A blue and black text

AI-generated content may be incorrect.

**Software Engineering Department  
Braude College**

**Capstone Project Phase A – 61998**

**SENTRA: Sentiment and Trading**

**25-2-D-19**

**Students:**  **Yosef Jirees :** yjirees@gmail.com

**Yaqob Sadran:** Jakob.sadran@braude.ac.il

**Supervisor: Dr. Reuven Cohen**

**GitHub:-** [**https://github.com/yosef2000138/SENRAT**](https://github.com/yosef2000138/SENRAT)

**Table of Contents**

[1. Introduction 2](#_Toc767519036)

[2. Literature Survey 3](#_Toc1888761120)

[2.1 Bloomberg Terminal 3](#_Toc1601942272)

[2.2 AlphaSense 4](#_Toc712978107)

[2.3 Accern 4](#_Toc1227377212)

[2.4 Kavout (Kai Score) 4](#_Toc1725648523)

[3. Background 4](#_Toc1644439184)

[3.1 Fast Market Moves After Major News 4](#_Toc2036776656)

[3.2 Volume Weighted Average Price (VWAP) — Summary 5](#_Toc1730310154)

[3.2.1 Execution Benchmark & Trend Indicator 5](#_Toc673266104)

[3.2.2 Dynamic Support and Resistance 5](#_Toc71656928)

[3.3 Why VWAP Matters in Market Analysis 6](#_Toc80891446)

[3.4 VWAP Lag in Fast Markets 6](#_Toc1408046438)

[3.5 Fast Market Reactions to News 6](#_Toc808041479)

[3.6 BERT Language Model 7](#_Toc1978006929)

[4. Expected Achievements 7](#_Toc423342381)

[4.1 News Sentiment Analysis 7](#_Toc1321506900)

[4.2 Backtesting Environment 8](#_Toc372465200)

[5. Engineering Process 8](#_Toc498839929)

[5.1 Research and Tools Selection 9](#_Toc311065160)

[5.2 Data Pipeline Development 9](#_Toc547063359)

[5.3 Sentiment & Price Integration 9](#_Toc378162796)

[5.4 Signal Generation Logic 10](#_Toc334008711)

[5.5 Multi-Factor Strategy Design 10](#_Toc586932314)

[5.6 Multi-Factor Trading Strategy 11](#_Toc157225077)

[6. Algorithms and Models Used 14](#_Toc785151752)

[6.1 FinBERT – News Sentiment Analysis Model 14](#_Toc1390296759)

[6.2 VWAP – Volume Weighted Average Price 15](#_Toc1947704433)

[6.3 Volume Spike Detection 15](#_Toc805344648)

[6.4 1-Hour Price Momentum 15](#_Toc2137797191)

[6.5 Composite Signal Scoring 15](#_Toc794793525)

[6.5 Data Structures 16](#_Toc1571419581)

[7. Data Set and Evaluation 16](#_Toc682781749)

[7.1 Data Set 16](#_Toc1894813862)

[7.2 Evaluation Criteria and Metrics 17](#_Toc1341860706)

[7.3 Testing Methodology 18](#_Toc1972512474)

[8. Product 19](#_Toc1497157868)

[8.1 System Data Flow (Software Architecture Diagram) 19](#_Toc1634286435)

[8.2 Use Case Diagram 22](#_Toc1457565629)

[8.3 Activity Diagram 23](#_Toc215397542)

[8.4 User-Friendly Interface 24](#_Toc1713704685)

[9. Common Problems 26](#_Toc1917743584)

[9.1 Technical Challenges 27](#_Toc1204380682)

[9.2 Team Challenges 29](#_Toc653032722)

[References 29](#_Toc773667860)

**Abstract**

Financial markets react rapidly to breaking news and shifts in public sentiment. Traditional trading models, which rely primarily on historical prices and technical indicators, often miss these short-term impacts. This project aims to enhance intraday trading strategies by incorporating **real-time news sentiment analysis** using machine learning and natural language processing (NLP). We propose a **hybrid decision engine** that integrates news sentiment (specifically for **Bitcoin** with technical indicators like **VWAP (Volume Weighted Average Price)** to generate actionable trading signals. The project aims to develop a robust tool for day traders to capitalize on rapid market movements, offering more timely and accurate signals than traditional methods. **Back testing** and comprehensive performance metrics will be used to validate its effectiveness and demonstrate its practical applicability in real-world trading scenarios.

# 1. Introduction

Financial markets are highly sensitive to news and public sentiment, with breaking news often triggering immediate price movements in assets like stocks, currencies, and cryptocurrencies. While traditional trading models primarily rely on historical price data and technical indicators, they frequently fall short in capturing the rapid, short-term impact of significant news events. With the advent of big data analytics and advanced machine learning techniques, there's a significant opportunity to enhance market analysis by integrating **real-time news sentiment** into predictive models.

