Data Scraping, Data Analysis, predictions model

Load the Data:

```
Read the CSV file into a DataFrame using a library like pandas.
```

Preprocess the Data:

```
Clean and preprocess the data as necessary (e.g., handling missing values, encoding categorical variables, etc.).
```

Feature Engineering:

```
Create features that will be useful for the model.
```

Split the Data:

```
Divide the data into training and testing sets.
```

Select a Model:

```
Choose a machine learning model appropriate for your prediction task (e.g., regression, classification).
```

Train the Model:

```
Fit the model to the training data.
```

Make Predictions:

```
Use the trained model to make predictions on the test set or new data.
```

Evaluate the Model:

```
Assess the model's performance using appropriate metrics.
```

1. Data Scraping

Data Collection Scrape or extract data from social media platforms.

Platforms:

Twitter (using Twitter API or scrapers like snscrape) Reddit (using Python's PRAW library or Pushshift API) Telegram (using libraries like Telethon or scraping tools) Key Data Points to Extract:

Text content (posts, tweets, comments) Timestamps User engagement (likes, shares, comments) Stock symbols or mentions (e.g., $AAPL, GME, etc.) Tools and Libraries:

tweepy, snscrape for Twitter PRAW, psaw for Reddit Telethon for Telegram Preprocessing Data:

Remove irrelevant posts (spam, off-topic). Normalize text (e.g., removing URLs, hashtags, special characters).

In [1]:
```python
import praw
import pandas as pd
import datetime

# Reddit API setup
reddit = praw.Reddit(
    client_id="LZTl2Q37jVYJc-aOMx8xpw",  # Replace with your Reddit app's client ID
    client_secret="tVdnlBFlpR6rxd9U5mpPmQ4dNG6P1Q",  # Replace with your Reddit app's client secret
    user_agent="StockScraper/1.0 by RiteshKumarYadav"  # Replace with your Reddit app's user agent
)

# Function to scrape stock market-related subreddits
def scrape_subreddit(subreddit_name, limit=100, keywords=None):
    subreddit = reddit.subreddit(subreddit_name)
    data = []

    for post in subreddit.new(limit=limit):
        if keywords:
            # Only scrape posts that match the keywords
            if any(keyword.lower() in post.title.lower() for keyword in keywords):
                data.append({
                    'title': post.title,
                    'text': post.selftext,
                    'score': post.score,
                    'created': datetime.datetime.fromtimestamp(post.created),
                    'url': post.url,
                    'comments': post.num_comments
                })
        else:
            data.append({
                'title': post.title,
                'text': post.selftext,
                'score': post.score,
                'created': datetime.datetime.fromtimestamp(post.created),
                'url': post.url,
                'comments': post.num_comments
            })

    return data

# Example: Scrape from subreddits related to stock market discussions
subreddits = ['stocks', 'WallStreetBets', 'investing']
keywords = ['stock', 'investment', 'market', 'trade', 'prediction']  # Example keywords for filtering

all_data = []
for sub in subreddits:
    print(f"Scraping {sub} subreddit...")
    scraped_data = scrape_subreddit(sub, limit=100, keywords=keywords)
    all_data.extend(scraped_data)

# Convert to DataFrame
df = pd.DataFrame(all_data)

# Save to CSV for further analysis
df.to_csv('stock_market_discussions.csv', index=False)

# Preview the data
print(df.head())
```

```
Scraping stocks subreddit...
Scraping WallStreetBets subreddit...
Scraping investing subreddit...
                                               title  \
0  /r/Stocks Weekend Discussion Saturday - Dec 07...
1  Should I talk my friend out of a $50k investme...
2         Pros and cons of this investment strategy?
3        These are the stocks on my watchlist (12/6)
4  r/Stocks Daily Discussion & Fundamentals Frida...

                                                text  score  \
0  This is the weekend edition of our stickied di...      3
1  First, me and my friend talk every few week ab...     16
2  I have Roth IRA with 150k in it and like every...      0
3  Hi! I am an ex-prop shop equity trader.\n\nThi...      1
4  This is the daily discussion, so anything stoc...     11

              created                                                url  \
0 2024-12-07 16:00:18  https://www.reddit.com/r/stocks/comments/1h8pr...
1 2024-12-06 23:22:55  https://www.reddit.com/r/stocks/comments/1h879...
2 2024-12-06 22:28:08  https://www.reddit.com/r/stocks/comments/1h85y...
3 2024-12-06 19:46:34  https://www.reddit.com/r/stocks/comments/1h82b...
4 2024-12-06 16:00:10  https://www.reddit.com/r/stocks/comments/1h7yg...

   comments
0         0
1        53
2        13
3        20
4       311
```

