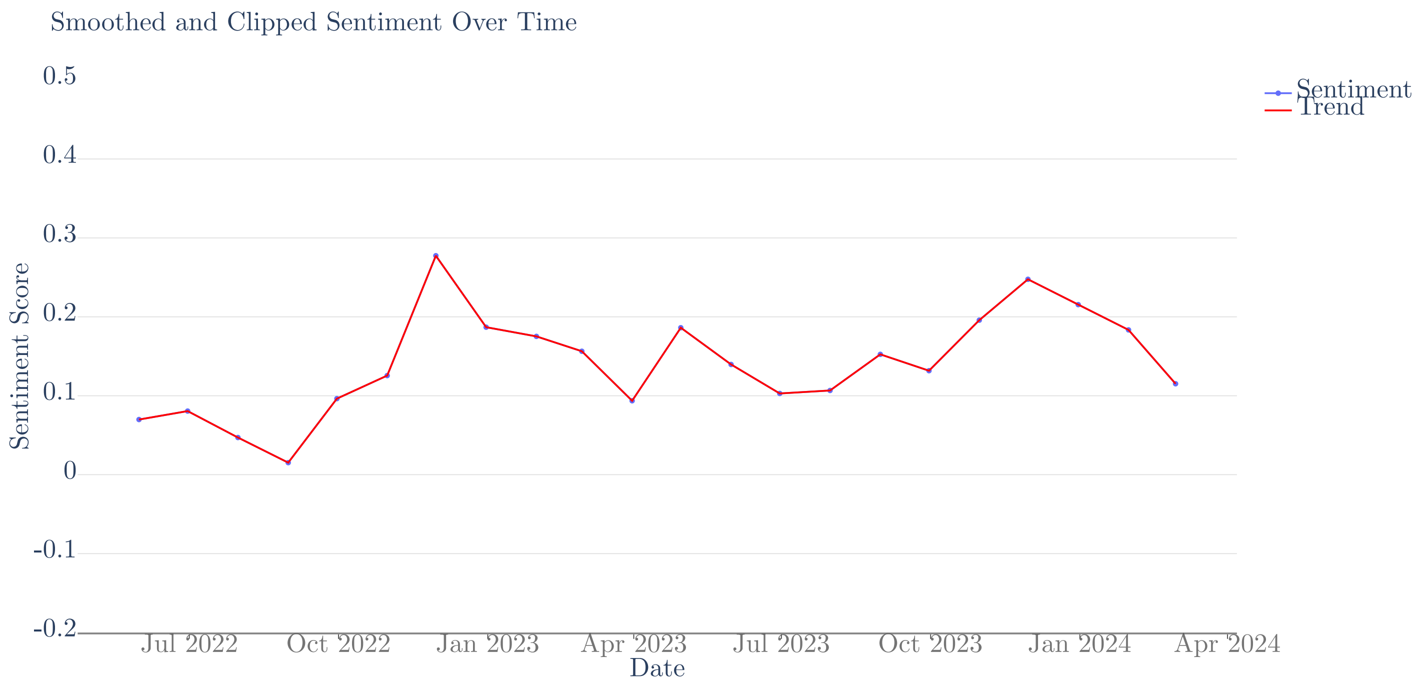

Figure 6.1: Aggregated sentiment scores plotted over time, illustrating the overall sentiment trend within the cryptocurrency discussions.

Correlation Analysis

The analysis unveils a notable correlation between market capitalisation and sentiment for several major cryptocurrencies, such as BNB, BTC, ETH, MATIC, XMR and ICP, illustrating the interplay between public sentiment and market valuation.[43]

BNB (Binance Coin) Sentiment vs. Market Cap Correlation

The sentiment vs price analysis for BNB shows that while the lines may not completely align, there are notable similarities in the overall trends. The primary sentiment (green line) remains relatively stable in the 0.6-0.8 range, while the BNB price (blue line) shows more fluctuation. However, the general shapes of the lines exhibit some commonalities, particularly in the peaks and troughs, suggesting a potential relationship between sentiment and price movements.

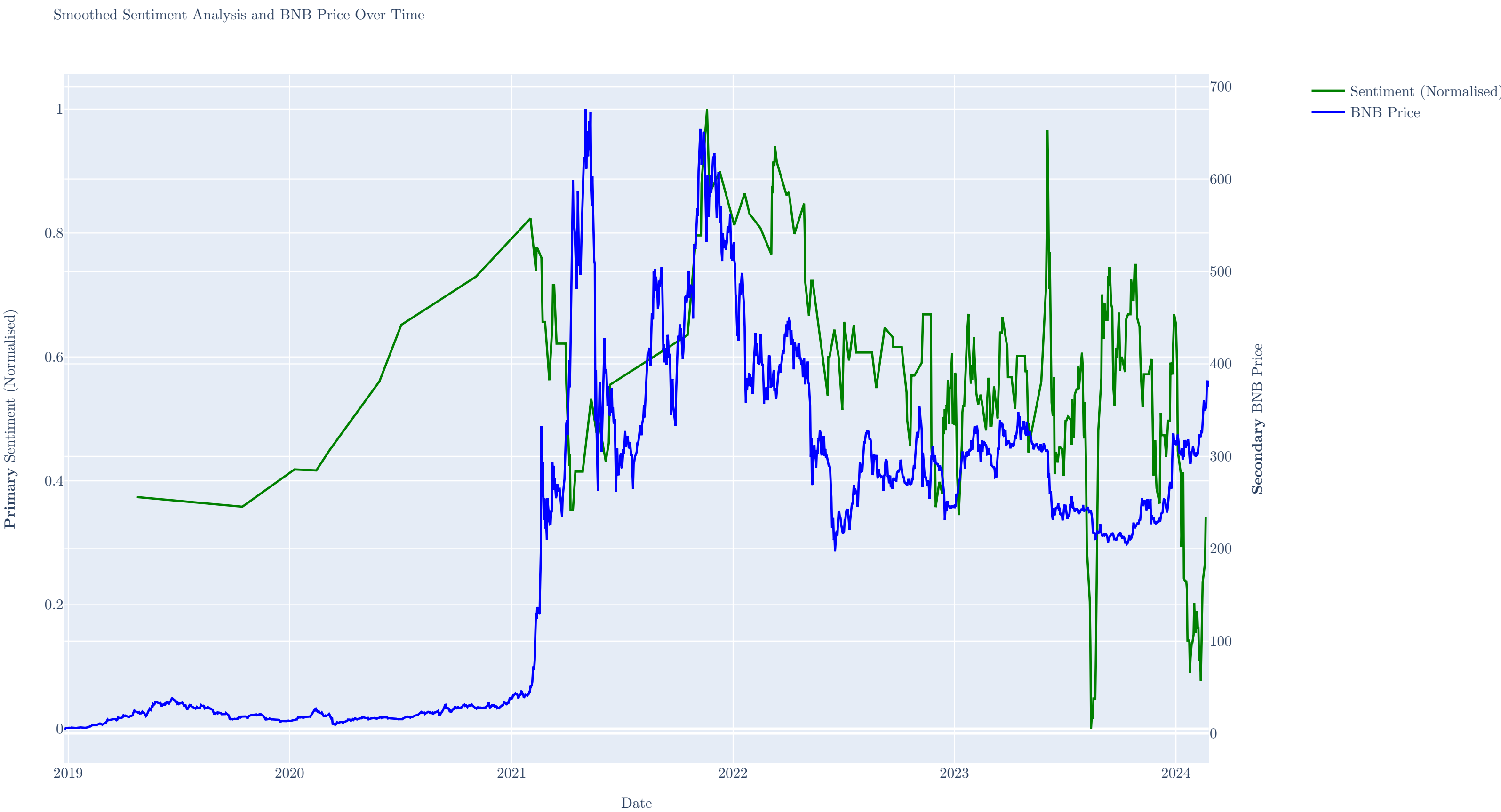

Figure 6.2: Correlation between sentiment and market cap for BNB.

BTC (Bitcoin) Sentiment vs. Market Cap Correlation

For BTC, the sentiment (green line) and price (blue line) display more apparent similarities in their overall shapes. The peaks and dips in sentiment often correspond with similar movements in price, even if the magnitude of the changes differs. This suggests a stronger correlation between BTC sentiment and price compared to BNB. The late 2023 BTC price peak around $52k coincides with a sentiment peak. Sentiment declines in early 2024 as price also retraces. However, the relationship is not perfect, as the early 2023 sentiment spike preceded the subsequent price rally.

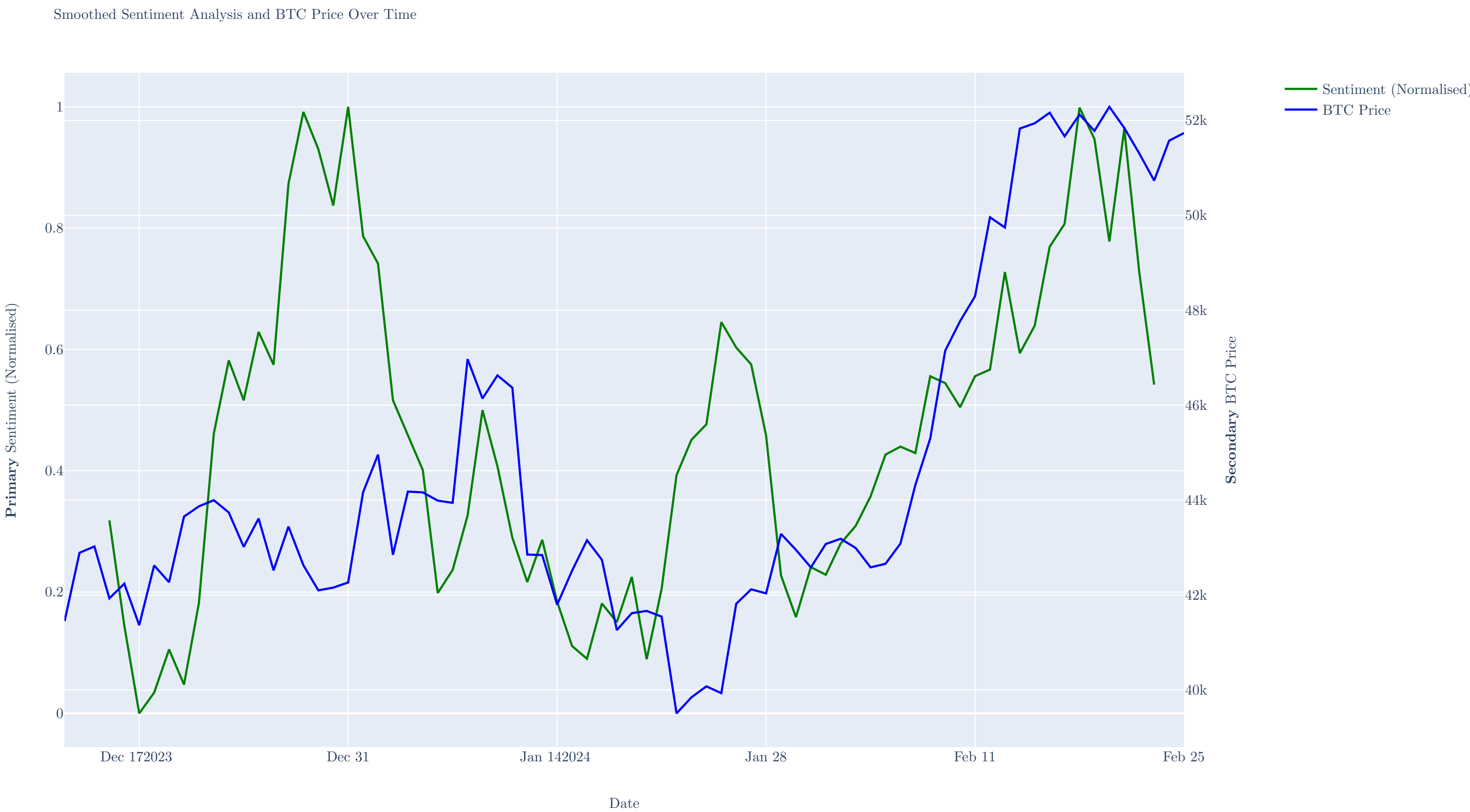

Figure 6.3: Correlation between sentiment and market cap for BTC.

ETH (Ethereum) Sentiment vs. Market Cap Correlation

ETH displays a chart pattern similar to BTC, with sentiment (green line) largely ranging between 0.6-0.9 and showing comparable trends with price action (blue line) throughout the period. Sentiment spikes in early 2023 as price begins to rally. The ETH price reaches a peak around $3000 in late January 2023 while sentiment remains elevated. Subsequently, sentiment declines through February 2023 as price also retraces. The shapes of the lines are quite similar, with sentiment spikes and dips often aligning with corresponding price movements. This indicates a notable correlation between ETH sentiment and price, even if the lines do not always line up exactly.

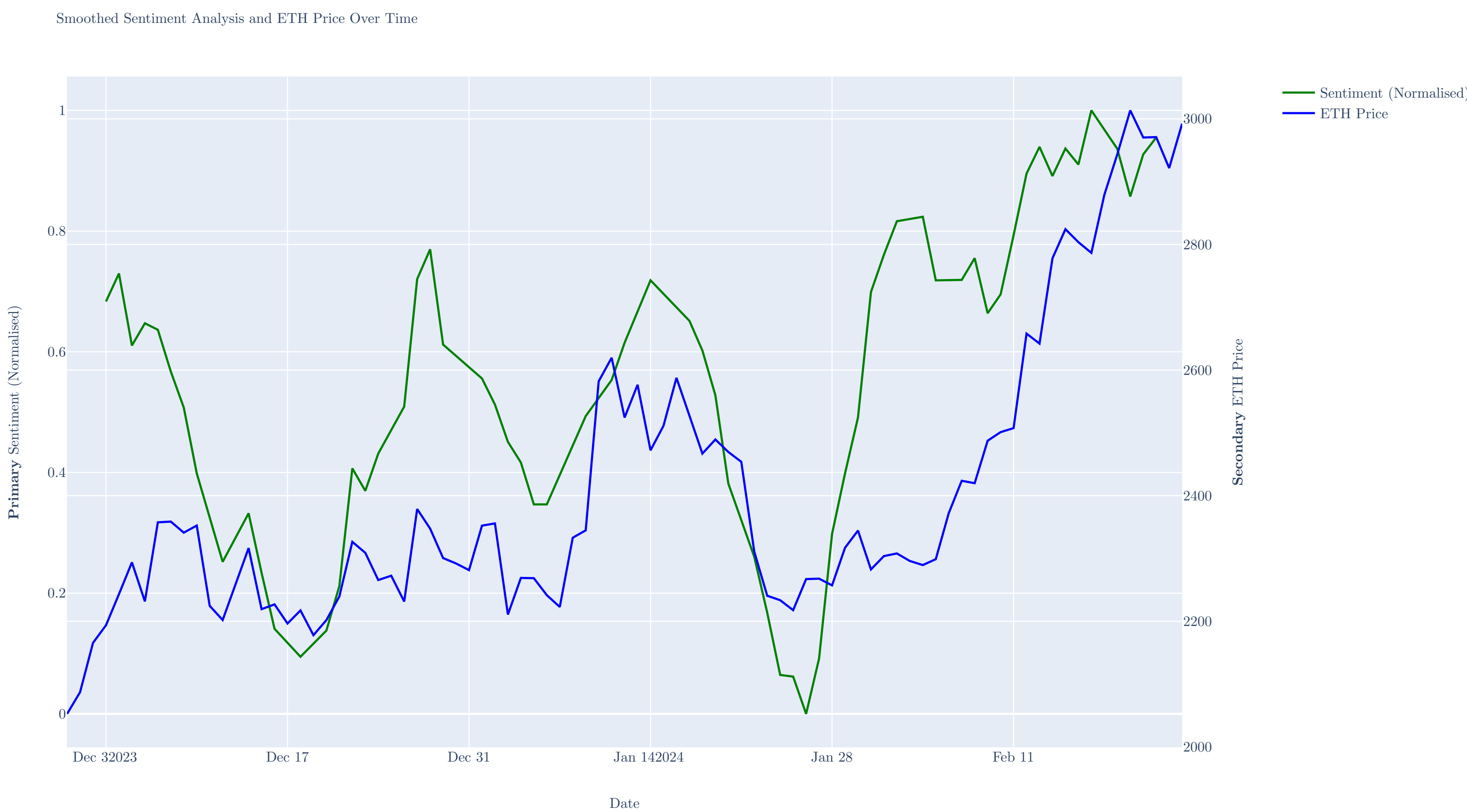

Figure 6.4: Correlation between sentiment and market cap for ETH.