This project focuses on developing a **machine learning-based system** that analyzes financial news, specifically for **Bitcoin (BTC)** , and evaluates their immediate influence on market indicators and price charts. By employing **natural language processing (NLP) techniques**, **sentiment analysis**, and **time-series forecasting**, the model aims to identify patterns between news sentiment and subsequent short-term market movements. The ultimate goal is to create a **real-time trading indicator** that integrates sentiment analysis into a signal generation pipeline, offering a competitive edge in fast-moving markets.

This document outlines the development of this system, covering its core components, methodologies, and expected outcomes. We will delve into how news data is processed, how market data is integrated, the models developed, and how their performance is evaluated. The project aims to provide a practical tool for traders, allowing them to make more informed and timely decisions by understanding the combined influence of fundamental news and technical market behavior.

Key aspects of this project include:

* **News Data Processing**: Collecting and preprocessing financial news from multiple sources, such as Investing.com and real news networks, to extract relevant information.
* **Market Data Integration**: Combining news sentiment with historical price data for BTC/USD to train machine learning models that predict market reactions.
* **Model Development**: Implementing supervised learning and deep learning approaches (e.g., **LSTM, Transformer models**) to assess the correlation between news sentiment and price changes. **LSTM (Long Short-Term Memory)** networks are a type of recurrent neural network (RNN) well-suited for processing sequences, making them ideal for time-series data like market prices. **Transformer models**, like BERT, are powerful architectures known for their ability to capture long-range dependencies in sequential data, highly effective for complex NLP tasks.
* **Performance Evaluation**: Validating the model's accuracy in predicting short-term market movements and assessing its practical applicability in day trading strategies through robust backtesting methods.

# 2. Literature Survey

In recent years, the integration of news sentiment analysis into financial market prediction has attracted significant attention, particularly with the rise of machine learning and natural language processing techniques. As financial markets become increasingly sensitive to real-time information, several platforms and tools have emerged aiming to process and evaluate news content to support trading decisions. This section describes existing solutions that link financial news with market behavior. These systems vary in complexity, data sources, and analytical models, yet all share a common goal: enhancing the responsiveness and accuracy of market analysis by leveraging the informational value embedded in news data.

## 2.1 Bloomberg Terminal

A leading professional financial platform that provides real-time financial news, technical indicators, and market analytics [1]. Bloomberg integrates news sentiment and event-driven data into trading analysis, allowing traders to react quickly to breaking news alongside traditional technical metrics like VWAP and moving averages.

## 2.2 AlphaSense

An AI-based search engine for financial documents and news [2]. AlphaSense analyzes sentiment and extracts insights from earnings calls, research reports, and financial news articles to help investment professionals make faster and more informed decisions.

## 2.3 Accern

A real-time AI news monitoring system automatically analyzes financial news events' impact [3]. Accern delivers sentiment scores and relevance ratings, helping traders and analysts understand the importance of new information and adjust their strategies accordingly.

## 2.4 Kavout (Kai Score)

A financial intelligence platform that combines machine learning with news analysis, technical indicators, and fundamental data [4]. Kavout generates a predictive "**Kai Score**" for stocks, which is an integrated view based on both quantitative (technical indicators, financial data) and qualitative (news sentiment) information, indicating a stock's potential movement.

# 3. Background

## 3.1 Fast Market Moves After Major News

Major financial news events often trigger rapid and significant market movements. For instance, on May 5, 2025, U.S. stock futures dropped sharply within 10–30 minutes after President Donald Trump announced a 100% tariff on foreign films, triggering fears of a global trade war and causing major media stocks like Netflix and Disney to fall significantly (Reuters, 2025-05-05). Similarly, oil prices declined over 2% immediately after OPEC+ revealed plans to accelerate production increases, prompting Gulf markets—such as Saudi Arabia’s index—to fall 0.7% within the first trading hour (Reuters, 2025-05-05). In a related shift in sentiment, U.S. consumer confidence data from April 2025 showed continued weakness, and markets responded within 5–15 minutes of the release, reflecting investor concerns over the economic impact of tariffs (Reuters, 2025-04-25). These examples underscore the critical need for trading strategies that can rapidly integrate and react to news sentiment.

## 3.2 Volume Weighted Average Price (VWAP) — Summary

The **Volume Weighted Average Price (VWAP)** is a widely used intraday trading indicator that measures the average price of a security **[5] Anese et al. (2023)** , weighted by the volume traded, over a specific time period. Unlike simple averages, VWAP incorporates both price and volume, providing a more accurate reflection of where the majority of trading activity has occurred. It is a crucial tool for institutional traders and provides insights into market trends and liquidity .