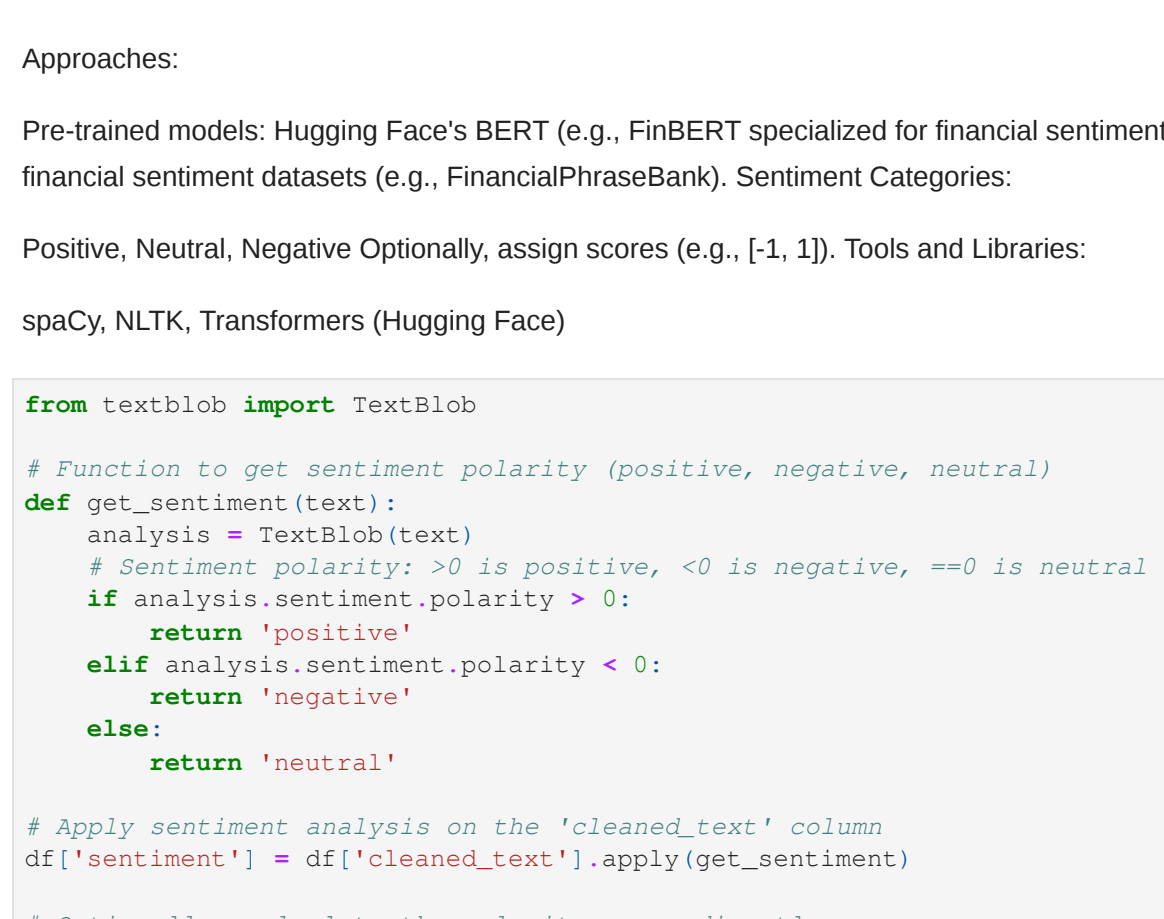
In [2]:
```python
#Data Cleaning and Preprocessing
import re
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from textblob import TextBlob

# Download NLTK data for stopwords
nltk.download('punkt')
nltk.download('stopwords')

# Function to clean and preprocess text
def clean_text(text):
    # Remove URLs
    text = re.sub(r'http\S+', '', text)
    # Remove non-alphanumeric characters
    text = re.sub(r'[^A-Za-z0-9 ]+', '', text)
    # Convert to lowercase
    text = text.lower()
    return text

# Function to remove stopwords
def remove_stopwords(text):
    stop_words = set(stopwords.words('english'))
    words = word_tokenize(text)
    return " ".join([word for word in words if word not in stop_words])

# Function for sentiment analysis using TextBlob
def get_sentiment(text):
    analysis = TextBlob(text)
    # Classify polarity: Positive (> 0), Negative (< 0), Neutral (0)
    if analysis.sentiment.polarity > 0:
        return 'positive'
    elif analysis.sentiment.polarity < 0:
        return 'negative'
    else:
        return 'neutral'

# Apply text cleaning to the 'title' and 'text' columns
df['cleaned_title'] = df['title'].apply(clean_text)
df['cleaned_text'] = df['text'].apply(clean_text)

# Apply stopword removal to 'cleaned_title' and 'cleaned_text'
df['cleaned_title'] = df['cleaned_title'].apply(remove_stopwords)
df['cleaned_text'] = df['cleaned_text'].apply(remove_stopwords)

# Perform sentiment analysis on the cleaned text
df['sentiment'] = df['cleaned_title'].apply(get_sentiment)

# Handle missing data (remove rows with missing titles or texts)
df = df.dropna(subset=['cleaned_title', 'cleaned_text'])

# Preview the cleaned data
print(df.head())
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\rites\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\rites\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