MATIC (Polygon) Sentiment vs. Market Cap Correlation

MATIC sentiment (green line) shows the highest volatility among the coins analysed, with dramatic spikes and dips between 0.2 to 1.0 throughout the period. Price action (blue line) also sees significant volatility, though the correlation with sentiment appears weaker than for BTC and ETH. The most notable alignment is in January 2022 when a sentiment spike coincides with a MATIC price peak around $2.80. However, when looking at shape, MATIC demonstrates a striking similarity. The pronounced spikes and dips in sentiment closely mirror the price action, suggesting a strong relationship between MATIC sentiment and market movements. Even though the magnitude of the changes may vary, the overall trends are remarkably alike.

Figure 6.5: Correlation between sentiment and market cap for MATIC.

XMR (Monero) Sentiment vs. Market Cap Correlation

The sentiment vs price analysis for XMR shows that the primary sentiment (blue line) has remained relatively stable, fluctuating between 0.6 and 0.8 for most of the period with a slight upward trend starting from mid-2022. The XMR price (green line) shows significant volatility, with notable peaks in early 2021 and 2023. Despite the price volatility, the overall shape of the sentiment line suggests a gradual increase in positive sentiment over time. This indicates that while the correlation between XMR sentiment and price may not always align, there is a general trend of increasing sentiment alongside long-term price appreciation.

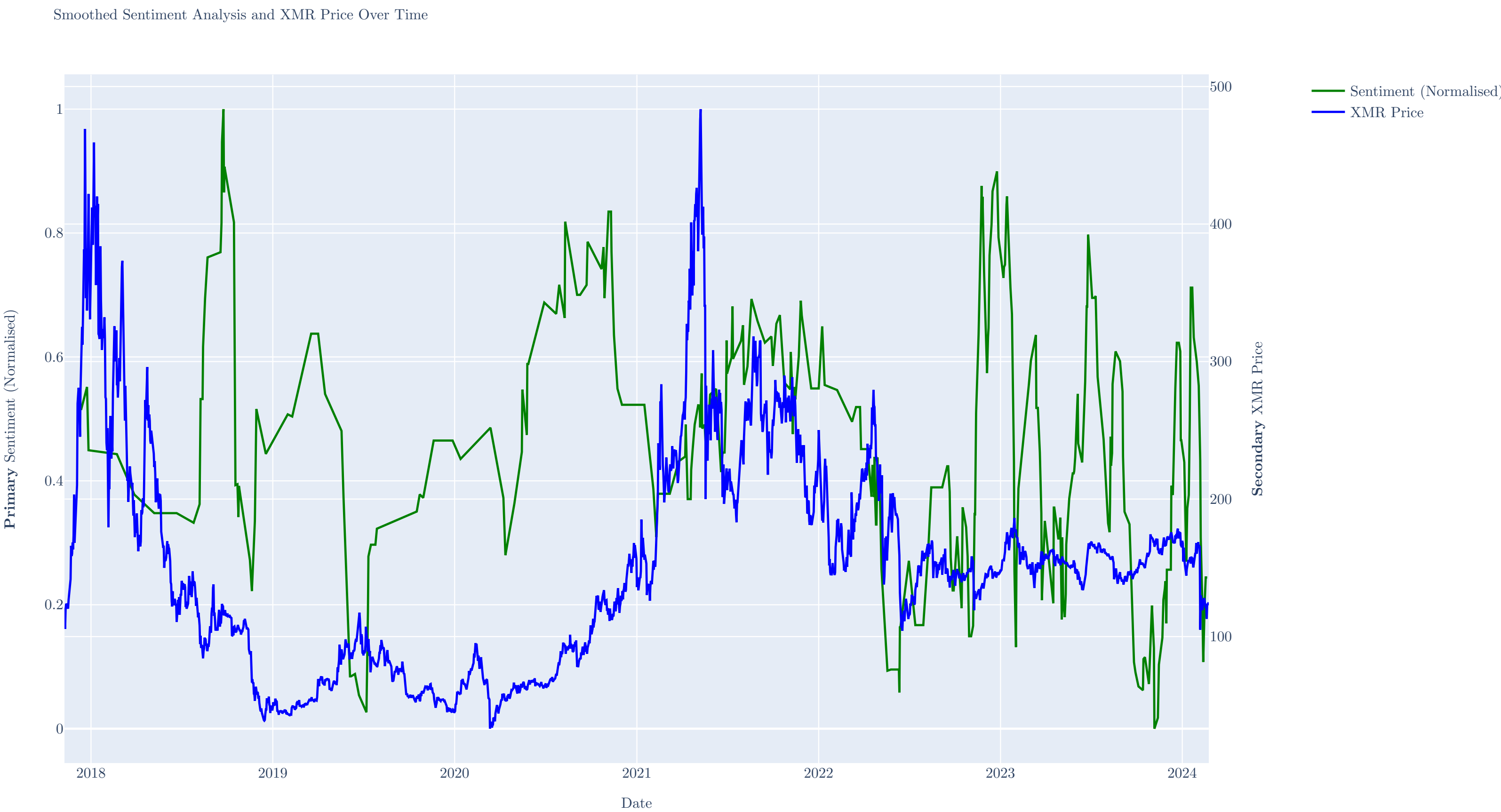

Figure 6.6: Correlation between sentiment and market cap for XMR.

ICP Sentiment vs. Market Cap Correlation

For ICP, the sentiment vs price graph has a slightly shortened time scale to accommodate a sharp decline in market cap value towards the start of the period. The ICP sentiment (blue line) shows significant volatility, ranging from around 0.2 to 1.0 throughout the period, with a slight upward trend from mid-2022 onwards. The ICP price (green line) also displays high volatility, with a significant peak in early 2021 followed by a sharp decline and stabilisation at a lower level. Despite the volatility, the overall shapes of the sentiment and price lines show some similarity, with sentiment increasing during periods of price appreciation and decreasing during price declines. This suggests that while the correlation may not always match up, there is some relationship between ICP sentiment and price movements.

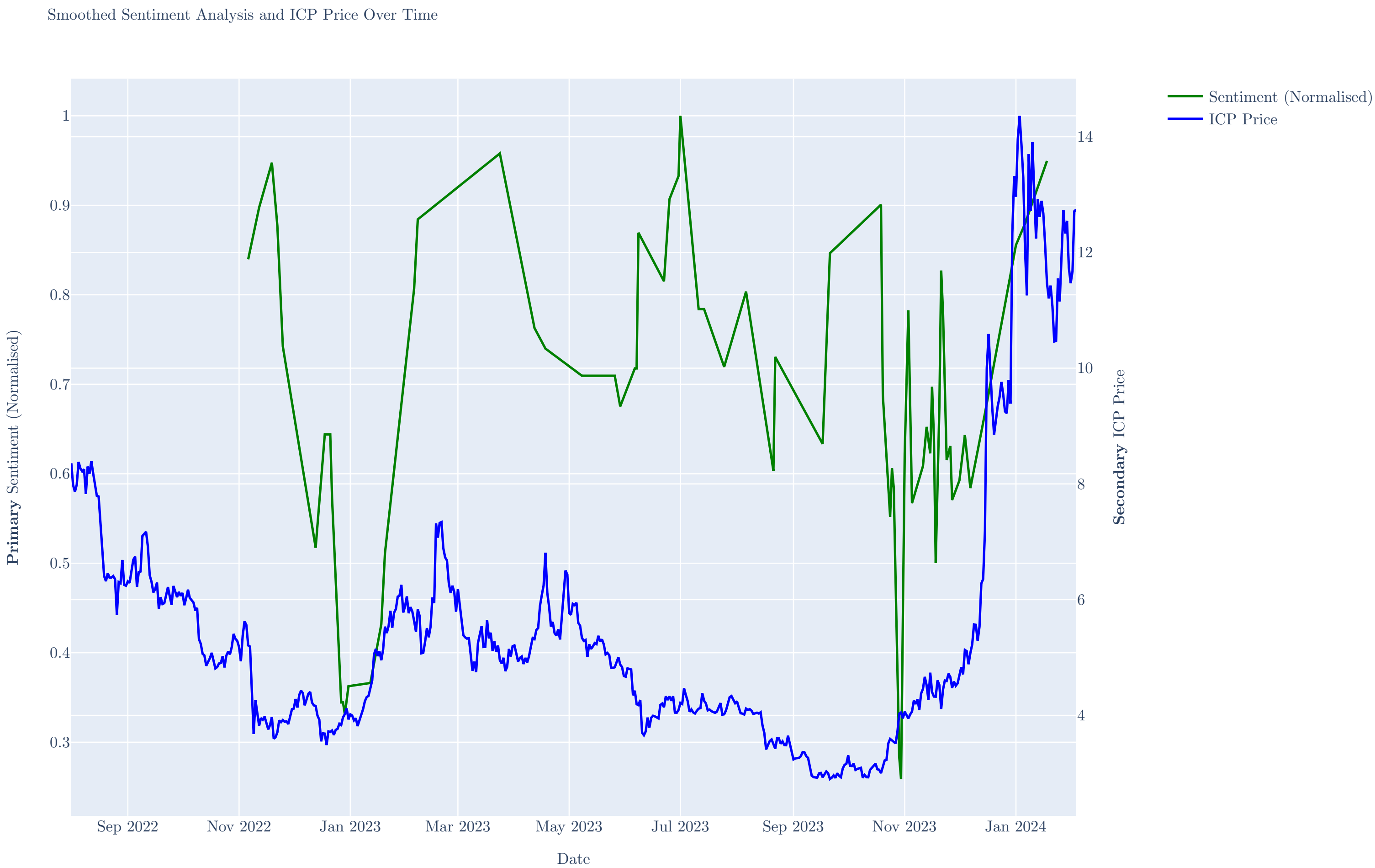

Figure 6.7: Correlation between sentiment and market cap for ICP (Shortened Time Frame in order to scale data appropriately).

listing:1 ICP graph before scale adjustment

LDO Sentiment vs Price Analysis:

For some cryptocurrencies, such as LDO, there was insufficient data to effectively map sentiment onto the market cap. This presented a challenge in the analysis and highlights the limitations of the data available. In such cases, the sentiment vs price graphs could not be meaningfully interpreted or used to draw conclusions about the relationship between sentiment and price. This graph is included in the appendix to showcase this challenge encountered during the project.

listing:1 LDO graph (showcase of insufficient data)

6.2 Empirical Analysis of Topic Modeling

LDA Topic Distribution Visualisation

This visualisation delineates the topic distribution as determined by the LDA model, offering a breakdown of the key terms that define each of the five topics derived from the cryptocurrency-related discussions in the dataset. [44]

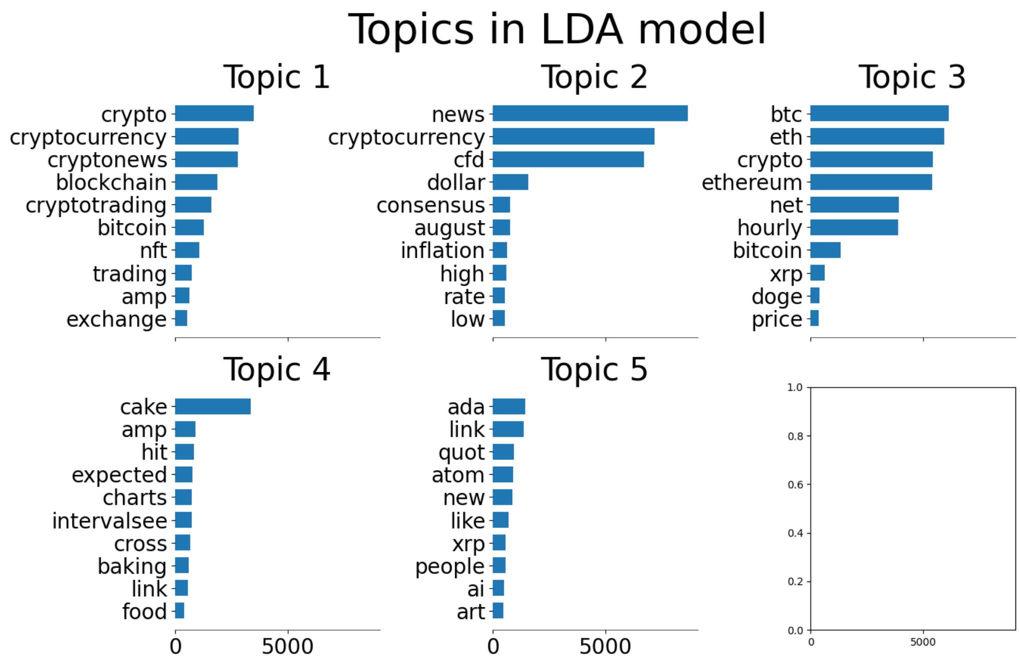

Figure 6.8: Word frequencies for the five topics extracted from the LDA model. Each topic is characterised by a set of keywords that illustrate the thematic essence of discussions within the cryptocurrency community.

In interpreting this graph, we observe that:

- **Topic 1** is characterised by general cryptocurrency terminology and trading concepts, signaling a broad conversation about the crypto market and exchanges.
- **Topic 2** focuses on financial aspects and market dynamics, including terms like "news," "dollar," and "inflation," hinting at discussions around economic impacts on cryptocurrency.
- **Topic 3** seems to capture the technical and specific discourse on various cryptocurrencies, with a mix of names such as "btc" and "eth" and terms like "price" and "hourly," which may reflect trading and valuation discussions.
- **Topic 4** possibly represents a more practical application side of cryptocurrencies, with references to "cake" and "baking" potentially linked to PancakeSwap (this could also be a case of

topic dilution, where there are many posts about pastries), and "charts" and "expected," which could relate to price predictions and trading strategies.

- **Topic 5** appears to revolve around community and development, with terms such as "ada," "link," and "ai" suggesting a focus on technology, community sentiment, and possibly the development of blockchain projects.

This comprehensive graphical analysis assists in understanding the multifaceted nature of cryptocurrency discussions, revealing how different themes emerge and intersect within the community's dialogue.

Topics Over Time

The topic trends over time graph illustrates the shifts in prevalence of various topics related to cryptocurrency, providing a time-series analysis of topic popularity.

The graph showcases the popularity trends of various topics over time. Most topics exhibit significant fluctuations, with notable peaks and valleys throughout the observed period. The overall ranking of topics remains relatively consistent, with "ada" and "bitcoin" consistently ranking among the top topics. However, the gap between the highest and lowest ranked topics appears to be narrowing over time, suggesting a more even distribution of popularity across topics. The latter half of the observed period shows more pronounced changes in topic rankings compared to the earlier months.