The formula for VWAP is expressed as follows:

A black text on a white background

AI-generated content may be incorrect.

**where:**

• HLC **t** is the average between High, Low and Close of the security in minute **t**,

• Volume **t** is the trading volume of the security in minute **t**,

• The summation runs over all observations within the specified period.

### 3.2.1 Execution Benchmark & Trend Indicator

Institutional traders often compare their trade prices against VWAP to evaluate their performance. Achieving a better price than the VWAP indicates efficient execution. When the current market price is above the VWAP, it typically signals bullish sentiment; when below, it indicates bearish sentiment [**5] Anese et al. (2023).**

### 3.2.2 Dynamic Support and Resistance

VWAP can act as a real-time **support or resistance level**, helping traders make informed entry and exit decisions during the trading day. In practice, traders use VWAP not only to understand price movements but also to assess liquidity and market sentiment. Its real-time adaptation to both price and volume changes makes it particularly valuable for day traders aiming to ride intraday trends while maintaining risk management discipline. Overall, VWAP is considered an essential tool in modern trading due to its ability to synthesize crucial market information into a single, actionable indicator **[5] Anese et al. (2023).**

## 3.3 Why VWAP Matters in Market Analysis

VWAP is a valuable intraday indicator that shows the average price of an asset, weighted by volume. It helps traders assess whether they're buying or selling at a fair price and is commonly used to evaluate trade execution. A price above VWAP suggests bullish sentiment, while a price below VWAP indicates bearish pressure. By combining price and volume, VWAP also highlights key support and resistance levels during the trading day. It provides a more comprehensive view of market activity compared to simple moving averages, as it considers the strength of price movements through traded volume **[5] Anese et al. (2023).**

## 3.4 VWAP Lag in Fast Markets

According to Zarattini and Aziz (2023), the VWAP (Volume Weighted Average Price) indicator exhibits structural limitations when it comes to adapting to fast market movements. Since VWAP is calculated cumulatively from the start of the trading session, it reacts slowly to abrupt changes in price direction. This delay arises because the indicator smooths out volatility by averaging historical price and volume data, which makes it less sensitive to real-time market dynamics. As the authors explain, this characteristic becomes problematic in environments with frequent price reversals or heightened volatility, where rapid sentiment shifts occur. In such cases, VWAP is often unable to reflect the current momentum accurately, reducing its effectiveness as a decision-making tool for short-term trading. This lag is a primary motivation for integrating news sentiment, which can capture rapid shifts that VWAP might miss.

## 3.5 Fast Market Reactions to News

Public financial news has a measurable and rapid impact on stock market behavior, particularly within the first 20 minutes following its release. As shown in the study by Anese et al. (2023) [5], sentiment extracted from financial news—especially when processed through advanced models like LSTM neural networks—can significantly influence both returns and volatility in a short time window. The research demonstrates that news-related sentiment shifts are quickly absorbed by market participants, leading to immediate changes in price direction and increased volatility. This highlights how real-time news can serve as a powerful driver of market sentiment and momentum, sometimes faster than traditional indicators can respond. As stated in the paper: "Our findings show that these sentiments are significant market predictors in a 20-min time window after the public release of the news" **[9]** Zarattini (2023), This rapid market reaction to news is a core premise for our project.

## 3.6 BERT Language Model

**BERT (Bidirectional Encoder Representations from Transformers)** is a language representation model introduced by Google AI in 2018. It is designed to pre-train deep bidirectional representations by jointly conditioning on both left and right context in all layers. This is achieved through a **masked language modeling objective**, where randomly selected tokens in the input are masked, and the model learns to predict them, allowing it to capture contextual information from both directions. Additionally, BERT is trained on a **next sentence prediction task** to understand the relationship between sentence pairs. The model significantly improves performance on a wide range of natural language processing tasks, such as question answering and language inference, by enabling fine-tuning with minimal task-specific architecture changes. BERT's ability to understand context bidirectionally makes it highly effective for nuanced sentiment analysis in financial texts.

# 4. Expected Achievements

The final expectation for this project is a **working prototype of a news-driven financial trading indicator** that integrates real-time sentiment analysis into a trading signal pipeline, with backtested results and documented performance.

## 4.1 News Sentiment Analysis

Our system will perform the following steps for news sentiment analysis:

* **Analyze real-time and historical financial news**: This includes collecting data from various sources relevant to Bitcoin and USD.
* **Preprocess the data**: This involves filtering irrelevant articles, noise, or duplicates to ensure data quality.
* **Classify each news headline or article**: Using our specialized FinBERT model, each piece of news will be classified as positive, negative, or neutral.
* **Calculate a sentiment score**: A numerical sentiment score will be derived for each article, ranging from -1 (very negative) to +1 (very positive).
* **Match news articles to relevant financial instruments**: The system will identify if a news article has a direct impact on Bitcoin or the Dollar, ensuring signals are generated for the correct asset. For example, news regarding cryptocurrency regulations will be matched to Bitcoin, while economic policy news might be matched to the Dollar.