                                               title  \
0  /r/Stocks Weekend Discussion Saturday - Dec 07...
1  Should I talk my friend out of a $50k investme...
2         Pros and cons of this investment strategy?
3        These are the stocks on my watchlist (12/6)
4  r/Stocks Daily Discussion & Fundamentals Frida...

                                                text  score  \
0  This is the weekend edition of our stickied di...      3
1  First, me and my friend talk every few week ab...     16
2  I have Roth IRA with 150k in it and like every...      0
3  Hi! I am an ex-prop shop equity trader.\n\nThi...      1
4  This is the daily discussion, so anything stoc...     11

              created                                                url  \
0 2024-12-07 16:00:18  https://www.reddit.com/r/stocks/comments/1h8pr...
1 2024-12-06 23:22:55  https://www.reddit.com/r/stocks/comments/1h879...
2 2024-12-06 22:28:08  https://www.reddit.com/r/stocks/comments/1h85y...
3 2024-12-06 19:46:34  https://www.reddit.com/r/stocks/comments/1h82b...
4 2024-12-06 16:00:10  https://www.reddit.com/r/stocks/comments/1h7yg...

   comments                                      cleaned_title  \
0         0    rstocks weekend discussion saturday dec 07 2024
1        53                     talk friend 50k investment xrt
2        13                      pros cons investment strategy
3        20                               stocks watchlist 126
4       311  rstocks daily discussion fundamentals friday d...

                                        cleaned_text sentiment
0  weekend edition stickied discussion thread dis...   neutral
1  first friend talk every week trades hundred fi...   neutral
2  roth ira 150k like everyone want grow know mem...   neutral
3  hi exprop shop equity traderthis daily watchli...   neutral
4  daily discussion anything stocks related fine ...   neutral
```