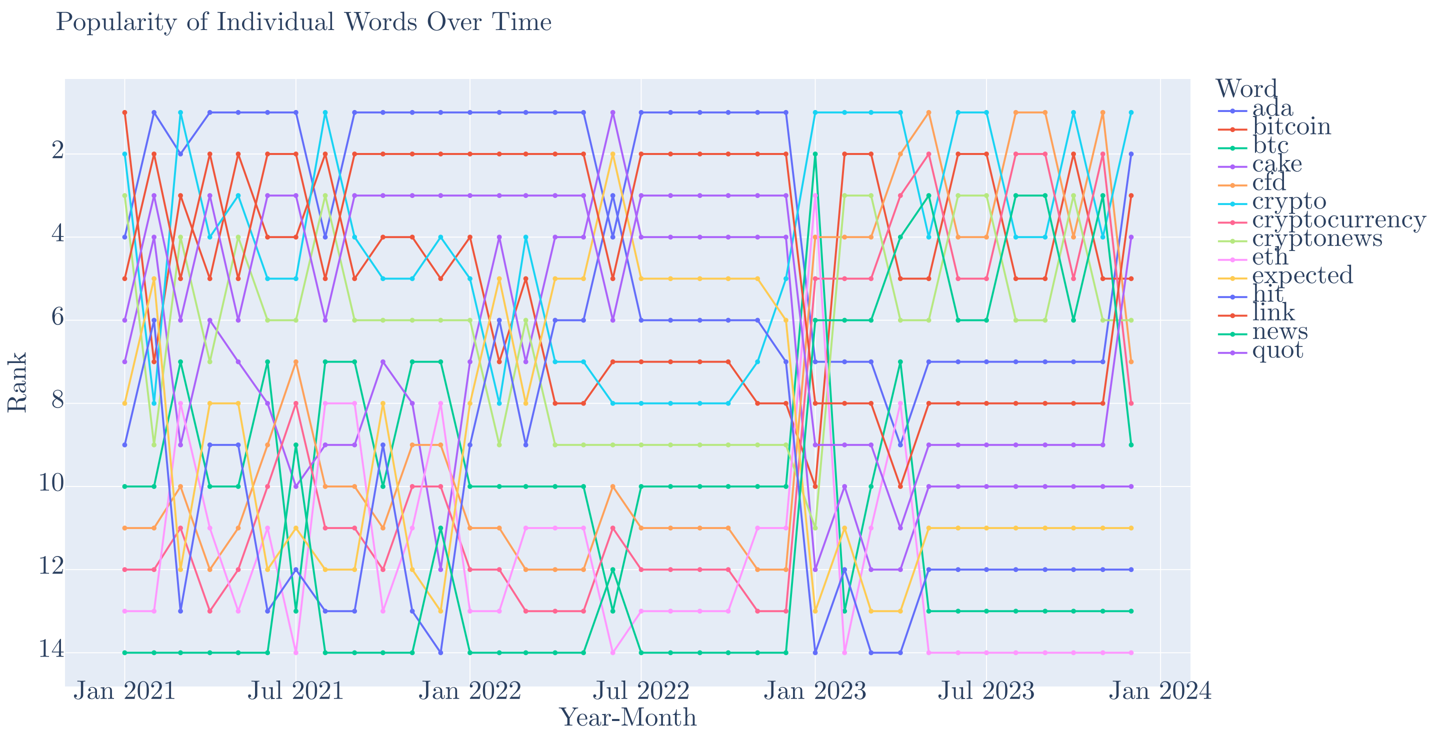

Figure 6.9: Temporal trends of cryptocurrency-related topics, depicting how each topic's prevalence has evolved over time.

ADA (Cardano)

The popularity of the topic ”ada” demonstrates a notable upward trend over the observed period. Starting from a relatively low rank, ”ada” experiences a significant increase in popularity, particularly from mid-2021 onwards. However, the topic experiences a large decline during 2023, which may be attributed to various factors, such as the overall market conditions and the delayed implementation of key releases covering stablecoins, scalability issues, and the platform’s continuing development of oracles [45]. Despite the decline in popularity, the average price of ADA in 2023 was predicted to be $2.85 [46], indicating a potential for recovery in the long term. The graph suggests a fluctuating interest in ”ada” over time, with periods of growth followed by a significant decrease in popularity.

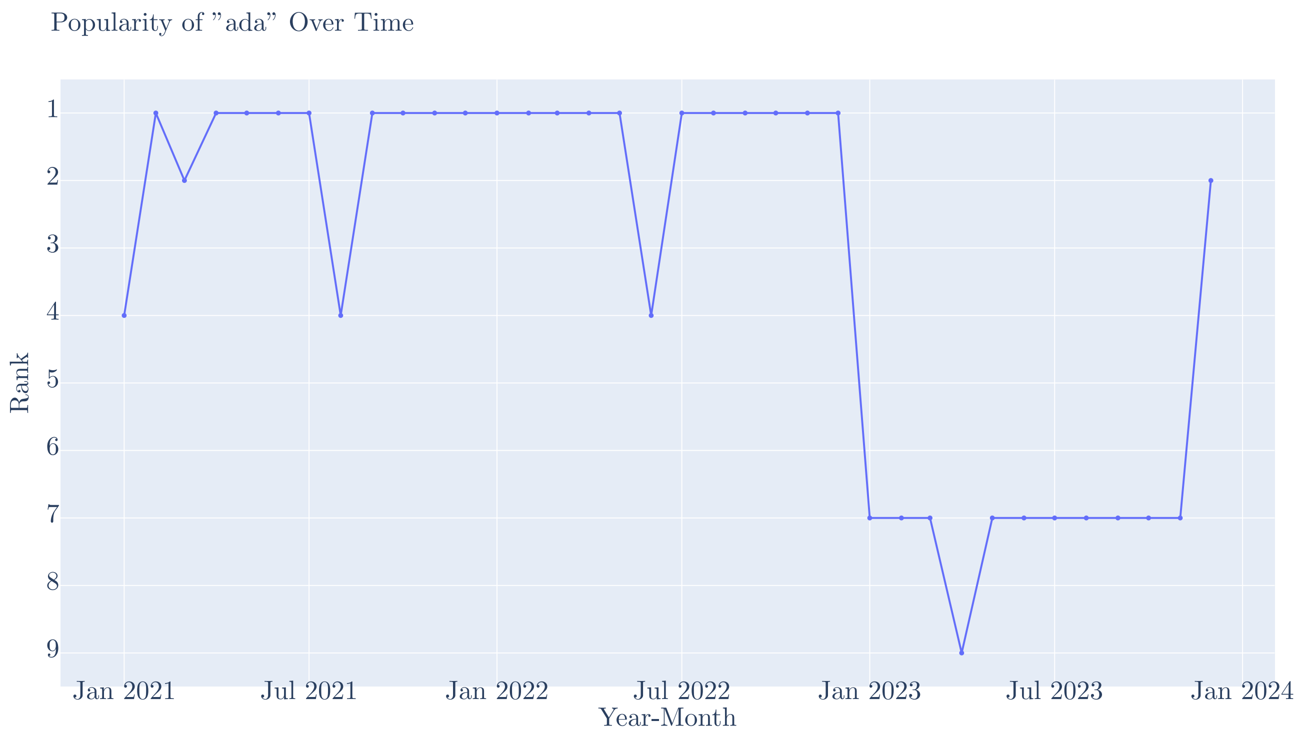

Figure 6.10: Trends in discussions about ADA, indicating Cardano’s engagement and presence in the cryptocurrency community discourse.

Bitcoin (General Discussion)

The topic "bitcoin" exhibits a fluctuating pattern of popularity over the observed period. Initially ranking among the top topics, "bitcoin" experiences a gradual decline in popularity during the first half of the analysis period, particularly in 2022. This decline may be attributed to several factors, including the onset of a new "crypto winter" characterised by the collapse of high-profile companies and a crash in digital currency prices [47]. The failure of stablecoin terraUSD (UST) and the bankruptcy of crypto hedge fund Three Arrows Capital also contributed to the overall decline in the crypto market [47]. Furthermore, the collapse of FTX, one of the world's largest cryptocurrency exchanges, in November 2022 had ripple effects across the industry [47]. Rising interest rates also impacted risk assets like stocks and crypto, with Bitcoin's price dropping approximately 75% from its all-time high in November 2021 [47]. Despite these challenges, "bitcoin" regains its popularity in the latter months, showing a notable resurgence. This suggests a consistent level of interest in the topic, even amidst market volatility.

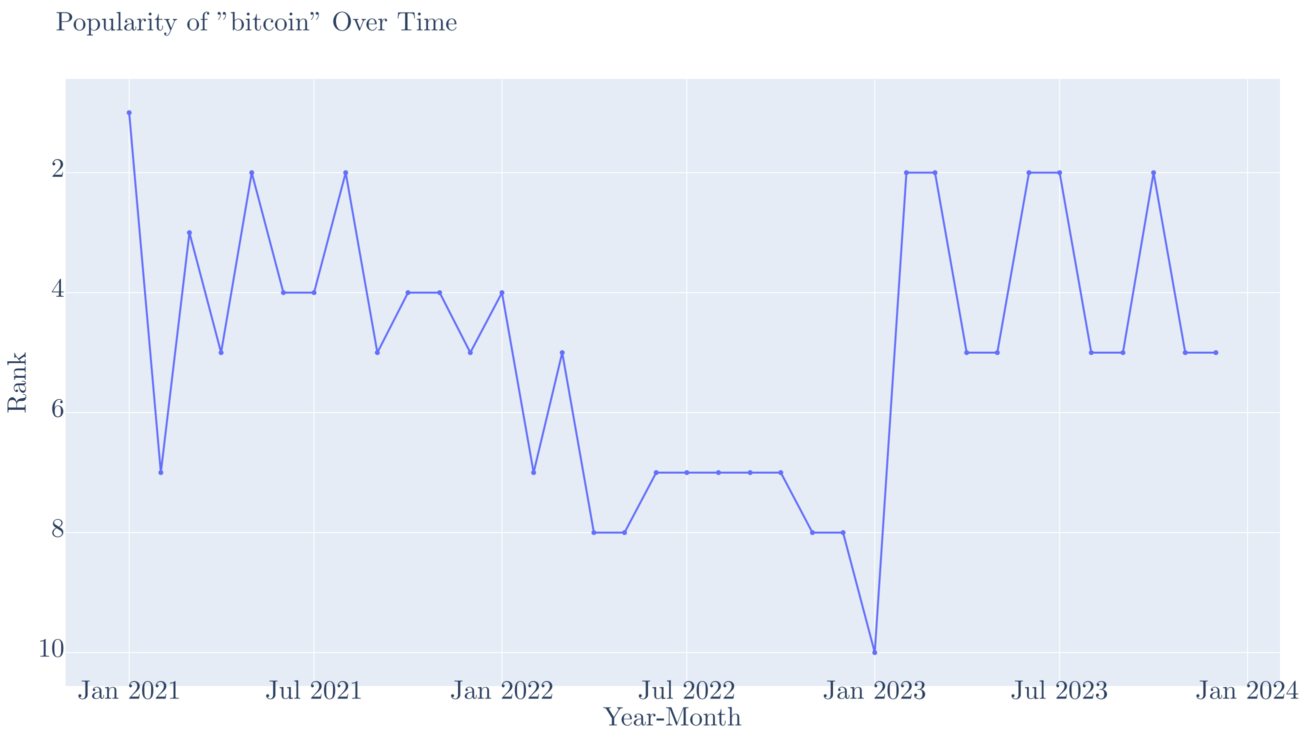

Figure 6.11: Analysis of the general discussion trends surrounding Bitcoin, highlighting the influence of key events on the discourse volume.

BTC (Bitcoin Specific)

The graph showcasing the popularity of the term "BTC" over time reveals a series of fluctuations in its ranking. From January 2021 to July 2021, "BTC" maintains a relatively stable rank between 6 and 8, indicating consistent interest and discussion. However, starting in July 2021, the popularity of "BTC" experiences a notable uptick, reaching a peak rank of approximately 4 in November 2021. This surge in interest suggests a heightened focus on "BTC" during this period, possibly driven by significant developments or market events mentioned earlier. Following the peak, the popularity of "BTC" gradually declines, settling back into a rank range of 6-8 by January 2023. "BTC" follows a very similar pattern to "bitcoin", as can be expected, with it's peaks and dips but normally at a lower ranking. It seems that the term "bitcoin" is preferred within the community with "BTC" ranking consistently in 7th - 13th.

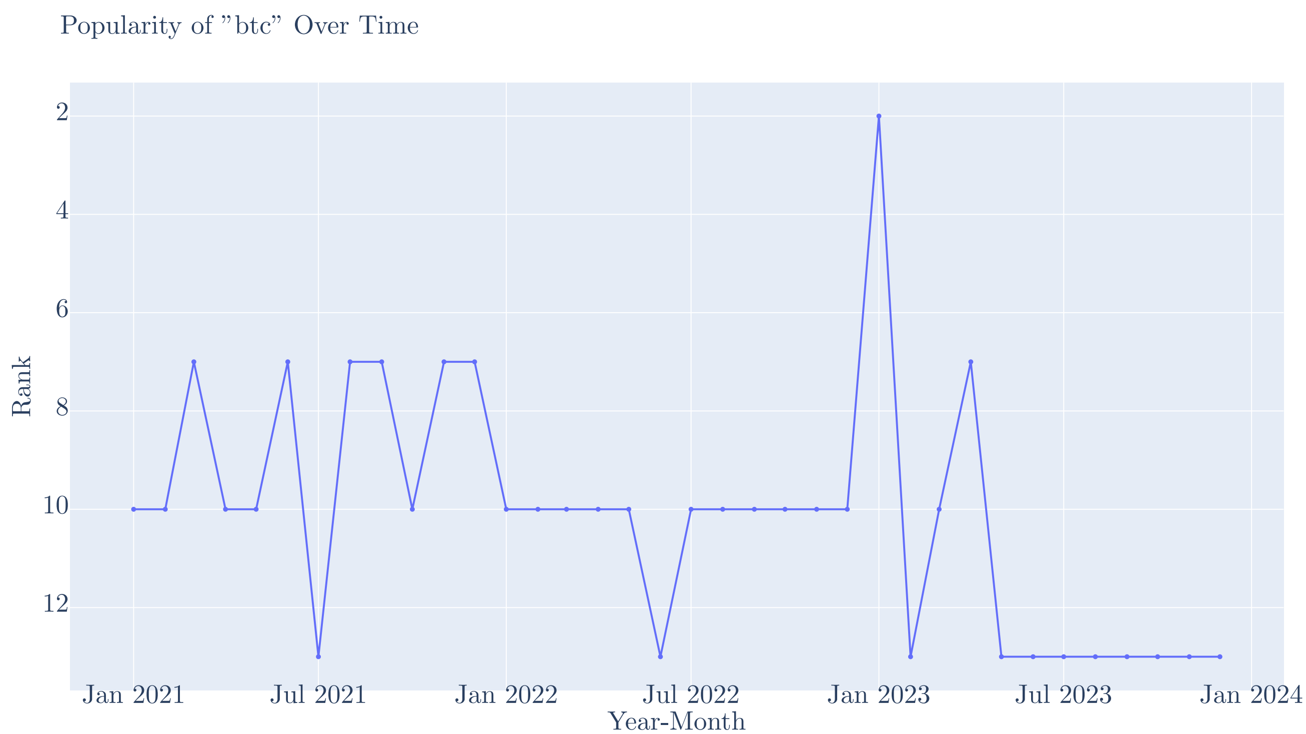

Figure 6.12: Focused examination of discussions specifically mentioning "BTC," providing insights into targeted Bitcoin-related conversations.