## 4.2 Backtesting Environment

A robust backtesting environment is crucial for validating our strategy:

* **Simulate trading strategies**: Using historical news and price data for BTC/USD, we will simulate the performance of our hybrid strategy.
* **Evaluate key performance metrics**:
  + **Profit/Loss (P/L)**: Total financial gain or loss over the backtesting period.
  + **Sharpe Ratio**: Measures risk-adjusted return (fractextReturn−textRisk−FreeRatetextStandardDeviationofReturns), indicating how much return is generated per unit of risk.
  + **Win Rate**: The percentage of profitable trades.
  + **Maximum Drawdown**: The largest peak-to-trough decline in the portfolio, representing the worst-case loss from a peak.
  + **Precision**: The proportion of positive identifications that were actually correct (fractextTruePositivestextTruePositives+textFalsePositives). For trading, it means how many 'buy' signals actually led to profitable trades.
  + **Recall**: The proportion of actual positives that were identified correctly (fractextTruePositivestextTruePositives+textFalseNegatives). For trading, it means how many potential profitable trades were actually captured by our 'buy' signals.
  + **Visualize trade entry/exit points**: Displaying these points directly on price charts to illustrate the strategy's actions.
  + **Highlight sentiment spikes**: Overlaying sentiment changes on price charts to visually demonstrate the correlation between news sentiment and subsequent price movements.

5. Engineering Process

**5.1 Research and Tools Selection**

In the initial phase, we conducted market research and literature review on projects and academic studies that focus on news sentiment analysis and its impact on financial markets. We evaluated common sentiment analysis tools such as **FinBERT**, which are widely used for classifying the sentiment of financial news articles and social media posts. Additionally, we analyzed technical indicators, specifically **VWAP (Volume Weighted Average Price)**, which is a trusted indicator for determining the price trend strength during a trading session. This foundational research guided our choice of models and data sources.

**5.2 Data Pipeline Development**

Following the research phase, we built a basic data pipeline for collecting the following data:

* **Real-time and historical news headlines**: Using APIs such as NewsAPI and potentially Bloomberg for more comprehensive financial news.
* **Market data (OHLCV – Open, High, Low, Close, Volume)**: Using services like Yahoo Finance and CryptoCompare for Bitcoin and USD data.

We implemented a preprocessing module that cleans the news headlines (e.g., removing boilerplate text, standardizing formats), applies sentiment analysis using our fine-tuned FinBERT model, and **aligns each news item with corresponding asset prices and VWAP calculations based on precise timestamps**. This synchronization is crucial for accurate signal generation and backtesting.

5.3 Sentiment & Price Integration

The next phase involves the integration of sentiment data with VWAP-based technical analysis to create a hybrid decision engine. This engine will generate buy or sell signals based on both news sentiment and VWAP levels. For example:

* If sentiment is **positive (e.g., sentiment score > 0.3)** AND price is **above VWAP** → Generate a **BUY signal**.
* If sentiment is **negative (e.g., sentiment score &lt; -0.3)** AND price is **below VWAP** → Generate a **SELL signal**.

A **confidence or strength score** will be incorporated into the recommendation. For instance, if the composite signal score (explained later) is very high (e.g., >0.8), it's a strong buy recommendation, suggesting a larger trade size. If it's moderately positive (e.g., 0.5-0.7), it's a weaker buy, implying a smaller position or more cautious entry.

5.4 Signal Generation Logic

We will implement parameter tuning and perform backtesting on historical data to validate the performance of the hybrid strategy. We will evaluate the strategy using key performance metrics such as:

* **Precision, Recall** (as defined in Section 4.2)
* **Win Rate, Sharpe Ratio, Profit Factor, Maximum Drawdown** (as defined in Section 4.2)

Finally, we will develop an interactive dashboard to visualize signals, cumulative returns, and overlay them on price charts.

5.5 Multi-Factor Strategy Design

Modern financial markets are influenced by both technical price movements and sentiment-driven factors (e.g., breaking news, social media posts). While technical indicators like VWAP help identify trend strength and fair value during the trading session, they often fail to capture external events that can drastically move prices (e.g., regulatory news for Bitcoin, major economic announcements for the Dollar).

Thus, our motivation was to combine the strength of technical indicators (VWAP) with sentiment analysis of financial news to reduce false signals and increase the robustness of our trading strategy. This hybrid approach aims to capture both rational (technical) and emotional (news-driven) market behaviors, which we believe can enhance predictive power.