In [3]:
```python
# Save the cleaned data to a new CSV file
df.to_csv('cleaned_stock_market_discussions.csv', index=False)

# Preview the cleaned data
print(df.head())
```

```
                                               title  \
0  /r/Stocks Weekend Discussion Saturday - Dec 07...
1  Should I talk my friend out of a $50k investme...
2         Pros and cons of this investment strategy?
3        These are the stocks on my watchlist (12/6)
4  r/Stocks Daily Discussion & Fundamentals Frida...

                                                text  score  \
0  This is the weekend edition of our stickied di...      3
1  First, me and my friend talk every few week ab...     16
2  I have Roth IRA with 150k in it and like every...      0
3  Hi! I am an ex-prop shop equity trader.\n\nThi...      1
4  This is the daily discussion, so anything stoc...     11

              created                                                url  \
0 2024-12-07 16:00:18  https://www.reddit.com/r/stocks/comments/1h8pr...
1 2024-12-06 23:22:55  https://www.reddit.com/r/stocks/comments/1h879...
2 2024-12-06 22:28:08  https://www.reddit.com/r/stocks/comments/1h85y...
3 2024-12-06 19:46:34  https://www.reddit.com/r/stocks/comments/1h82b...
4 2024-12-06 16:00:10  https://www.reddit.com/r/stocks/comments/1h7yg...

   comments                                      cleaned_title  \
0         0    rstocks weekend discussion saturday dec 07 2024
1        53                     talk friend 50k investment xrt
2        13                      pros cons investment strategy
3        20                               stocks watchlist 126
4       311  rstocks daily discussion fundamentals friday d...

                                        cleaned_text sentiment
0  weekend edition stickied discussion thread dis...   neutral
1  first friend talk every week trades hundred fi...   neutral
2  roth ira 150k like everyone want grow know mem...   neutral
3  hi exprop shop equity traderthis daily watchli...   neutral
4  daily discussion anything stocks related fine ...   neutral
```