CAKE (PancakeSwap)

The popularity trend of the topic "cake" displays a relatively stable pattern over the observed period, with a notable increase in popularity in 2022 compared to its average. This increase in interest may be attributed to several developments within the PancakeSwap ecosystem. In 2022, the PancakeSwap community discussed and voted on a max token supply of 750 million CAKE tokens, aiming for long-term sustainability and providing clarity on CAKE's tokenomics [48]. PancakeSwap also upgraded its popular prediction platform for CAKE in June, allowing users to predict whether CAKE's price will rise or fall [48]. The introduction of iCAKE, a fixed-term CAKE staking benefit, and improvements to the Initial Farm Offerings (IFOs) and CAKE staking side pools may have also contributed to the increased interest in "cake" during this period [48]. However, the topic's popularity dropped off in 2023, possibly due to broader market conditions and a shift in focus within the cryptocurrency community.

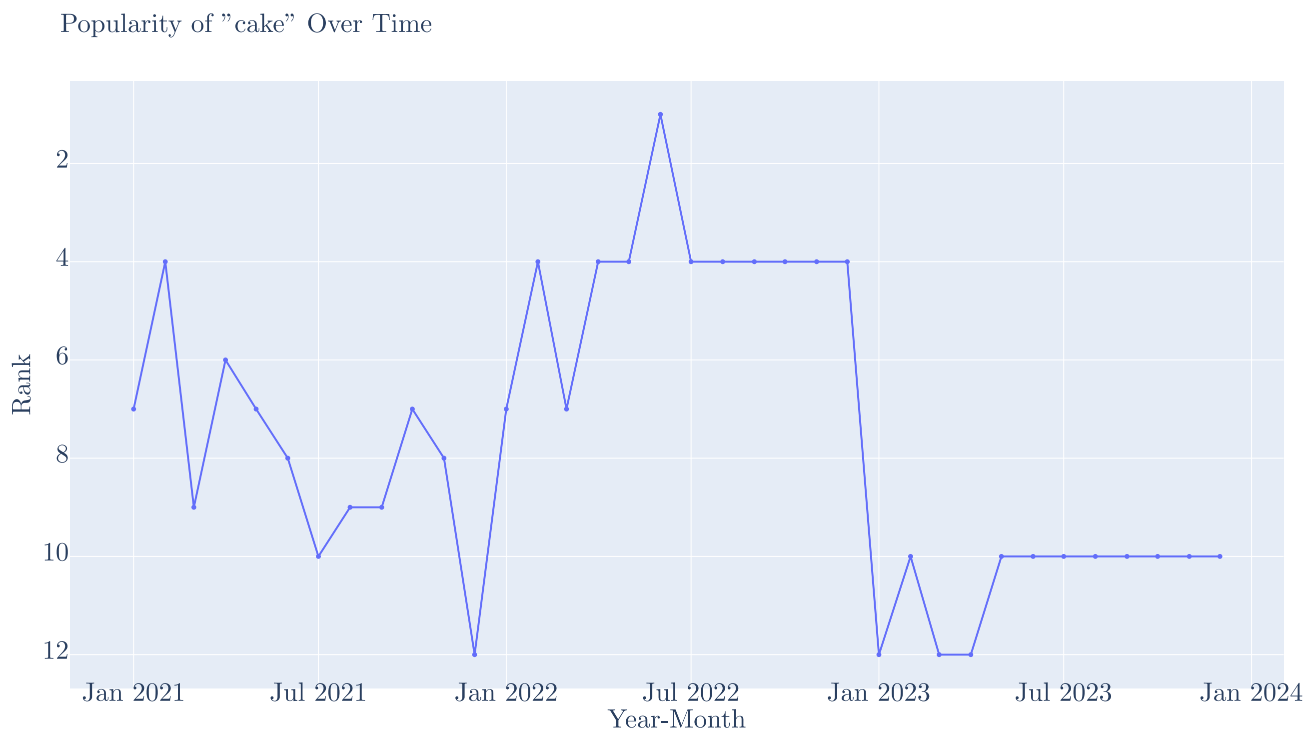

Figure 6.13: Trends in discussions about CAKE, reflecting community engagement with PancakeSwap and DeFi-related topics.

CFD (Contract for Difference)

The graph representing the popularity of the term "CFD" over time exhibits a relatively stable trend with minor fluctuations. Throughout the period from January 2021 to January 2023, the rank of "CFD" remains within the range of 10-12, indicating a consistent level of interest and discussion surrounding this topic. However, CFD experiences a notable spike at the start of 2023 and remains high throughout the year. This increased popularity may be influenced by various economic factors, such as inflation rates, global recessions, or shifts in dominant currencies, which can significantly impact the adoption rate and stability of cryptocurrencies in the trading ecosystem [49]. Cryptocurrency traders were also closely watching Federal Reserve (Fed) meetings, which are regarded as high-impact news and can affect the interest in CFD Crypto [50]. The overall trend suggests that CFD maintains a stable presence in discussions, with a significant increase in popularity during 2023.

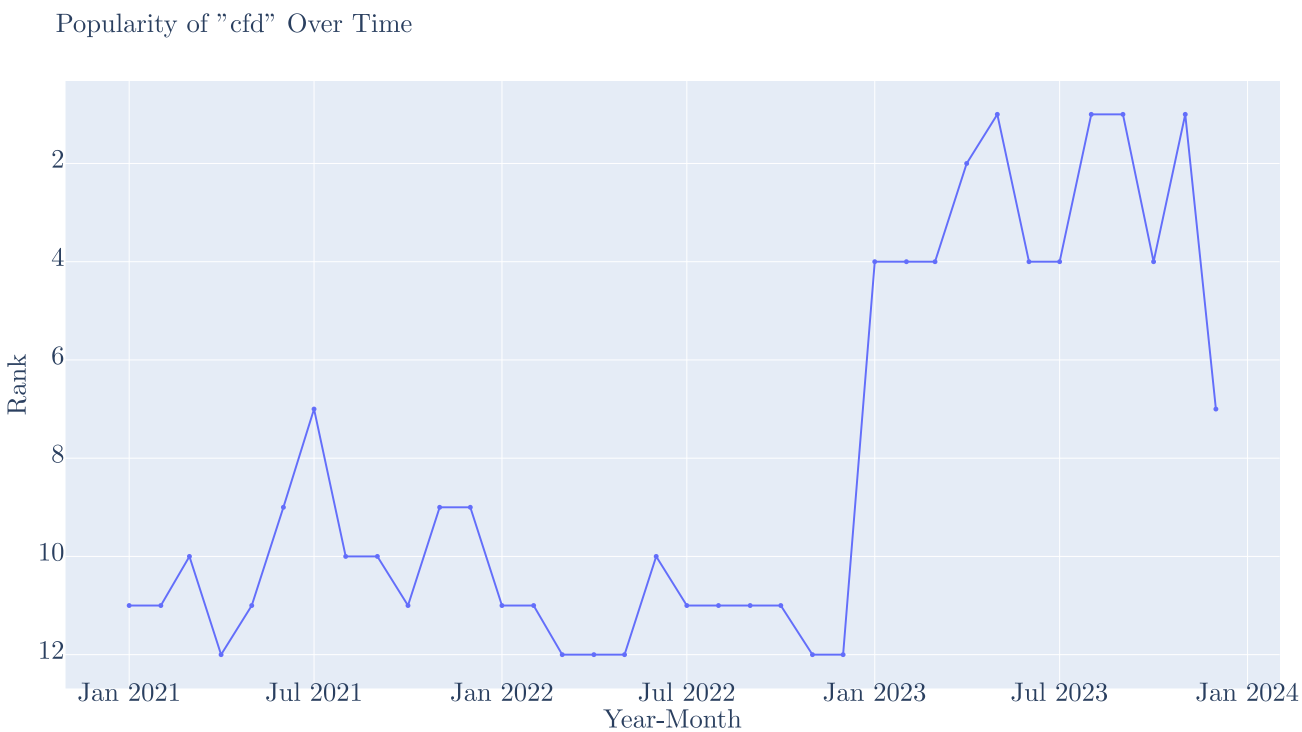

Figure 6.14: Insights into the conversation trends regarding Contracts for Difference within the cryptocurrency community.

Crypto (General Cryptocurrency)

The graph depicting the popularity of the term "Crypto" over time reveals a notable upward trend starting from July 2021. Prior to this point, "Crypto" maintains a rank between 6 and 8, indicating a consistent level of interest. However, beginning in July 2021, the popularity of "Crypto" surges, reaching a peak rank of approximately 3 in November 2021. This significant increase in interest suggests a heightened focus on cryptocurrency-related topics during this period. Following the peak, the popularity of "Crypto" slightly decreases but remains elevated compared to the pre-July 2021 levels, stabilising around a rank of 4-5 through January 2023. The sustained high ranking of "Crypto" indicates that it has become a prominent and frequently discussed topic within the given context. The overall trend highlights the growing importance and mainstream attention garnered by cryptocurrency-related subjects.

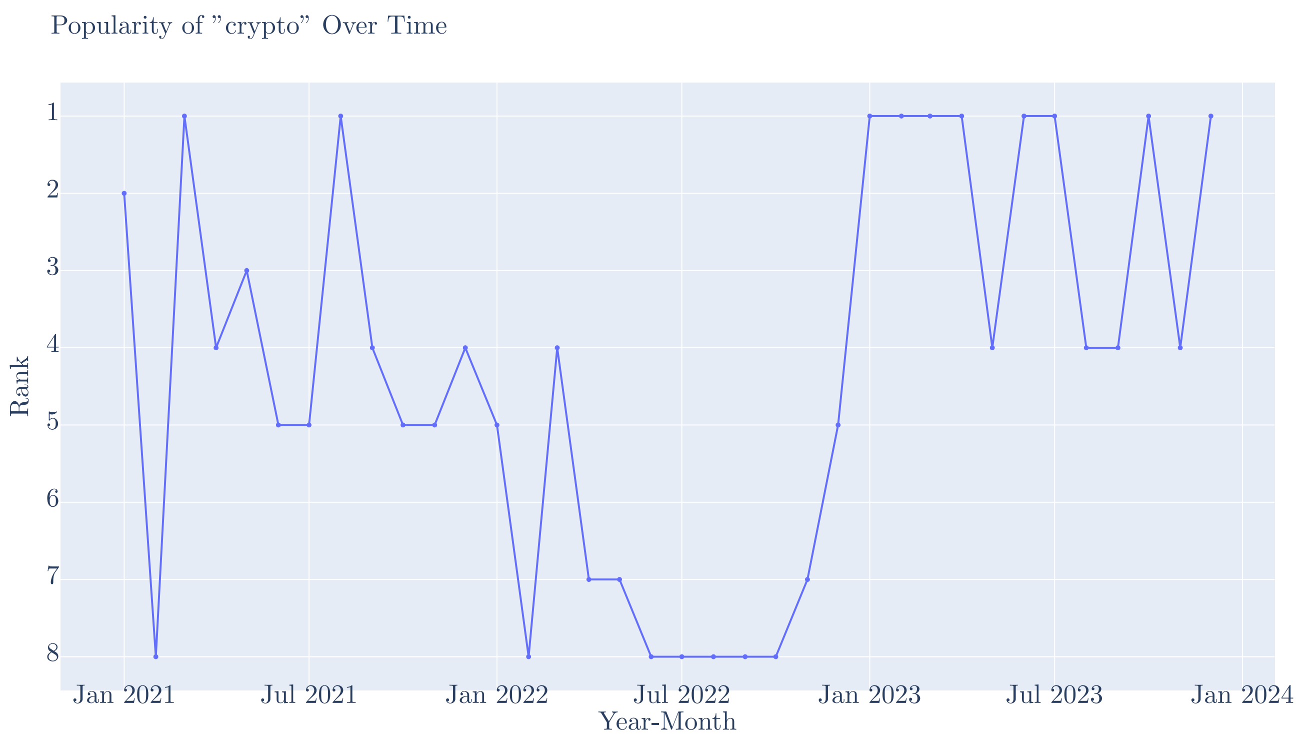

Figure 6.15: Exploration of broader discussions within the cryptocurrency space, capturing sentiment and interest shifts.

Cryptocurrency (Specific Term Usage)

The graph illustrating the popularity of the term "cryptocurrency" over time closely mirrors the trend observed for Crypto. Starting from July 2021, "cryptocurrency" experiences a substantial increase in popularity, rising from a rank of around 8 to a peak of approximately 3 in November 2021. This surge in interest coincides with the heightened focus on cryptocurrency-related topics during the same period. After the peak, the popularity of "cryptocurrency" slightly decreases but maintains a higher rank compared to the pre-July 2021 levels, hovering around a rank of 4-5 through January 2023. The sustained elevated ranking of "cryptocurrency" underscores its significance as a regularly discussed and prominent topic within the given context. The parallel trends observed for Crypto and "cryptocurrency" suggest a strong correlation between these terms and their collective importance in the discourse surrounding digital currencies.

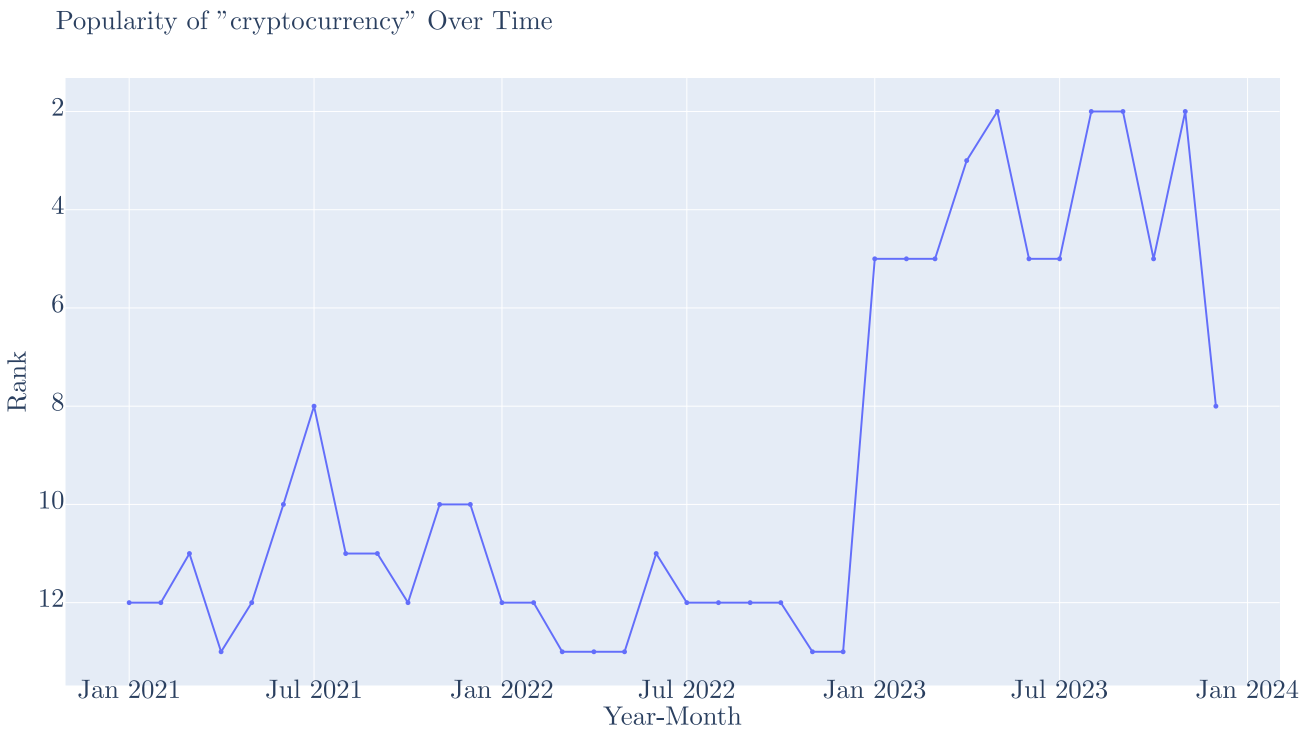

Figure 6.16: Focused analysis on the usage of the term "cryptocurrency," providing a lens on how digital currencies are framed in discussions.