**Constraints and Challenges Faced:**

1. **Data Availability & Licensing Limitations**: Access to quality real-time news and market data requires API subscriptions and often comes with rate limits (e.g., NewsAPI, Yahoo Finance).
   * **Mitigation**: We will prioritize free/affordable APIs initially (e.g., NewsAPI, CryptoCompare for crypto data, potentially web scraping public financial news sources with caution) and consider paid subscriptions if the prototype proves viable and additional funding is secured. We will also implement robust error handling and retry mechanisms for API calls.
2. **Computational Efficiency Requirements**: For real-time deployment, the model needs to process high-frequency data and generate signals quickly, which forces us to choose lightweight, efficient models rather than heavy deep learning models.
   * **Mitigation**: We will use a pre-trained and fine-tuned FinBERT model that balances accuracy with inference speed. We will optimize the data pipeline for parallel processing and minimize unnecessary computations. Initial focus will be on hourly or 30-minute timeframes rather than tick data to manage computational load.
3. **Noise in Sentiment Data**: News sentiment analysis is inherently noisy, especially when using general sentiment analyzers like VADER on financial text. Financial language often contains specific jargon that general models might misinterpret.
   * **Mitigation**: We will use a **FinBERT model**, which is specifically fine-tuned on financial corpora, making it more accurate for financial sentiment than general-purpose sentiment analyzers. We will also implement a **sentiment aggregation strategy** (e.g., averaging sentiment scores over a short time window) to smooth out individual noisy readings.
4. **Market Reaction Lag**: Aligning the time of news publication with price movements is challenging and requires careful timestamp alignment and data preprocessing. News often has a "decay" effect on prices.
   * **Mitigation**: We will ensure precise timestamping of news articles and market data down to the minute. We will also research and apply techniques like event-time analysis to identify the immediate impact window (e.g., 5, 15, or 30 minutes post-news release) and potentially incorporate a time-decay factor into the sentiment score.

**5.6 Multi-Factor Trading Strategy**

The system will analyze the following trade indicators:

1. **Price and Volume Changes (1-Hour Price Momentum)**:
   * If Bitcoin's price increases or drops more than ±2% in 1 hour, and volume is significantly above average, it might signal strong interest.
   * This indicates short-term momentum. You can enter in the direction of the spike.
2. **Volume Spike Detection**:
   * Look for volume that is significantly higher than the recent average.
   * **Spikes often precede large moves or mark turning points**. A common threshold is current volume exceeding 1.5 times the average volume over the last 24 hours. This confirms the strength and validity of a price movement.
3. **Candle Body Strength (Volatility-Based Filter)**:
   * Large candles (measured by abs(Close - Open) / Open in %) indicate strong momentum and significant price movement within a period.
   * Combine with sentiment: e.g., if the sentiment is positive AND the candle is large bullish (Close > Open by a significant margin) → strong confirmation for a buy signal.
4. **RSI (Relative Strength Index) + Sentiment Filter**:
   * **RSI** is a momentum oscillator that measures the speed and change of price movements. It ranges from 0 to 100.
     + **Oversold (RSI leq30)**: Suggests the asset may be undervalued or due for a price correction upwards.
     + **Overbought (RSI geq70)**: Suggests the asset may be overvalued or due for a price correction downwards.
   * Combine with FinBERT sentiment:
     + If **RSI &lt; 30** (oversold) AND **sentiment > 0.3** (positive) → Stronger **BUY** signal. This suggests a potential rebound confirmed by positive news.
     + If **RSI > 70** (overbought) AND **sentiment &lt; -0.3** (negative) → Stronger **SELL** signal. This suggests a potential reversal confirmed by negative news.
   * If RSI is between 30 and 70, it is considered neutral for this filter (output 0).

**Normalize Signals**

Convert each signal into a normalized value between -1, 0, or +1:

|  |  |
| --- | --- |
| **Signal Type** | **Output** |
| Sentiment > 0.3 | +1 |
| Sentiment &lt; -0.3 | -1 |
| Otherwise (neutral sentiment) | 0 |
| Price > VWAP | +1 |
| Price &lt; VWAP | -1 |
| Price == VWAP | 0 |
| Volume > 1.5×AvgVol | +1 |
| Volume &lt; AvgVol | -1 (Indicates low interest, less strong signal) |
| 1h Return > +2% | +1 |
| 1h Return &lt; -2% | -1 |
| Otherwise (neutral momentum) | 0 |
| RSI &lt; 30 | +1 |
| RSI > 70 | -1 |
| Otherwise (neutral RSI) | 0 |

**Assign Weights**

Each factor will be assigned a weight, reflecting its perceived importance in generating a reliable signal. These weights can be tuned during backtesting using optimization algorithms to find the best combination for maximizing returns and minimizing risk.