In [4]:
```python
# using matplotlib and seaborn
import matplotlib.pyplot as plt
import seaborn as sns

# Plot the distribution of sentiment
sns.countplot(x='sentiment', data=df)
plt.title('Sentiment Distribution of Stock Market Posts')
plt.show()
```



2. Data Analysis

Sentiment Analysis

Analyze the sentiment of user-generated content to gauge market sentiment.

Approaches:

Pre-trained models: Hugging Face's BERT (e.g., FinBERT specialized for financial sentiment). Sentiment analysis APIs (e.g., Google NLP, AWS Comprehend). Custom model: Train a sentiment classifier using labeled financial sentiment datasets (e.g., FinancialPhraseBank). Sentiment Categories:

Positive, Neutral, Negative Optionally, assign scores (e.g., [-1, 1]). Tools and Libraries:

spaCy, NLTK, Transformers (Hugging Face)

In [5]:
```python
from textblob import TextBlob

# Function to get sentiment polarity (positive, negative, neutral)
def get_sentiment(text):
    analysis = TextBlob(text)
    # Sentiment polarity: >0 is positive, <0 is negative, ==0 is neutral
    if analysis.sentiment.polarity > 0:
        return 'positive'
    elif analysis.sentiment.polarity < 0:
        return 'negative'
    else:
        return 'neutral'

# Apply sentiment analysis on the 'cleaned_text' column
df['sentiment'] = df['cleaned_text'].apply(get_sentiment)

# Optionally, calculate the polarity score directly
df['polarity'] = df['cleaned_text'].apply(lambda text: TextBlob(text).sentiment.polarity)

# Preview the sentiment results
print(df[['title', 'sentiment', 'polarity']].head())

# Save the sentiment data to CSV
df.to_csv('sentiment_analyzed_stock_discussions.csv', index=False)
```

```
                                               title sentiment  polarity
0  /r/Stocks Weekend Discussion Saturday - Dec 07...  negative -0.031250
1  Should I talk my friend out of a $50k investme...  positive  0.200962
2         Pros and cons of this investment strategy?  negative -0.028571
3        These are the stocks on my watchlist (12/6)  positive  0.145052
4  r/Stocks Daily Discussion & Fundamentals Frida...  positive  0.018981
```