Cryptonews

The graph representing the popularity of the term ”cryptonews” over time displays a relatively stable trend with minor fluctuations. Throughout the period from January 2021 to January 2023, the rank of ”cryptonews” remains within the range of 5-7, indicating a consistent level of interest and discussion surrounding this topic. While ”cryptonews” does not exhibit any significant spikes or drops in popularity, it maintains a solid presence as a regularly mentioned term. The stability in the ranking of ”cryptonews” suggests that it is a topic of ongoing relevance and significance within the context of the data presented. Compared to some of the other terms shown in the legend, ”cryptonews” generally holds a middle-to-upper position in terms of overall popularity. The consistent presence of ”cryptonews” in discussions highlights its importance as a reliable source of information and updates related to the cryptocurrency industry.

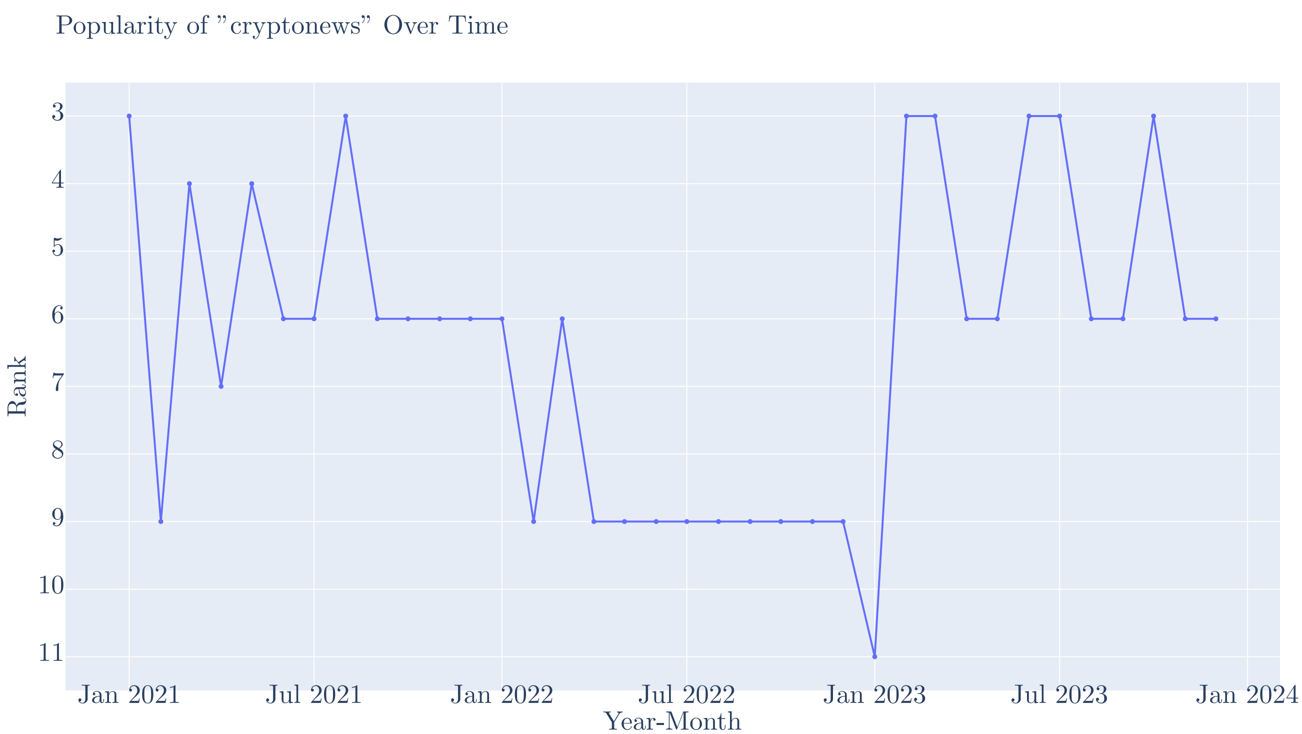

Figure 6.17: Visualisation of the impact of news on cryptocurrency community discourse, highlighting responsiveness to media.

Eth (Ethereum)

The graph showcases the popularity trend of the topic "ETH" over time. The trend exhibits a relatively stable pattern, with a notable spike in January 2023. This surge in popularity may be attributed to the anticipation and implementation of the Shanghai or Shapella upgrade, which enabled users to unlock and withdraw their staked ETH, addressing a critical aspect of user control and liquidity in the Ethereum ecosystem [51]. Ethereum's ongoing transition from proof-of-work to proof-of-stake as part of Ethereum 2.0, aimed at improving scalability, security, and sustainability, may have also contributed to the increased interest in the topic during this period [51]. Despite some slight dips and peaks, the overall trend suggests that the topic maintains a steady level of interest and discussion within the analysed context.

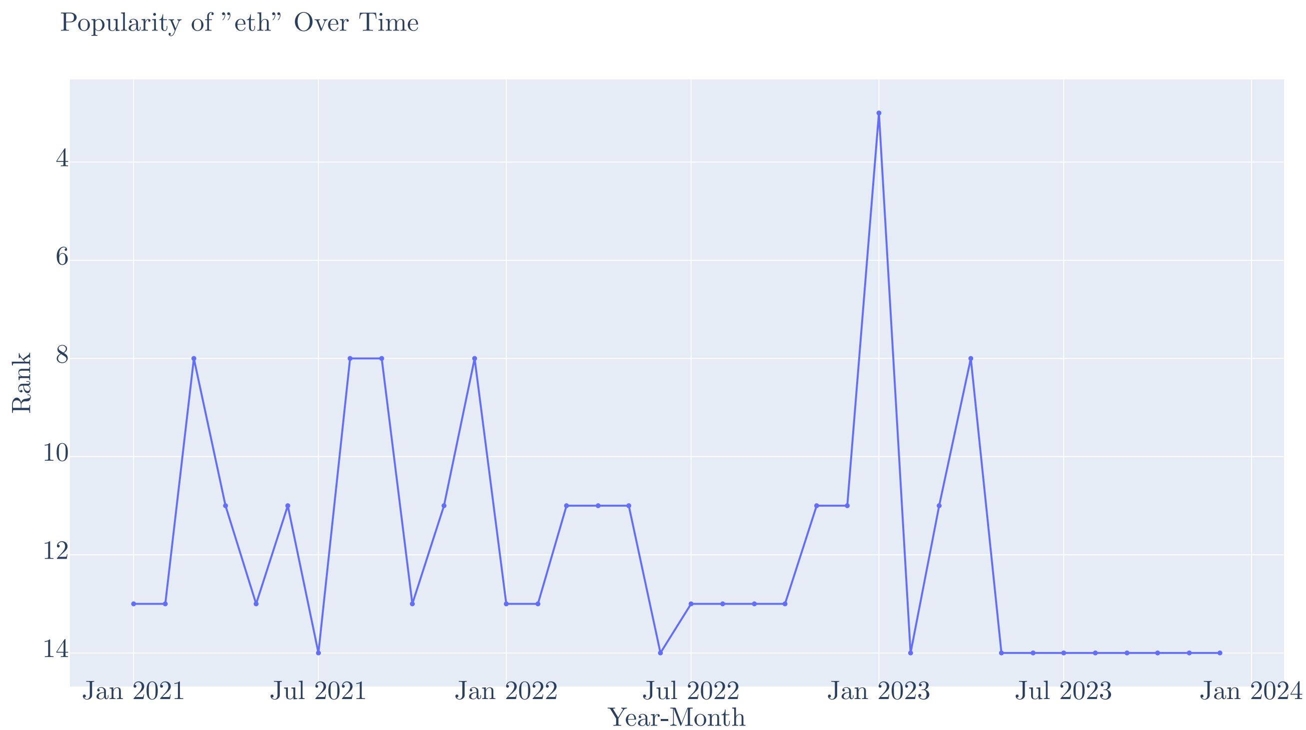

Figure 6.18: Discussion trends regarding Ethereum, indicating the community's focus on Ethereum's development and ecosystem.

Expected

The popularity trend of the topic ”expected” displays a distinct pattern over the given time period. Starting from a rank of approximately 4 in January 2021, the trend line showcases a significant upward trajectory, reaching its peak around July 2022. This suggests a growing interest and relevance of the ”expected” topic during this timeframe. However, following the peak, the trend experiences a notable decline, indicating a decrease in popularity and discussion surrounding the topic towards the end of the analysed period.

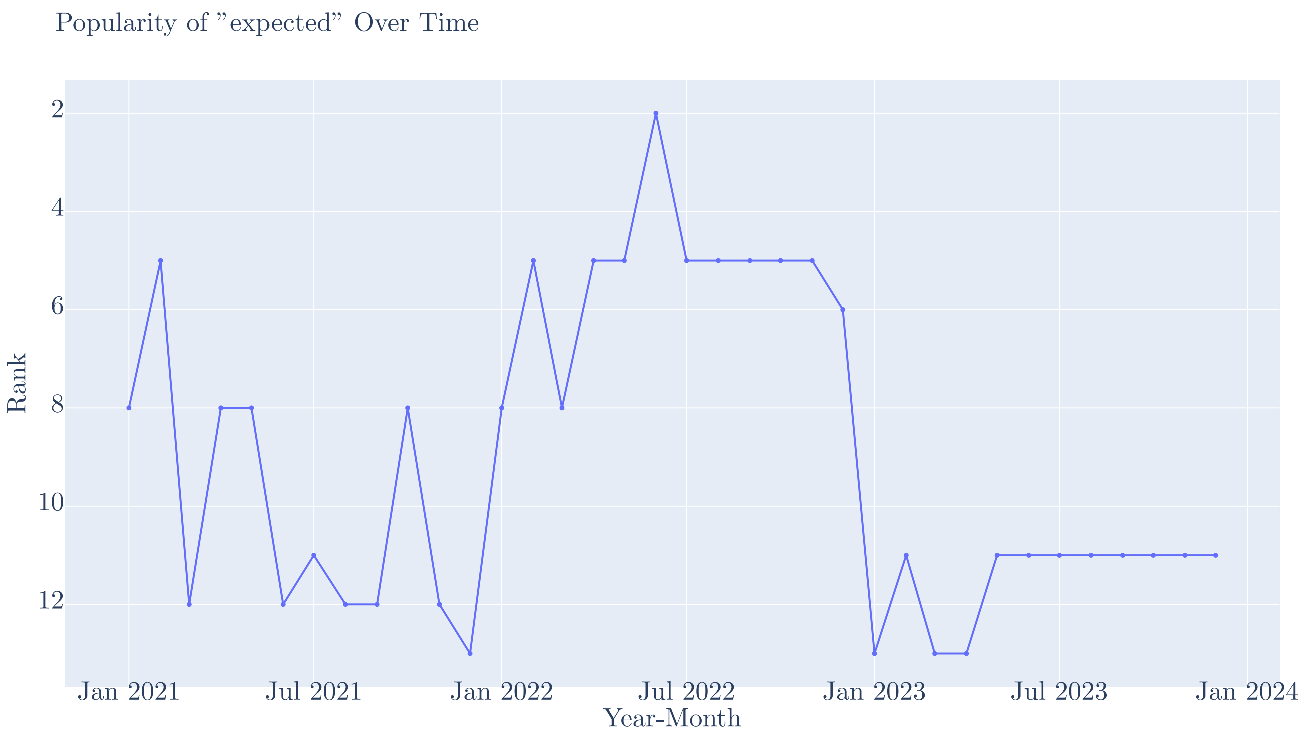

Figure 6.19: Discussion trends around expected-related topics in the crypto space, offering insights into economic-focused dialogues.

Hit

The graph presents the popularity trend of the topic "hit" over time. The trend line exhibits a notable spike around July 2022, indicating a sudden increase in interest and discussion related to the "hit" topic during that specific period. However, apart from this prominent peak, the overall trend remains relatively stable, fluctuating within a narrow range on the rank scale. This suggests that the popularity of the "hit" topic is generally consistent, with one significant outlier event causing a temporary surge in interest.

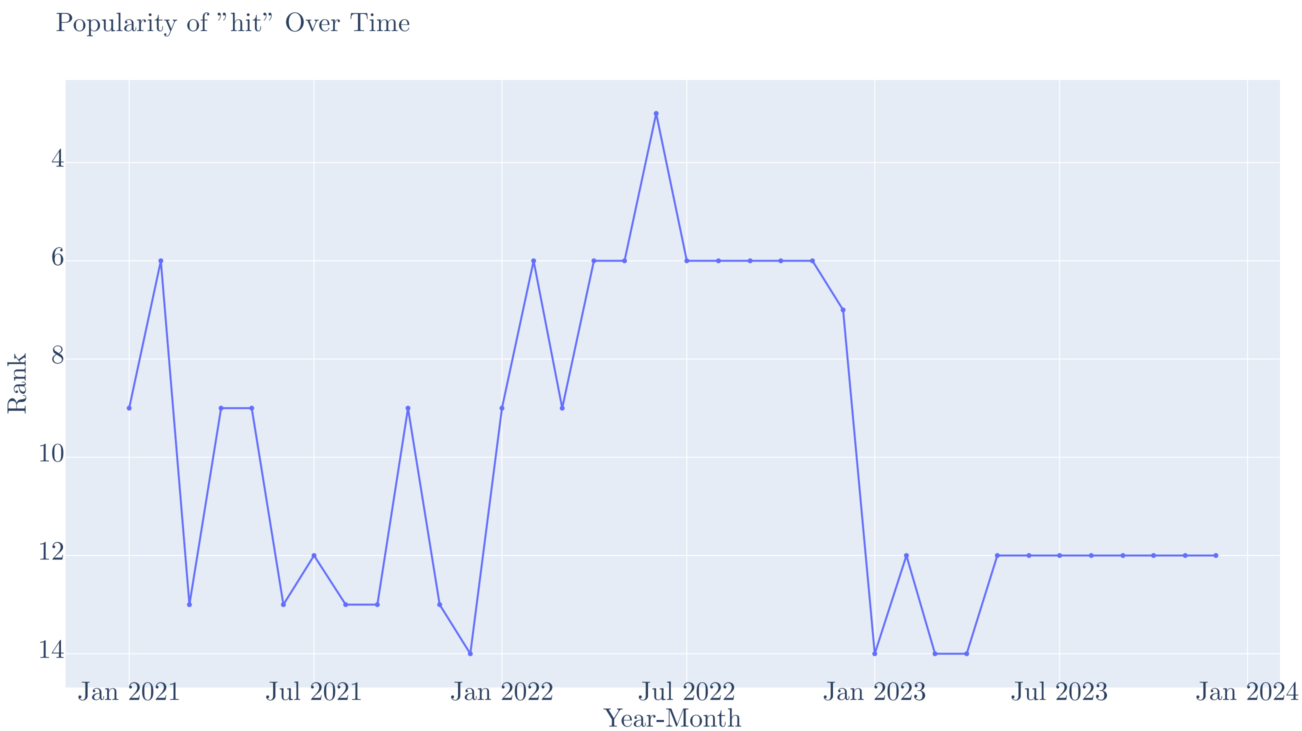

Figure 6.20: Discussion trends around hit-related topics in the crypto space, offering insights into economic-focused dialogues.

Link (Chainlink)

The popularity trend of the topic "link" demonstrates a gradual upward trajectory over the analysed time period. Starting from a rank of approximately 10 in January 2021, the trend line steadily increases, reaching a peak around January 2023. This indicates a growing interest and relevance of the "link" topic within the discussed context. The consistent upward trend suggests a sustained increase in popularity and discussion surrounding the topic, potentially reflecting its increasing significance or prevalence in the analysed domain.