|  |  |  |
| --- | --- | --- |
| **Signal Factor** | **Weight** | **Explanation** |
| FinBERT Sentiment | 0.4 | Sentiment is a strong, fast predictor of direction in news-driven markets. |
| VWAP Position | 0.2 | Technical positioning; institutional traders use this for fair value and trend confirmation. (Reduced weight compared to sentiment as it lags). |
| Volume Spike | 0.2 | Confirms a strong move is meaningful and backed by market participation. |
| 1-Hour Momentum | 0.1 | Short-term movement confirmation, indicates immediate price pressure. |
| RSI Filter | 0.1 | Provides overbought/oversold context, enhancing signal reliability at extremes. |

*Total weights sum to 1.0 — we can tune these during backtesting.*

**Generate Final Signal**

A **Composite Score** will be calculated as the weighted sum of the normalized signals. This score will then determine the final action:

|  |  |
| --- | --- |
| **Composite Score Range** | **Action** |
| geq+0.5 | **BUY** (Stronger buy signals for scores > 0.7, weaker for 0.5-0.7) |
| leq−0.5 | **SELL** (Stronger sell signals for scores &lt; -0.7, weaker for -0.5 to -0.7) |
| Otherwise | **HOLD** (Neutral or unclear market conditions) |

# 6. Algorithms and Models Used

Our strategy integrates both Natural Language Processing (NLP) and technical analysis algorithms in a unified trading model.

## 6.1 FinBERT – News Sentiment Analysis Model

* We plan to use a **custom sentiment analysis model based on BERT**, specifically fine-tuned for financial news, inspired by the structure of FinBERT developed by Prosus AI [6].
* The model will be trained to classify each news headline or article into **positive, neutral, or negative sentiment classes** (e.g., a news article about a company's strong earnings is positive, a regulatory warning is negative, and a routine announcement is neutral).
* It will output a **probability distribution over sentiment classes** (e.g., P(positive)=0.8, P(neutral)=0.1, P(negative)=0.1). We will convert this into a numerical sentiment score ranging from -1 to +1, calculated as: $$ $$$$\text{Sentiment Score} = P(\text{positive}) - P(\text{negative}) $$ $$$$ $$
* This sentiment score will be **synchronized with market data by associating each news event's sentiment with the market data timestamp immediately following its publication**. This ensures accurate alignment between news events and market reactions for signal generation.

## 6.2 VWAP – Volume Weighted Average Price

* VWAP is a widely-used technical indicator that reflects the average price at which an asset has traded throughout the day, weighted by volume.
* It is used as a trend confirmation tool: if the price is above VWAP, the trend is considered bullish; if below, bearish. The calculation uses the formula provided in Section 3.2.

## 6.3 Volume Spike Detection

* To detect unusual trading activity, we calculate a rolling average of volume (e.g., over the last 24 hours) and compare current volume to this historical average.
* If the current volume exceeds **1.5 times the average volume** over the last 24 hours, it is considered a significant volume spike, indicating heightened trading interest.

## 6.4 1-Hour Price Momentum

* We calculate the percentage change in Bitcoin’s price over the last hour.
* If the price change exceeds pm2, it signals significant momentum and is used as a condition in our composite strategy. If the change is within pm2, the momentum signal is considered neutral (output 0).

## 6.5 Composite Signal Scoring

* Each of the above components (Sentiment, VWAP Position, Volume Spike, 1-Hour Momentum, RSI Filter) is assigned a specific weight (as detailed in Section 5.5).
* A final composite score is computed based on the **weighted sum of normalized signal values** (where each signal is -1, 0, or +1).
* Trading signals are generated based on the composite score threshold:
  + Score geq0.5rightarrow **Buy**
  + Score leq−0.5rightarrow **Sell**
  + Otherwise rightarrow **Hold**

## 6.5 Data Structures

While not explicitly detailed in the document, internal data structures will include:

* **Time-series dataframes**: For storing OHLCV (Open, High, Low, Close, Volume) market data, indexed by timestamp.
* **News event objects/dictionaries**: Containing news text, publication timestamp, and calculated sentiment score.
* **Signal logs**: To record each generated signal, its composite score, and the underlying factor values for analysis.
* **Trade records**: For backtesting, storing entry/exit prices, timestamps, and profit/loss for each trade.

# 7. Data Set and Evaluation

## 7.1 Data Set

Our project will primarily rely on **historical and real-time data for Bitcoin (BTC) and the US Dollar (USD)**, specifically the BTC/USD currency pair. The choice of cryptocurrency markets (Bitcoin) is due to their high volatility and frequent, often news-driven, price movements, which make them ideal for testing the responsiveness of our sentiment-driven strategy.