Topic Modeling Using LDA

In [6]:
```python
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.decomposition import LatentDirichletAllocation

# Download NLTK data (tokenizer, stopwords)
nltk.download('punkt')
nltk.download('stopwords')

# Function to preprocess the text (tokenization and stopword removal)
def preprocess_text(text):
    stop_words = set(stopwords.words('english'))
    words = word_tokenize(text.lower())
    return " ".join([word for word in words if word.isalnum() and word not in stop_words])

# Apply preprocessing to the 'cleaned_text' column
df['processed_text'] = df['cleaned_text'].apply(preprocess_text)

# Vectorize the processed text using CountVectorizer
vectorizer = CountVectorizer(max_features=1000)  # You can adjust the number of features
X = vectorizer.fit_transform(df['processed_text'])

# Perform LDA (Topic Modeling)
lda = LatentDirichletAllocation(n_components=5, random_state=42)  # 5 topics, you can adjust
lda.fit(X)

# Print the top words for each topic
num_words = 10
for index, topic in enumerate(lda.components_):
    print(f"Topic {index+1}:")
    print([vectorizer.get_feature_names_out()[i] for i in topic.argsort()[-num_words:]])
    print("\n")
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\rites\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\rites\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
Topic 1:
['dont', 'meme', 'shares', 'know', 'ive', 'fund', 'investment', 'im', 'price', 'would']


Topic 2:
['money', 'think', 'ago', 'good', 'much', 'like', 'stocks', 'years', 'stock', 'market']


Topic 3:
['roi', 'list', 'companies', 'price', 'company', 'value', 'intrinsic', 'like', 'market', 'stock']


Topic 4:
['im', 'trading', 'question', 'daily', 'fundamentals', 'earnings', 'stock', 'market', 'news', 'stocks']


Topic 5:
['financial', 'production', 'market', 'optimus', 'competitors', 'level', 'like', 'autonomy', 'tesla', 'teslas']
```

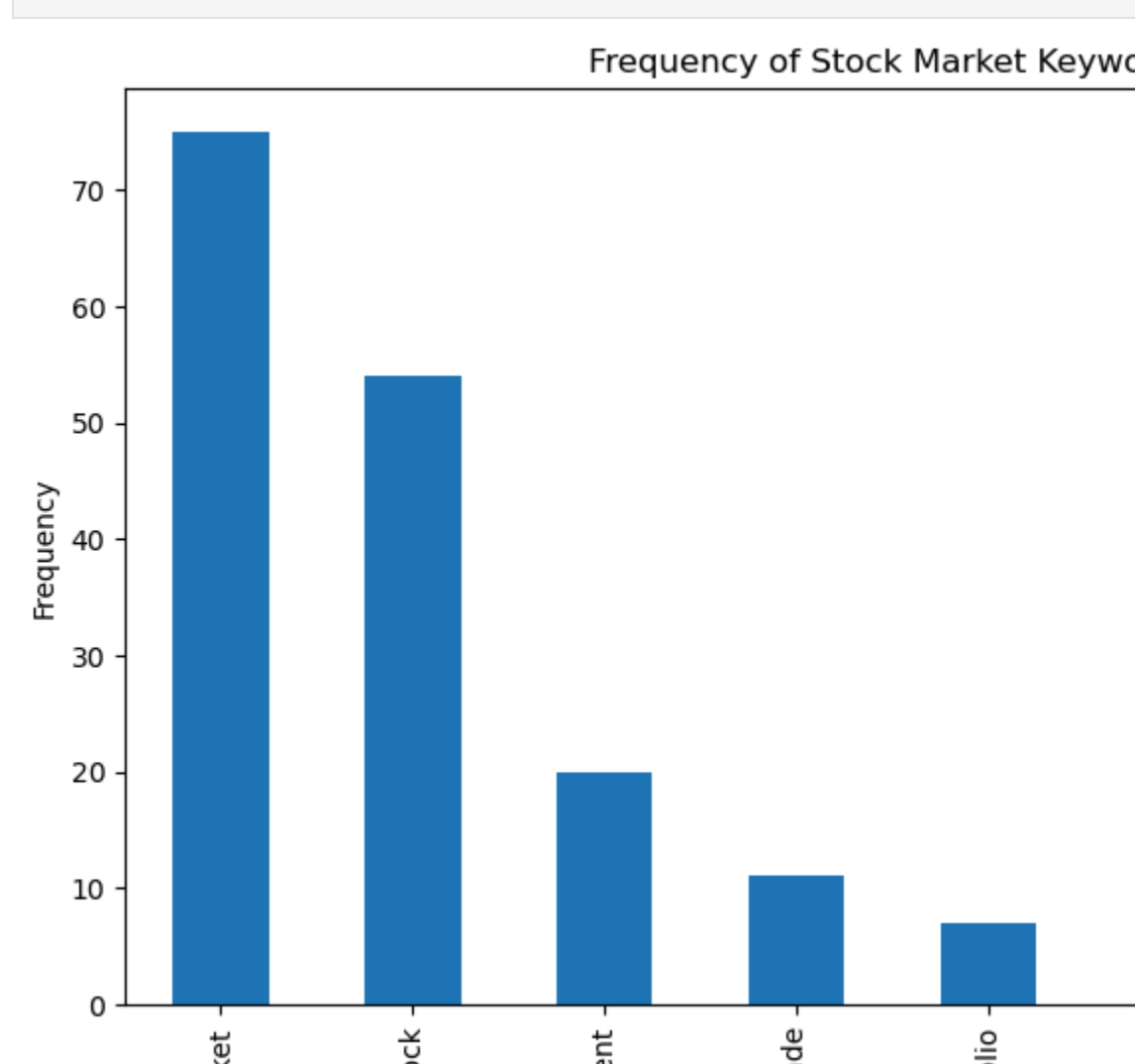
Frequency of Mentions (Keyword Frequency)