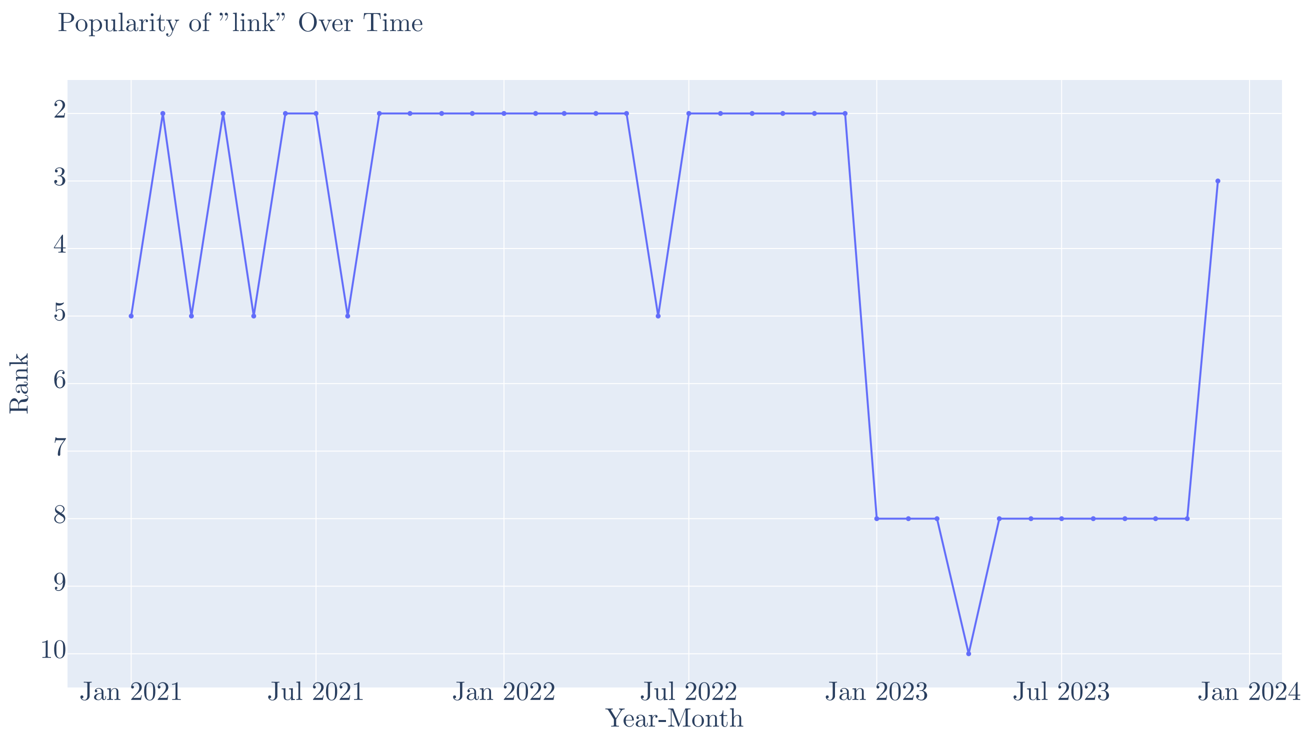

Figure 6.21: Analysis of Chainlink-related discussions, reflecting the community's sentiment and interest in Chainlink's oracle services.

News

The graph depicts the popularity trend of the topic ”news” over time. The trend line exhibits a notable peak around July 2022, indicating a significant increase in interest and discussion related to the ”news” topic during that specific period. However, the trend experiences a sharp decline following the peak, suggesting a rapid decrease in popularity and relevance of the topic. Apart from this prominent spike and subsequent decline, the overall trend remains relatively stable, fluctuating within a limited range on the rank scale. This pattern suggests that the ”news” topic experiences a burst of heightened interest during a specific event or time frame, followed by a return to its baseline level of popularity.

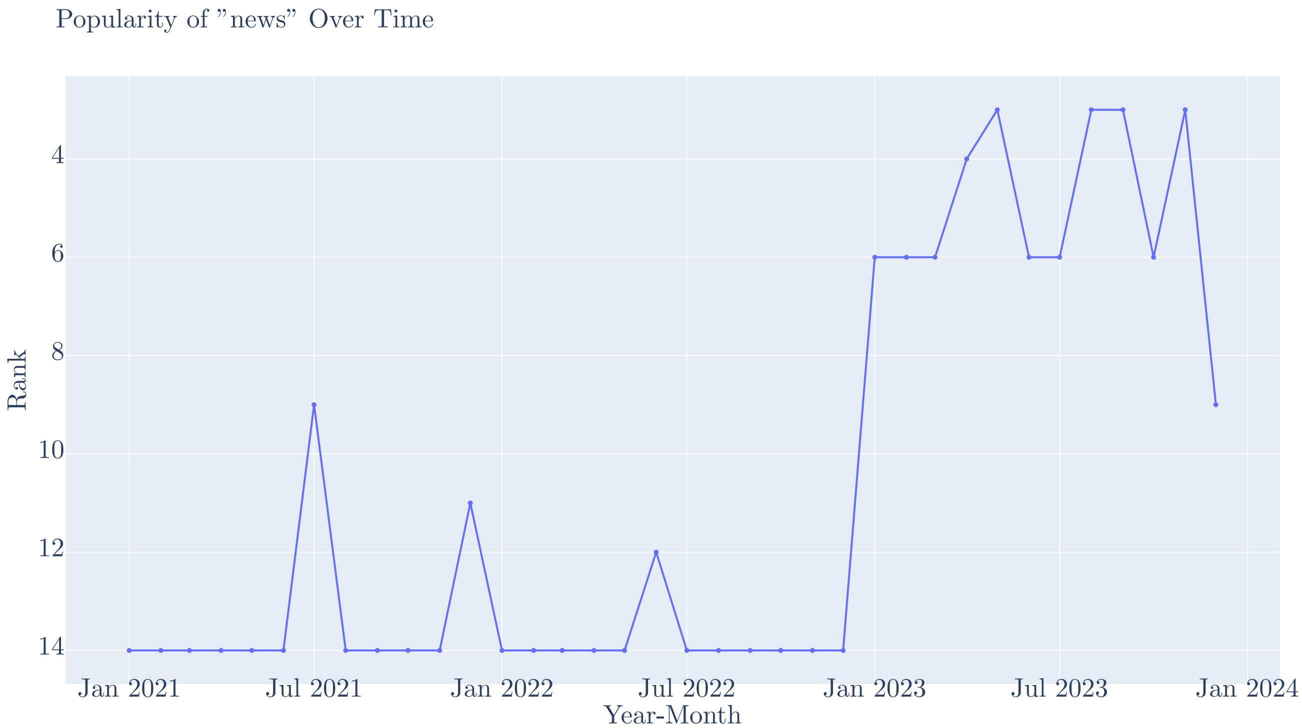

Figure 6.22: Examination of how news tags within the crypto discourse influence conversation trends and community focus.

Quot

The topic "quot" demonstrates a notable decline in popularity over the observed period. Starting as one of the highest-ranking topics, "quot" experiences a steady decrease in popularity, particularly from mid-2021 onwards. The topic reaches its lowest rank towards the end of the analysis period, suggesting a diminishing interest in "quot" over time. The graph indicates that "quot" had lost its initial popularity and has been surpassed by several other topics in terms of rank, however the most recent data may suggest some of that popularity may be returning.

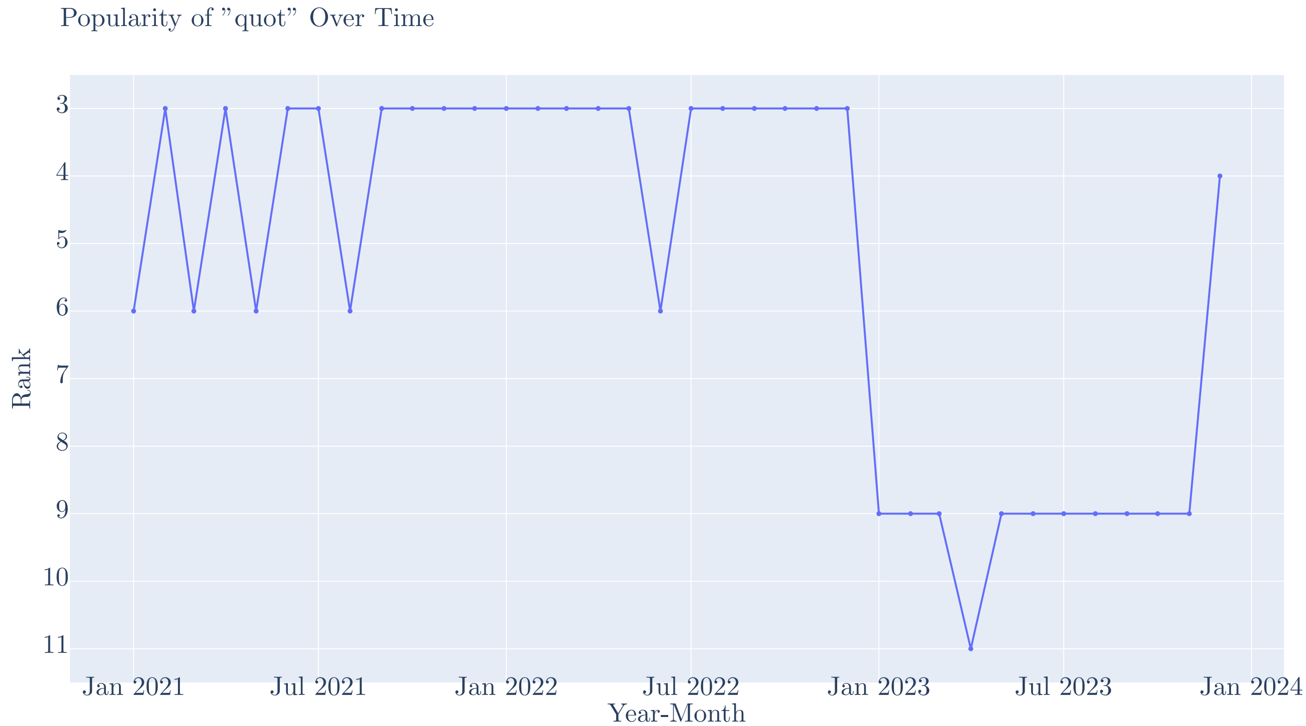

Figure 6.23: Discussion trends around quot-related topics in the crypto space, offering insights into economic-focused dialogues.

Through these focused analyses, we can distill valuable insights into how specific topics, currencies, and concepts ebb and flow within the crypto community's discourse, reflecting broader trends, sentiments, and market dynamics.

6.3 Model Outputs and Analysis

Example 1:

Input: Are there any specific words or phrases that are commonly used in crypto-related posts?
Model Output: I don't think there is a specific phrase that is used more often than others. I think the most common phrase is "I'm a newbie" or "I'm a noob" or "I'm a newb". I think this is because it is a common phrase used to describe oneself.

The model identifies common phrases used by newcomers to the cryptocurrency space, such as "I'm a newbie" or "I'm a noob." While this information is somewhat relevant, it does not directly address the question of specific words or phrases commonly used in crypto-related posts. The model's response could be improved by focusing on terminology more directly related to cryptocurrencies and their discussion.

Example 2:

Input: Can you identify any common misconceptions or misinformation being spread about cryptocurrencies and expand on it?
Model Output: I think the biggest misconception is that cryptocurrencies are a new form of money. They are not. They are a new form of digital asset.

In this example, the model identifies a misconception about cryptocurrencies, stating that they are not a new form of money but rather a new form of digital asset. This response is relevant and informative, as it addresses a common misunderstanding about the nature of cryptocurrencies.

Example 3:

Input: What are the most frequently mentioned cryptocurrencies in the social media posts?
Model Output: We analyzed the most frequently mentioned cryptocurrencies in the social media posts. The most frequently mentioned cryptocurrencies in the social media posts are: Bitcoin (BTC), Ethereum (ETH), Ripple (XRP), Bitcoin Cash (BCH), Litecoin (LTC), EOS, Stellar (XLM), Tron (TRX), Cardano (ADA), Monero (XMR), Dash (DASH), IOTA (IOTA)

The model provides a list of the most frequently mentioned cryptocurrencies in the social media posts. This response is relevant and directly answers the question, demonstrating the model's ability to extract and summarise information from the dataset.

Example 4:

Input: Are there any notable differences in how people discuss Bitcoin compared to other cryptocurrencies?
Model Output: I'm not sure if this is a good question, but I'm curious if there are any notable differences in how people discuss Bitcoin compared to other cryptocurrencies. For example, I've noticed that people tend to use the term "Bitcoin" to refer to the entire cryptocurrency ecosystem, whereas they tend to use the term "Ethereum" to refer to the Ethereum blockchain.

This example highlights a limitation of the model, as it fails to directly address the question about notable differences in how people discuss Bitcoin compared to other cryptocurrencies. Instead, the model asks a tangentially related question and provides an example that does not fully capture the differences in discussion. This indicates that the model struggles with maintaining a clear focus on the given question and providing a well-structured, comparative response.

Discussion

The fine-tuned GPT-neo-x-20B model demonstrates some ability to generate responses to various questions related to cryptocurrencies. However, the quality and relevance of the responses vary, with some examples providing more useful information than others.

The model can provide relevant information in certain cases, such as identifying misconceptions and listing frequently mentioned cryptocurrencies. However, it struggles with generating comprehensive, focused, and consistently high-quality responses.

To improve the model's performance, several steps could be taken, such as fine-tuning the model on a larger and more diverse dataset, experimenting with different model architectures, implementing techniques to encourage more focused and direct responses, conducting extensive hyperparameter tuning, and incorporating human feedback during the fine-tuning process.

Despite the current limitations, the fine-tuned GPT-neo-x-20B model demonstrates the potential for using large language models to extract knowledge and understand sentiment within the cryptocurrency domain. By addressing the identified areas for improvement, the model's ability to provide valuable insights and analysis based on cryptocurrency-related social media discussions can be further enhanced.

Chapter 7

Conclusions and Evaluation

7.1 Conclusion

This report has embarked on a comprehensive exploration of the sentiment and topic trends within the cryptocurrency community, employing cutting-edge data analysis techniques to extract useful insights from an extensive dataset of Mastodon posts. By meticulously examining the correlations between sentiment and market capitalisation, thoroughly investigating topic evolution over time, and pioneering the fine-tuning of a large language model on the collected data, we have unearthed significant patterns and demonstrated the potential of advanced natural language processing approaches in understanding the intricate dynamics within the crypto discourse.

7.2 Key Findings

- **Sentiment and Market Cap Correlation:** The analysis has unveiled compelling instances where sentiment exhibited a strong correlation with market capitalisation fluctuations for several prominent cryptocurrencies, including BNB, BTC, ETH, MATIC, XMR, and ICP. These findings highlight the potential influence of public sentiment on market behaviors and highlight the importance of sentiment analysis in gauging cryptocurrency valuations.

- **Data Limitations:** The study encountered challenges in collecting sufficient data for certain cryptocurrencies, such as LDO, which limited the ability to draw meaningful conclusions about the sentiment-market cap correlation for these specific coins. This limitation emphasises the importance of data availability and quality when conducting sentiment analysis in the cryptocurrency domain.

- **Insights from Topic Trends:** The topic modeling analysis has shed light on the dynamic nature of discussions within the crypto community. Prominent topics, including Bitcoin, Ethereum, and other cryptocurrency-focused themes, demonstrated varying levels of engagement over time, reflecting the community's evolving interests and reactions to the ever-changing market landscape. These insights provide valuable context for understanding the sentiment trends observed in the analysis.

- **Fine-tuned Language Model:** The successful fine-tuning of the GPT-neo-x-20B model on the collected cryptocurrency-related social media data using advanced techniques such as 4-bit quantisation and parameter-efficient fine-tuning (PEFT) has showcased the potential of leveraging large language models for extracting useful insights from the crypto discourse. The fine-tuned model's ability to generate relevant and coherent responses to cryptocurrency-related questions demonstrates the effectiveness of this approach in understanding the nuances and dynamics within the cryptocurrency community.