In [7]:
```python
from collections import Counter

# Define stock-related keywords
keywords = ['stock', 'investment', 'market', 'trade', 'prediction', 'bull', 'bear', 'portfolio']

# Function to count keyword frequency in each post
def count_keywords(text, keywords):
    words = text.lower().split()
    keyword_counts = {keyword: words.count(keyword) for keyword in keywords}
    return keyword_counts

# Apply the keyword count function to each post
df['keyword_counts'] = df['cleaned_text'].apply(count_keywords, keywords=keywords)

# Convert keyword counts into separate columns for analysis
keyword_df = pd.DataFrame(df['keyword_counts'].tolist())

# Merge the keyword counts with the original DataFrame
df = pd.concat([df, keyword_df], axis=1)

# Preview the DataFrame with keyword frequencies
print(df[['title', 'keyword_counts']].head())

# You can also calculate the total frequency of mentions for each keyword across all posts
total_keyword_counts = keyword_df.sum()
print("Total keyword frequencies across all posts:")
print(total_keyword_counts)
```

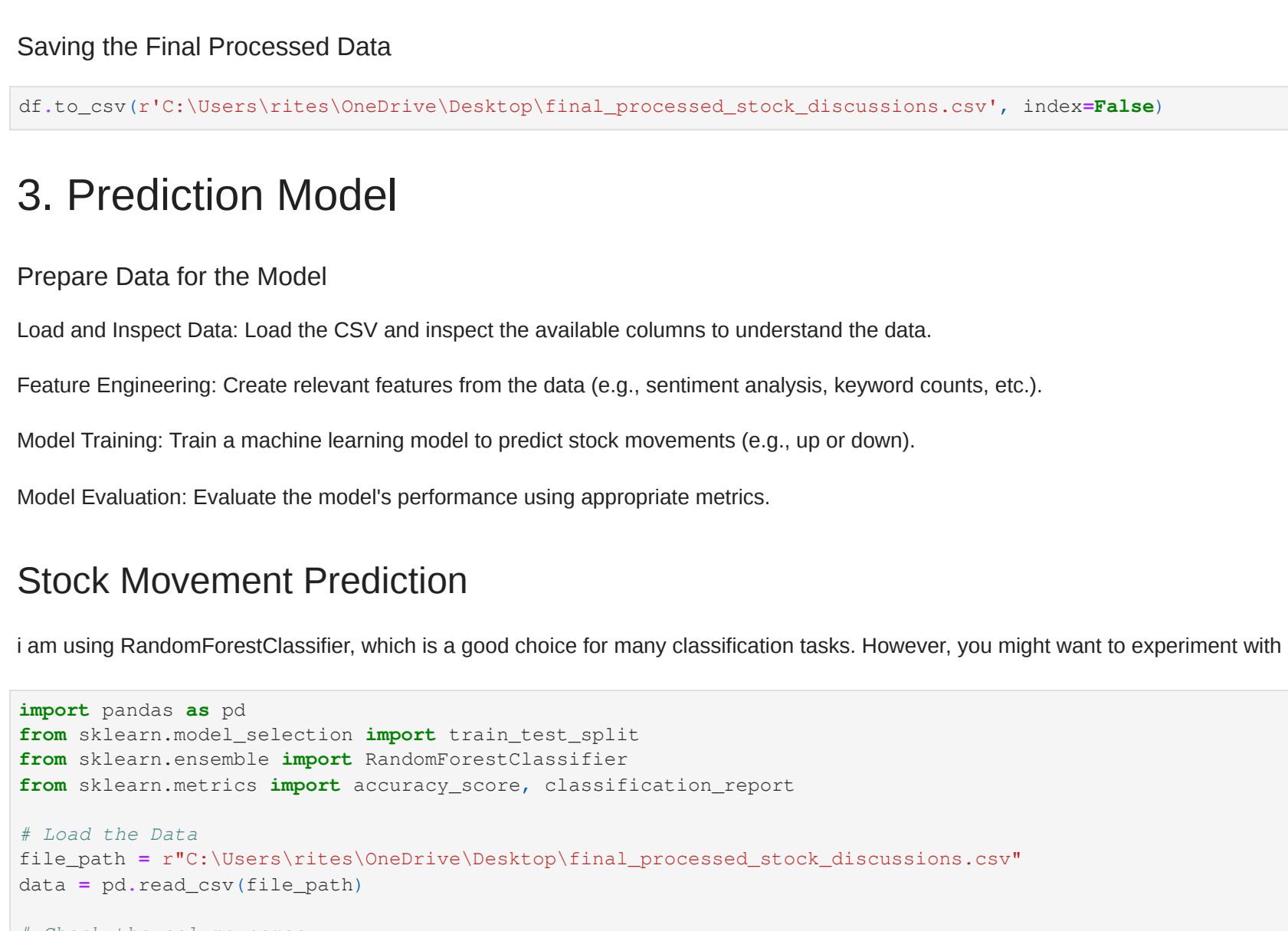
```
                                               title  \
0  /r/Stocks Weekend Discussion Saturday - Dec 07...
1  Should I talk my friend out of a $50k investme...
2         Pros and cons of this investment strategy?
3        These are the stocks on my watchlist (12/6)
4  r/Stocks Daily Discussion & Fundamentals Frida...

                                      keyword_counts
0  {'stock': 0, 'investment': 0, 'market': 3, 'tr...
1  {'stock': 0, 'investment': 0, 'market': 0, 'tr...
2  {'stock': 1, 'investment': 4, 'market': 0, 'tr...
3  {'stock': 2, 'investment': 0, 'market': 1, 'tr...
4  {'stock': 1, 'investment': 0, 'market': 3, 'tr...
Total keyword frequencies across all posts:
stock         54
investment    20
market        75
trade         11
prediction     0
bull           0
bear           0
portfolio      7
dtype: int64
```

Visualization

In [8]:
```python
#Sentiment Distribution
import seaborn as sns
import matplotlib.pyplot as plt

# Plot the sentiment distribution
sns.countplot(x='sentiment', data=df)
plt.title('Sentiment Distribution of Stock Market Posts')
plt.show()
```



In [11]:
```python
#Keyword Frequency Distribution
# Plot the total frequency of keywords
total_keyword_counts.sort_values(ascending=False).plot(kind='bar', figsize=(10, 6))
plt.title('Frequency of Stock Market Keywords')
plt.ylabel('Frequency')
plt.show()
```



Saving the Final Processed Data

In [12]:
```python
df.to_csv(r'C:\Users\rites\OneDrive\Desktop\final_processed_stock_discussions.csv', index=False)
```

3. Prediction Model

Prepare Data for the Model

Load and Inspect Data: Load the CSV and inspect the available columns to understand the data.

Feature Engineering: Create relevant features from the data (e.g., sentiment analysis, keyword counts, etc.).

Model Training: Train a machine learning model to predict stock movements (e.g., up or down).

Model Evaluation: Evaluate the model's performance using appropriate metrics.

Stock Movement Prediction

i am using RandomForestClassifier, which is a good choice for many classification tasks. However, you might want to experiment with other models or tune hyperparameters for better performance.

In [12]:
```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report

# Load the Data
file_path = r"C:\Users\rites\OneDrive\Desktop\final_processed_stock_discussions.csv"
data = pd.read_csv(file_path)

# Check the column names
print("Column names:", data.columns.tolist())

# Check the first few rows of the DataFrame
print(data.head())

# Step 2: Preprocess the Data
# Handle missing values
data.fillna(method='ffill', inplace=True)

# Encode categorical variables if necessary
data = pd.get_dummies(data)
```

```
Column names: ['title', 'text', 'score', 'created', 'url', 'comments', 'cleaned_title', 'cleaned_text', 'sentiment', 'polarity', 'processed_text', 'keyword_counts', 'stock', 'inves
tment', 'market', 'trade', 'prediction', 'bull', 'bear', 'portfolio']
                                               title  \
0  /r/Stocks Weekend Discussion Saturday - Dec 07...
1  Should I talk my friend out of a $50k investme...
2         Pros and cons of this investment strategy?
3        These are the stocks on my watchlist (12/6)
4  r/Stocks Daily Discussion & Fundamentals Frida...

                                                text  score  \
0  This is the weekend edition of our stickied di...      3
1  First, me and my friend talk every few week ab...     16
2  I have Roth IRA with 150k in it and like every...      0
3  Hi! I am an ex-prop shop equity trader.\n\nThi...      1
4  This is the daily discussion, so anything stoc...     11

               created                                                url  \
0  2024-12-07 16:00:18  https://www.reddit.com/r/stocks/comments/1h8pr...
1  2024-12-06 23:22:55  https://www.reddit.com/r/stocks/comments/1h879...
2  2024-12-06 22:28:08  https://www.reddit.com/r/stocks/comments/1h85y...
3  2024-12-06 19:46:34  https://www.reddit.com/r/stocks/comments/1h82b...
4  2024-12-06 16:00:10  https://www.reddit.com/r/stocks/comments/1h7yg...

   comments                                      cleaned_title  \
0         0    rstocks weekend discussion saturday dec 07 2024
1        53                     talk friend 50k investment xrt
2        13                      pros cons investment strategy
3        20                               stocks watchlist 126
4       311  rstocks daily discussion fundamentals friday d...

                                        cleaned_text sentiment  polarity  \
0  weekend edition stickied discussion thread dis...  negative -0.031250
1  first friend talk every week trades hundred fi...  positive  0.200962
2  roth ira 150k like everyone want grow know mem...  negative -0.028571
3  hi exprop shop equity traderthis daily watchli...  positive  0.145052
4  daily discussion anything stocks related fine ...  positive  0.018981

                                      processed_text  \
0  weekend edition stickied discussion thread dis...
1  first friend talk every week trades hundred fi...
2  roth ira 150k like everyone want grow know mem...
3  hi exprop shop equity traderthis daily watchli...
4  daily discussion anything stocks related fine ...

                                      keyword_counts  stock  investment  \
0  {'stock': 0, 'investment': 0, 'market': 3, 'tr...      0           0
1  {'stock': 0, 'investment': 0, 'market': 0, 'tr...      0           0
2  {'stock': 1, 'investment': 4, 'market': 0, 'tr...      1           4
3  {'stock': 2, 'investment': 0, 'market': 1, 'tr...      2           0
4  {'stock': 1, 'investment': 0, 'market': 3, 'tr...      1           0

   market  trade  prediction  bull  bear  portfolio
0       3      0           0     0     0          1
1       0      0           0     0     0          0
2       0      0           0     0     0          0
3       1      2           0     0     0          0
4       3      0           0     0     0          0
```

Feature Importance: After training the model, you might want to check which features are most important for predictions. This can help you understand your model better and refine your feature set.

Hyperparameter Tuning: Consider using techniques like Grid Search or Random Search to find the best hyperparameters for your model.

Cross-Validation: Instead of a single train-test split, you might want to use cross-validation to get a better estimate of your model's performance.