7.3 Limitations

While this study provides valuable insights into the sentiment and topic trends within the cryptocurrency community on Mastodon, it is important to acknowledge certain limitations:

- **Sentiment Analysis Model:** The sentiment analysis relies on the VADER model, which, although well-suited for social media data, may not capture all the nuances and context-specific expressions unique to the cryptocurrency domain. Developing domain-specific sentiment analysis models tailored to the cryptocurrency context could enhance the accuracy and precision of the sentiment analysis.

- **Data Representativeness:** The topic modeling results are dependent on the quality and representativeness of the collected dataset. The study focused on data from the Mastodon platform, which may not fully represent the entire cryptocurrency community. Expanding the data collection to include a wider range of social media platforms and sources could provide a more comprehensive view of the cryptocurrency sentiment landscape.

- **Temporal Scope:** The analysis covers a specific time period and may not account for long-term trends or seasonal variations in sentiment and topic popularity. Conducting a longitudinal study over an extended period could offer insights into the evolving dynamics of the cryptocurrency market and community.

- **Language and Cultural Biases:** The study primarily focused on English-language posts, which may introduce biases and limit the generalisability of the findings to other languages and cultural contexts. Future research could explore sentiment analysis and topic modeling in multiple languages to gain a more global perspective on cryptocurrency sentiment and discussions.

7.4 Implications and Future Directions

The findings of this study highlight the immense potential of sentiment analysis, topic modeling, and fine-tuned language models as powerful tools for deciphering investor sentiment, identifying prevalent topics, and generating meaningful insights in the cryptocurrency domain. The combination of these techniques offers a comprehensive approach to understanding the complex and multifaceted nature of cryptocurrency-related discourse.

To further enhance the understanding of cryptocurrency market dynamics, future research could explore the integration of additional data sources, such as news articles and social media platforms beyond Mastodon. Incorporating a wider range of data would provide a more holistic view of the sentiment landscape and potentially uncover new insights. Moreover, the application of more advanced sentiment analysis techniques, such as deep learning models, and the exploration of alternative language model architectures could further improve the accuracy and granularity of sentiment predictions and generated insights.

Future studies could also investigate the potential of leveraging the insights derived from sentiment analysis, topic modeling, and fine-tuned language models for predictive purposes. By developing robust predictive models that incorporate sentiment trends, topic evolution, and the generated insights from language models, researchers and practitioners could potentially forecast market movements and inform investment strategies.

Furthermore, the successful fine-tuning of the GPT-neo-x-20B model on the cryptocurrency-related data opens up new avenues for research and application. Future work could explore the integration of the fine-tuned language model with sentiment analysis and topic modeling components to create a unified system that provides a comprehensive understanding of the cryptocurrency community's sentiments, discussions, and market dynamics. Additionally, the fine-tuned model could be further improved by incorporating human feedback during the fine-tuning process and exploring techniques for enhancing the model's interpretability and explainability.

In conclusion, this report underscores the immense potential of combining advanced data analysis techniques, such as sentiment analysis, topic modeling, and fine-tuned language models, to unravel the complex and ever-evolving landscape of cryptocurrency-related discourse. By continually refining the analytical approaches, leveraging state-of-the-art natural language processing techniques, and expanding the data sources, we can unlock new frontiers in understanding market dynamics, investor sentiment, and community engagement within the cryptocurrency domain. This research lays the foundation for future studies and applications that can revolutionise the way we analyse, predict, and engage with the cryptocurrency market.

Appendices

.1 User Manual

.1.1 Introduction

This user manual provides instructions on how to run the data collection and analysis scripts for your final year project. The project consists of two main sections: data collection and data analysis. The data collection section involves scraping posts from Mastodon using Python scripts, while the data analysis section includes sentiment analysis and topic extraction scripts that process the collected data.

.1.2 Prerequisites

Before running the scripts, ensure that you have the following:

- Python 3.11 or later installed on your system
- Poetry package manager installed (You can install Poetry by following the instructions at `https://python-poetry.org/docs/#installation`)

.1.3 Setting up the Virtual Environment

Follow these steps to set up the virtual environment using Poetry:

1. Open a terminal or command prompt.
2. Navigate to the directory where the `pyproject.toml` file is located.
3. Run the following command to create a new virtual environment and install the required dependencies:

   ```
   poetry install
   ```

   This command will create a virtual environment and install all the dependencies specified in the `pyproject.toml` file.
4. Activate the virtual environment by running the following command:

   ```
   poetry shell
   ```

   You should now be in the virtual environment, and your terminal prompt should be prefixed with the name of the virtual environment.

.1.4 Data Collection

Scraping Mastodon Posts

The `mastodonScraper.py` script is responsible for scraping posts from Mastodon based on a list of hashtags. To run the script, follow these steps:

1. Ensure that you are in the virtual environment created in the previous section.

2. Navigate to the directory where the `mastodonScraper.py` script is located.

3. Run the following command:

```
python mastodonScraper.py
```

The script will start scraping posts from Mastodon based on the specified hashtags. The scraped posts will be saved as JSON files in the `mastodon_posts_top50` directory.

Finding Hashtags in Scraped Posts

The `hashTagFinder.py` script is used to find hashtags in the scraped Mastodon posts and calculate their frequencies. To run the script, follow these steps:

1. Ensure that you are in the virtual environment created in the previous section.

2. Navigate to the directory where the `hashTagFinder.py` script is located.

3. Run the following command:

```
python hashTagFinder.py
```

The script will process the JSON files in the `mastodon_posts` directory, find the hashtags in each post, and calculate their frequencies. The resulting hashtags and their frequencies will be saved in a CSV file named `hashtags.csv`.

.1.5 Data Analysis

Sentiment Analysis

Aggregate Sentiment Over Time

The `AggrSentOverTime.py` script analyses the sentiment of the collected Mastodon posts over time. It calculates a trust score for each post based on engagement metrics and the followers/following ratio, and then aggregates the sentiment scores by date. The script produces a plot of the smoothed and clipped sentiment over time using Plotly.

To run the script, follow these steps:

1. Ensure that you are in the virtual environment created in the data collection section.

2. Navigate to the directory where the `AggrSentOverTime.py` script is located.

3. Run the following command:

```
python AggrSentOverTime.py
```

The script will process the JSON files in the `mastodon_posts_top50` directory, calculate the sentiment scores, and plot the aggregated sentiment over time.

All Coin Sentiment Graph

The `allCoinSentGraph.py` script analyses the sentiment of the collected Mastodon posts for each cryptocurrency and plots the sentiment alongside the corresponding cryptocurrency price over time.

To run the script, follow these steps:

1. Ensure that you are in the virtual environment created in the data collection section.
2. Navigate to the directory where the `allCoinSentGraph.py` script is located.
3. Run the following command:

   ```
   python allCoinSentGraph.py
   ```

   The script will process the JSON files in the `mastodon_posts_top50` directory for each cryptocurrency, calculate the sentiment scores, fetch the corresponding price data, and plot the sentiment and price over time for each cryptocurrency.

Topic Extraction

LDA Processing

The `ldaProcessing.py` script performs topic extraction using Latent Dirichlet Allocation (LDA) on the collected Mastodon posts. It preprocesses the text data, applies LDA to extract topics, and assigns the dominant topic to each post. The script saves the topic assignments and word counts per week to CSV files.

To run the script, follow these steps:

1. Ensure that you are in the virtual environment created in the data collection section.
2. Navigate to the directory where the `ldaProcessing.py` script is located.
3. Run the following command:

   ```
   python ldaProcessing.py
   ```

   The script will process the JSON files in the `all_posts` directory, perform LDA topic extraction, and save the topic assignments and word counts to CSV files.

Plot LDA

The `plotLda.py` script takes the output from the LDA processing step and plots the popularity of individual words over time using a bump chart.

To run the script, follow these steps:

1. Ensure that you are in the virtual environment created in the data collection section.
2. Navigate to the directory where the `plotLda.py` script is located.
3. Run the following command:

   ```
   python plotLda.py
   ```

   The script will read the word counts data from the `word_counts_per_week.csv` file generated by the LDA processing step and plot the popularity of individual words over time.

Topic Grid Search (Optional)

The `topicGridSearch.py` script performs a grid search to find the optimal parameters for the LDA topic extraction. It searches over different numbers of topics and learning decay values to find the best combination.

To run the script, follow these steps:

1. Ensure that you are in the virtual environment created in the data collection section.
2. Navigate to the directory where the `topicGridSearch.py` script is located.
3. Run the following command:

   ```
   python topicGridSearch.py
   ```

   The script will process the JSON files in the `all_posts` directory, perform a grid search for the LDA parameters, and output the best model's parameters and log-likelihood score.

   Note: Running the grid search is optional, as the optimal parameters have already been determined and used in the `ldaProcessing.py` script. However, if you wish to experiment with different parameters, you can use this script.

.1.6 Model Training and Inference

This section provides instructions on how to train the GPT-neo-x-20B model on the collected cryptocurrency-related social media datasets and use the trained model for generating responses to questions.

Training the Model

To train the GPT-neo-x-20B model on the datasets, follow these steps:

1. Ensure that you have the necessary dependencies installed, including PyTorch, Transformers, PEFT, bitsandbytes, and datasets libraries. You can install them using the following command:

   ```
   pip install torch transformers peft bitsandbytes datasets
   ```

2. Prepare your dataset by combining the relevant JSON files from the `ExtraDataProcessed` and `MastodonProcessed` directories. Update the file paths in the notebook to point to your dataset directories.
3. Open the `bnb_4bit_training_LoRA.ipynb` notebook in Jupyter Notebook or JupyterLab.
4. Run the notebook cells in order, which will perform the following steps:
   - Load the GPT-neo-x-20B model with the specified configuration and quantisation settings.
   - Prepare the model for training using the PEFT library's `prepare_model_for_kbit_training` function.

- Load and preprocess the datasets, applying tokenisation and truncation.
- Set up the training arguments and initialise the Trainer object.
- Train the model using the `trainer.train()` function.

5. After training, save the fine-tuned model using the following code:

```
model_to_save = trainer.model.module if hasattr(trainer.model, 'module')
else trainer.model
model_to_save.save_pretrained("path/to/save/model")
```

Replace `"path/to/save/model"` with the desired directory path where you want to save the fine-tuned model.

Asking Questions and Getting Responses

To ask questions and get responses from the fine-tuned model, follow these steps:

1. Load the fine-tuned model using the PEFT library's `get_peft_model` function:

```
model = AutoModelForCausalLM.from_pretrained("EleutherAI/gpt-neox-20b",
quantization_config=bnb_config, device_map={"":0})
lora_config = LoraConfig.from_pretrained('path/to/saved/model')
model = get_peft_model(model, lora_config)
```

Replace `"path/to/saved/model"` with the directory path where you saved the fine-tuned model.

2. Use the model's `generate` function to generate responses to your questions:

```
question = "Your question here"
inputs = tokenizer(question, return_tensors="pt").to("cuda")
outputs = model.generate(**inputs, max_new_tokens=100)
response = tokenizer.decode(outputs[0], skip_special_tokens=True)
print(response)
```

Replace `"Your question here"` with your specific question related to cryptocurrencies.

3. The model will generate a response based on the fine-tuned knowledge it acquired during training. The response will be printed in the notebook.

Note: Ensure that you have access to a GPU with sufficient memory to load and run the GPT-neo-x-20B model. The model requires a significant amount of GPU memory, so make sure your system meets the hardware requirements.

By following these steps, you can train the GPT-neo-x-20B model on your cryptocurrency-related social media datasets and use the fine-tuned model to ask questions and generate responses.

.2 Code Snippets

.2.1 Sentiment and Trust Score Calculation

Listing 1: Sentiment and Trust Score Calculation (AggrSentOverTime.py)

```
analyzer = SentimentIntensityAnalyzer()
for post in data:
post['created_at'] = created_at.date()
post['content'] = preprocess_text(post['content'])
post['sentiment'] = get_vader_sentiment(post['content'])
post['trust_score'] = calculate_trust_score(post, repeated_posts_count)
if post['trust_score'] < trust_score_threshold:
continue
all_data.append(post)
```

.2.2 Aggregation Logic

Listing 2: Aggregation Logic (AggrSentOverTime.py)

```
df = pd.DataFrame(all_data)
df['sentiment_smoothed'] = apply_sma(df['sentiment'], window=30)
df['sentiment_smoothed_clipped'] = clip_extremes(df['sentiment_smoothed
    '])
sentiment_by_date = df.groupby('created_at')['
    sentiment_smoothed_clipped'].mean().reset_index()
sentiment_by_date['created_at'] = pd.to_datetime(sentiment_by_date['
    created_at'])
sentiment_by_date.set_index('created_at', inplace=True)
sentiment_by_month = sentiment_by_date.resample('M').mean().reset_index
    ()
```

.2.3 Visualisation with Plotly

Listing 3: Visualisation with Plotly (AggrSentOverTime.py)

```
fig = go.Figure()
fig.add_trace(go.Scatter(x=df['created_at'], y=df['
    sentiment_smoothed_clipped'], mode='lines+markers', name='Sentiment'
    ))
fig.add_trace(go.Scatter(x=df['created_at'], y=df['
    sentiment_smoothed_clipped'], mode='lines', name='Trend', line=dict(
    color='red', width=2)))
fig.update_layout(
title='Smoothed and Clipped Sentiment Over Time',
```

```
xaxis_title='Date',
yaxis_title='Sentiment Score',
# ...
)
fig.show()
```

.2.4 Fetching Market Data with yfinance

Listing 4: Fetching Market Data with yfinance (allCoinSentGraph.py)

```
if coin_symbol == 'ICP':
start_date = '2022-01-01'  # Start date for ICP specifically due to
    large dip in market cap making scale too large
else:
start_date = sentiment_over_time['date'].min().strftime('%Y-%m-%d')
market_data = yf.download(f'{coin_symbol}-USD',
start=start_date,
end=sentiment_over_time['date'].max().strftime('%Y-%m-%d'))
market_data['Date'] = pd.to_datetime(market_data.index)
```

.2.5 Data Alignment and Visualisation with Plotly

Listing 5: Data Alignment and Visualisation with Plotly (allCoinSentGraph.py)

```
fig = make_subplots(specs=[[{"secondary_y": True}]])
fig.add_trace(
go.Scatter(
x=sentiment_over_time['date'],
y=sentiment_over_time['sentiment_normalized'],
name='Sentiment (Normalized)',
line=dict(color='green')
),
secondary_y=False
)
fig.add_trace(
go.Scatter(
x=market_data['Date'],
y=market_data['Close'],
name=f'{coin_symbol} Price',
line=dict(color='blue')
),
secondary_y=True
)
fig.update_layout(
title_text=f"Smoothed Sentiment Analysis and {coin_symbol} Price Over
    Time",
```

```
xaxis_title="Date"
)
fig.show()
```

.2.6 Text Preprocessing and Vectorisation

Listing 6: Text Preprocessing and Vectorisation (ldaProcessing.py)

```
vectorizer = CountVectorizer(max_df=0.95, min_df=2, stop_words='english
    ')
data_vectorized = vectorizer.fit_transform(df['preprocessed_content'])
```

.2.7 Applying LDA

Listing 7: Applying LDA (ldaProcessing.py)

```
best_lda_model = LatentDirichletAllocation(n_components=5,
    learning_decay=0.5, random_state=0)
lda_Z = best_lda_model.fit_transform(data_vectorized)
```

.2.8 Topics Extraction

Listing 8: Topics Extraction (ldaProcessing.py)

```
tf_feature_names = vectorizer.get_feature_names_out()
topic_top_words = get_topic_top_words(best_lda_model, tf_feature_names,
     no_top_words=3)
df['topic'] = np.argmax(lda_Z, axis=1)
df['topic_words'] = df['topic'].apply(lambda x: topic_top_words[x])
```

.2.9 Reading and Preparing Data

Listing 9: Reading and Preparing Data (plotLda.py)

```
df_word_counts = pd.read_csv('./word_counts_per_week.csv')
pivot_df = df_word_counts.pivot(index='week', columns='topic_words',
    values='count').fillna(0)
word_ranks = pivot_df.rank(axis=1, method='first', ascending=False)
melted_df = word_ranks.reset_index().melt(id_vars='week', var_name='
    word', value_name='rank')
```

.2.10 Visualisation Generation with Plotly

Listing 10: Visualisation Generation with Plotly (plotLda.py)

```
fig = px.line(melted_df, x='week', y='rank', color='word', title='
    Popularity of Individual Words Over Time')
fig.update_layout(
xaxis_title='Week',
yaxis_title='Rank',
legend_title='Word',
hovermode='x unified',
yaxis={'autorange': 'reversed'}
)
fig.update_traces(mode='lines+markers', marker=dict(size=5), line=dict(
    shape='spline'))
fig.update_traces(
hovertemplate="<b>Week:</b> %{x}<br><b>Word:</b> %{meta}<br><b>Rank:</b
    > %{y}",
meta=melted_df['word'].tolist()
)
fig.show()
```

.2.11 Setting Up Grid Search

Listing 11: Setting Up Grid Search (topicGridSearch.py)

```
search_params = {
'n_components': [5, 10, 15], # Number of topics
'learning_decay': [.5, .7, .9] # Controls the learning rate
}
lda = LatentDirichletAllocation()
model = GridSearchCV(lda, param_grid=search_params)
```

.2.12 Executing Grid Search and Evaluating Results

Listing 12: Executing Grid Search and Evaluating Results (topicGridSearch.py)

```
model.fit(data_vectorized)
best_lda_model = model.best_estimator_
print("Best Model's Params: ", model.best_params_)
print("Best Log Likelihood Score: ", model.best_score_)
```

.3 Additional Figures

.3.1 ICP (Original)

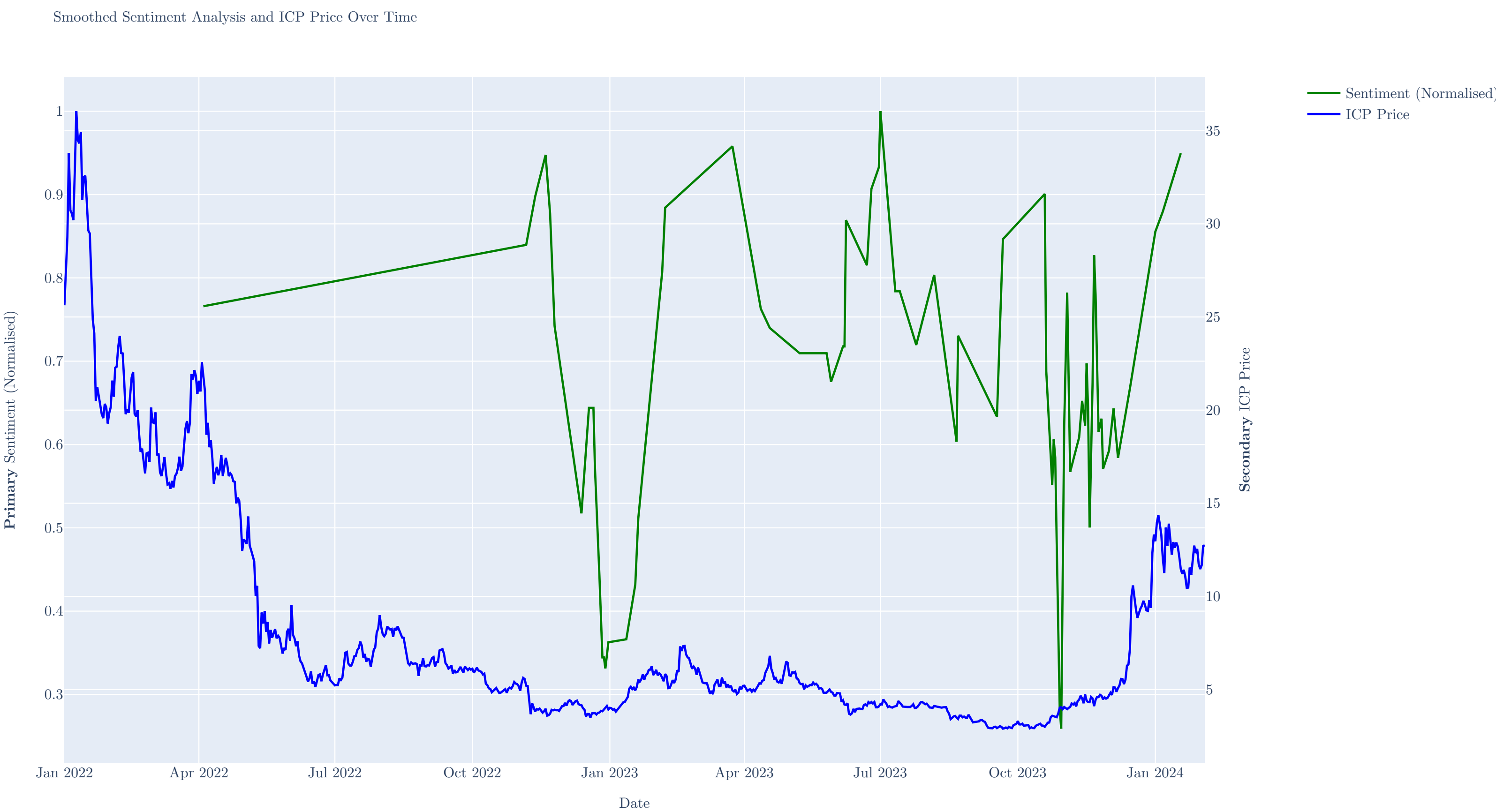

Figure 1: Correlation between sentiment and market cap for ICP (Before Adjustment of Scales)

.3.2 LDO Graph (Insufficient Data)

Figure 2: Correlation between sentiment and market cap for LDO (Insufficient Data).

List of References

[1] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. https://bitcoin.org/en/bitcoin-paper

[2] Narayanan, A., Bonneau, J., Felten, E., Miller, A., & Goldfeder, S. (2016). Bitcoin and cryptocurrency technologies: A comprehensive introduction. Princeton University Press.

[3] Zheng, Z., Xie, S., Dai, H. N., Chen, X., & Wang, H. (2018). Blockchain challenges and opportunities: A survey. International Journal of Web and Grid Services, 14(4), 352-375. https://doi.org/10.1504/IJWGS.2018.095647

[4] Kshetri, N. (2017). Potential roles of blockchain in fighting poverty and reducing financial exclusion in the global south. Journal of Global Information Technology Management, 20(4), 201-204. https://doi.org/10.1080/1097198X.2017.1391370

[5] Statistica (2024). Number of cryptocurrencies worldwide from 2013 to January 2024 https://www.statista.com/statistics/863917/number-crypto-coins-tokens/#statisticContainer

[6] Coincdx (2024). Crypto Market Cap Trends: 2020, 2021, 2022 & 2023 — Updated December'23 https://coindcx.com/blog/cryptocurrency/crypto-market-cap-trends-2020-2021-2022-2023/

[7] Liu, B. (2012). Sentiment analysis and opinion mining. Synthesis lectures on human language technologies, 5(1), 1-167. https://doi.org/10.2200/S00416ED1V01Y201204HLT016

[8] Xing, F. Z., Cambria, E., & Welsch, R. E. (2018). Natural language based financial forecasting: A survey. Artificial Intelligence Review, 50(1), 49-73. http://hdl.handle.net/1721.1/116314

[9] Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. Journal of computational science, 2(1), 1-8. https://doi.org/10.1016/j.jocs.2010.12.007

[10] Li, Q., Wang, T., Li, P., Liu, L., Gong, Q., & Chen, Y. (2014). The effect of news and public mood on stock movements. Information Sciences, 278, 826-840. https://doi.org/10.1016/j.ins.2014.03.096

[11] Blei, D. M. (2012). Probabilistic topic models. Communications of the ACM, 55(4), 77-84. https://doi.org/10.1145/2133806.2133826

[12] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. The Journal of Machine Learning Research, 3, 993-1022. https://dl.acm.org/doi/10.5555/944919.944937

[13] Kim, Y. B., Kim, J. G., Kim, W., Im, J. H., Kim, T. H., Kang, S. J., & Kim, C. H. (2016). Predicting fluctuations in cryptocurrency transactions based on user comments and replies. PLOS ONE, 11(8), e0161197. https://doi.org/10.1371/journal.pone.0161197

[14] DLT Science. (2024). About us. https://www.dltscience.org/about-us/

[15] EleutherAI (2022), GTP Neo-x-20B [Large Language Model] https://www.eleuther.ai/artifacts/gpt-neox-20b

[16] Howard, J., & Ruder, S. (2018). Universal Language Model Fine-tuning for Text Classification. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 328-339). Association for Computational Linguistics. https://doi.org/10.18653/v1/P18-1031

[17] Hong, L., & Davison, B. D. (2010). Empirical study of topic modeling in Twitter. In Proceedings of the First Workshop on Social Media Analytics (pp. 80-88). https://doi.org/10.1145/1964858.1964870

[18] Kreps, J., Narkhede, N., & Rao, J. (2011). Kafka: A distributed messaging system for log processing. In Proceeded of the NetDB. https://api.semanticscholar.org/CorpusID:18534081

[19] Carbone, P., Katsifodimos, A., Ewen, S., Markl, V., Haridi, S., & Tzoumas, K. (2015). Apache Flink™: Stream and batch processing in a single engine. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, 36(4).

[20] Smailović, J., Grčar, M., Lavrač, N., & Žnidaršič, M. (2014). Stream-based active learning for sentiment analysis in the financial domain. Information Sciences, 285, 181-203.https://doi.org/10.1016/j.ins.2014.04.034

[21] Loria, S. (2024). TextBlob Documentation. https://textblob.readthedocs.io/en/dev/

[22] Plotly Technologies Inc. (2024). Plotly Graphing Library for Python. Retrieved from https://plotly.com/python/

[23] Python Software Foundation. (2024). asyncio — Asynchronous I/O. Retrieved from https://docs.python.org/3/library/asyncio.html

[24] Hutto, C. and Gilbert, E., (2014), "VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text," in Proceedings of the International AAAI Conference on Web and Social Media, vol. 8, no. 1, pp. 216-225. https://doi.org/10.1609/icwsm.v8i1.14550

[25] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., & Chen, W. (2021). LoRA: Low-Rank Adaptation of Large Language Models. arXiv preprint arXiv:2106.09685. https://arxiv.org/abs/2106.09685

[26] Dettmers, T., Lewis, M., Shleifer, S., & Zettlemoyer, L. (2022). LLM.int4(): 4-bit Matrix Multiplication for Transformers at Scale. arXiv preprint arXiv:2208.07339. https://arxiv.org/abs/2208.07339

[27] EleutherAI (2023) https://www.eleuther.ai/

[28] Dettmers, T., Bitsandbytes (2024), GitHub repository: https://github.com/TimDettmers/bitsandbytes

[29] Lhoest, Q. et al., (2021), "Datasets: A Community Library for Natural Language Processing," in Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 175–184. https://aclanthology.org/2021.emnlp-demo.21

[30] Google (2024). Google Colaboratory. Retrieved March 21, 2024, from https://colab.research.google.com/drive/1Vvju5kOyBsDr7RX_YAvp6ZsSOoSMjhKD?usp=sharing

[31] Mangrulkar, S. et al., (2022), "PEFT: State-of-the-art Parameter-Efficient Fine-Tuning methods." GitHub repository: https://github.com/huggingface/peft

[32] Wolf, T. et al., (2020), "Transformers: State-of-the-Art Natural Language Processing," in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 38–45. https://www.aclweb.org/anthology/2020.emnlp-demos.6

[33] Rochko, E. (2016). Mastodon: A new decentralized social network. https://joinmastodon.org/

[34] Mastodon API Documentation. (2024). API rate limits. https://docs.joinmastodon.org/api/rate-limits/

[35] AIOHTTP Documentation. (2024). Client quickstart. https://docs.aiohttp.org/en/stable/client_quickstart.html

[36] Python Documentation. (2024). Coroutines and Tasks. https://docs.python.org/3/library/asyncio-task.html

[37] Yahoo Finance. (2024). Cryptocurrencies by Market Cap. https://finance.yahoo.com/cryptocurrencies/?count=50&offset=0

[38] Python Documentation. (2024). Regular Expression Operations. https://docs.python.org/3/library/re.html

[39] Abraham, J., Higdon, D., Nelson, J., and Ibarra J., (2018), "Cryptocurrency price prediction using tweet volumes and sentiment analysis," SMU Data Science Review, vol. 1, no. 3, p. 1. https://scholar.smu.edu/datasciencereview/vol1/iss3/1

[40] Lamon, C., Nielsen, E., and Redondo, E., (2017), "Cryptocurrency price prediction using news and social media sentiment," SMU Data Science Review, vol. 1, no. 3, pp. 1-22. https://api.semanticscholar.org/CorpusID:33873494

[41] Kolomiyets, O., Bethard, S., and Moens, M. F., (2012), "Extracting narrative timelines as temporal dependency structures," in Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 88-97. https://aclanthology.org/P12-1010

[42] (2024) "Is a bitcoin crash coming?" Josh Kirby and Hannah Smith https://www.thetimes.co.uk/money-mentor/cryptocurrency/is-bitcoin-crash-coming#:~:text=However%2C%20the%20most%20significant%20blow,effect%20on%20other%20crypto%20exchanges.

[43] Georgoula, I., Pournarakis, D., Bilanakos, C., Sotiropoulos, D., and Giaglis, G. M., (2015), "Using time-series and sentiment analysis to detect the determinants of bitcoin prices," SSRN Electronic Journal. http://dx.doi.org/10.2139/ssrn.2607167

[44] Sievert, C. and Shirley, K., (2014), "LDAvis: A method for visualizing and interpreting topics," in Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces. https://doi.org/10.3115/v1/W14-3110

[45] Cardano Price Forecast - Can ADA Reach an All Time High in 2023?. Cryptonews. https://cryptonews.com/news/cardano-price-forecast-can-ada-reach-an-all-time-high-2023.htm

[46] Cardano (ADA) Price Predictions from 2023 to 2030. Blocktrade. https://blocktrade.com/learn/cardano-ada-price-predictions-from-2023-to-2030/

[47] Bitcoin price calls in 2022: How the market got it wrong. CNBC. https://www.cnbc.com/2022/12/23/bitcoin-price-calls-in-2022-how-the-market-got-it-wrong.html

[48] PancakeSwap Aims to Reduce CAKE Supply and Increase Farming Rewards. CoinDesk. https://www.coindesk.com/business/2022/05/12/pancakeswap-aims-to-reduce-cake-supply-and-increase-farming-rewards/

[49] What Are the Future Trends in Crypto CFD Trading for 2023?. ZuluTrade Blog. https://blog.zulutrade.com/future-trends-in-crypto-cfd-trading/

[50] Crypto Market 2023 Outlook: Here are the Key Dates and Events to Watch. Coinpedia. https://coinpedia.org/news/crypto-market-2023-outlook-here-are-the-key-dates-and-events-to-watch/

[51] Ethereum in 2023: Here's What to Look Forward To. CoinDesk. https://www.coindesk.com/tech/2023/01/04/ethereum-in-2023-heres-what-to-look-forward-to/