* **Market Data**:
  + **Source**: CryptoCompare API and Yahoo Finance API.
  + **Type**: Hourly OHLCV (Open, High, Low, Close, Volume) data for BTC/USD. This granularity balances real-time responsiveness with computational efficiency.
  + **Period**: We aim to collect data for at least 2-3 years to ensure sufficient historical coverage for robust backtesting, capturing various market cycles and news events.
* **News Data**:
  + **Source**: NewsAPI (for general financial news), investing.com (for targeted financial news, potentially requiring web scraping or specialized APIs), and potentially other public financial news outlets relevant to crypto and forex markets.
  + **Type**: News headlines and short articles.
  + **Relevance Filtering**: News will be filtered to focus on topics relevant to Bitcoin (e.g., regulatory news, adoption, technological developments) and the US Dollar (e.g., macroeconomic indicators, Federal Reserve announcements, geopolitical events).
* **Labeled Sentiment Data (for FinBERT fine-tuning)**:
  + **Source**: Existing financial news sentiment datasets (e.g., Financial PhraseBank, stock market sentiment datasets). If suitable pre-labeled data for crypto/forex specific news is scarce, we might perform a small-scale manual labeling effort for a subset of data to fine-tune FinBERT specifically for our domain.

## 7.2 Evaluation Criteria and Metrics

The success of this project will be evaluated based on the performance of our hybrid trading strategy during backtesting. Our key criteria and metrics for success include:

1. **Profitability**:
   * **Total Return**: The overall percentage gain or loss over the backtesting period.
   * **Profit Factor**: Total gross profit divided by total gross loss. A value greater than 1 indicates a profitable strategy.
   * **Annualized Return**: For comparison with traditional investments.
2. **Risk Management**:
   * **Maximum Drawdown**: The largest percentage drop from a peak to a trough in the equity curve. A lower drawdown indicates better risk management.
   * **Sharpe Ratio**: Measures risk-adjusted return (higher is better).
   * **Sortino Ratio**: Similar to Sharpe, but only considers downside deviation (risk of negative returns).
3. **Signal Quality**:
   * **Win Rate**: Percentage of profitable trades.
   * **Precision and Recall of Signals**: As defined in Section 4.2. High precision ensures fewer false positive signals, while high recall ensures fewer missed opportunities.
4. **Responsiveness**:
   * **Average Reaction Time**: Measure the time difference between news publication and the generation of a relevant signal, aiming for minimal latency.
5. **Robustness**:
   * **Performance across different market conditions**: Testing the strategy during bull, bear, and sideways markets to ensure consistent performance.
   * **Sensitivity Analysis**: Evaluating how changes in model parameters (e.g., sentiment thresholds, weights) affect performance.

## 7.3 Testing Methodology

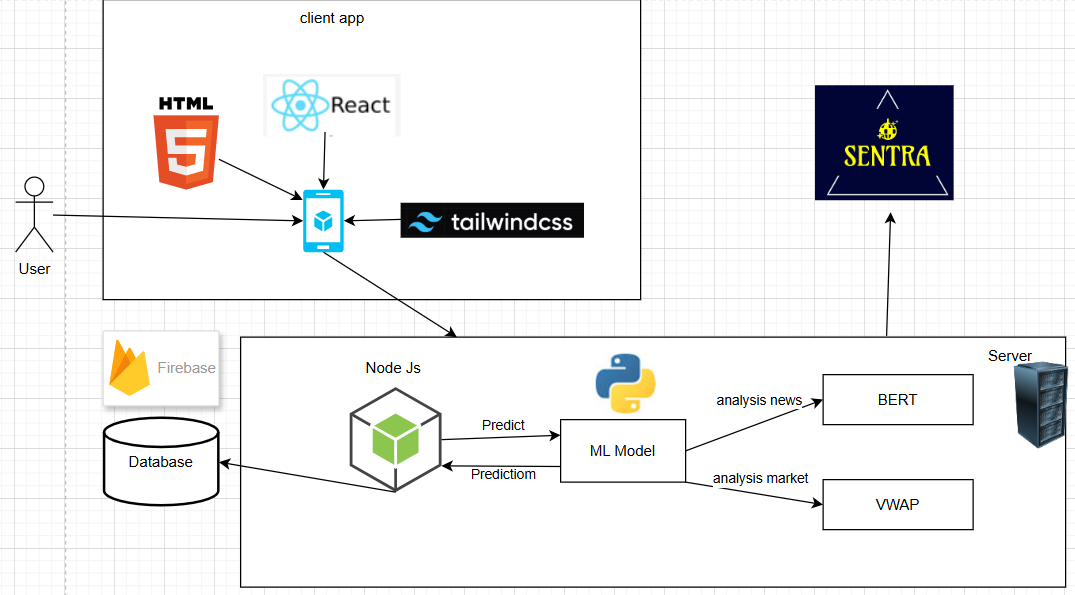
To ensure the reliability and validity of our system, we will employ a rigorous testing methodology:

1. **Unit Testing**: Individual modules (e.g., news scraper, sentiment analysis function, VWAP calculation, signal normalization) will be tested in isolation to ensure they function correctly and produce expected outputs for given inputs.
2. **Integration Testing**: Verify that different components of the system (e.g., data pipeline, sentiment analysis, indicator calculations, signal generation) interact seamlessly and data flows correctly between them.
3. **Backtesting**: This is the most critical testing phase.
   * **Historical Data Split**: The collected historical data will be split into training, validation, and testing sets (e.g., 70% training, 15% validation, 15% testing). The testing set will be strictly unseen data.
   * **Walk-Forward Optimization**: For optimal weight tuning, we will use walk-forward optimization, where weights are optimized on a rolling window of historical data and then applied to the next out-of-sample period. This prevents lookahead bias.
   * **Trade Simulation**: A simulator will execute trades based on generated signals, accounting for realistic factors like slippage and transaction costs (if applicable).
   * **Performance Reporting**: Automated generation of comprehensive reports detailing all evaluation metrics (as listed in Section 7.2), including equity curves and trade logs.
4. **Stress Testing**: Evaluate the system's performance under extreme market conditions (e.g., sudden market crashes, periods of unusually high volatility or low liquidity) to identify potential weaknesses.
5. **Paper Trading (Planned Future Step)**: After successful backtesting, a paper trading phase will be conducted in a live market environment without real capital, to observe performance in real-time and identify any discrepancies not caught in backtesting.

# 8. Product

## 8.1 System Data Flow (Software Architecture Diagram)

This diagram illustrates the key components of our application's software architecture, highlighting the main functions, their implementations, and how they interact with one another. It primarily includes the client-side (Frontend), server-side (Backend), and the database components.



**Client-Side (Frontend)**:

• The client application will be developed using React, a widely used JavaScript library designed for creating dynamic and interactive user interfaces.  
• We'll use HTML and CSS, along with the Tailwind framework, to handle styling and ensure a responsive, modern design. This is where the user interacts with the trading dashboard, visualizes charts, and views signals.

Users will be able to securely log in or create an account, monitor real-time market data and charts, receive and act on automated buy/sell signals, and access the latest financial news and analysis to support their trading decisions.

**Server-Side (Backend)**:

* The server side will be built using **Node.js**, a server-side execution environment for running JavaScript applications. This allows for a unified JavaScript codebase across frontend and backend.
* **Data Ingestion Module**: Responsible for collecting real-time news via APIs (News API, Investing.com) and market data via APIs (Crypto Compare, Yahoo Finance).
* **Data Preprocessing Module**: Cleans and normalizes raw news and market data, aligning timestamps accurately.
* **ML Model (Sentiment Analysis)**: This component houses the **BERT model** (fine-tuned on financial text) that analyzes incoming news to understand current sentiment. It processes news articles and outputs a numerical sentiment score (-1 to +1).
* **VWAP Calculation Module**: Continuously calculates the real-time VWAP based on incoming OHLCV data.
* **Signal Generation Engine**: This core logic combines the sentiment scores, VWAP position, volume spikes, 1-hour momentum, and RSI signals. It applies the defined weights and composite scoring logic to generate a final BUY, SELL, or HOLD recommendation.
  + **API Endpoints**: Node.js will expose RESTful APIs for the frontend to request data, signals, and back testing results.

**Database**:

* The application will primarily use **Firebase**, a NoSQL database, for storing user preferences, historical trade records (from paper trading), and potentially aggregated sentiment data for faster retrieval. For extensive historical OHLCV data, external data providers (like Crypto Compare's historical API) will be relied upon, or data could be cached locally using a more traditional time-series database if performance demands.

## 8.2 Use Case Diagram

This diagram illustrates the primary interactions between the **User** (Trader) and the **Trading System**, outlining the key functionalities offered.

A diagram of a diagram

AI-generated content may be incorrect.

 **Login:**  
The user can log in to access the system.

 **Signup:**  
The user can create a new account in the system.

 **Buy:**  
The user can perform a buy operation (buy stocks, crypto, or assets).

 **Sell:**  
The user can perform a sell operation.

 **Show Chart (extension points Use Case):**  
The user can view charts. This is an **extension point**, meaning other use cases can add extra steps or features to this action.

 **Add Indicator (extends Show Chart):**  
The user can add technical indicators to the chart. This is an **extension** of the “Show Chart” use case; it only becomes available when viewing charts.

 **Show news:**  
The user can view relevant financial news.

 **Transaction history:**  
The user can see their trading or account history.

## 8.3 Activity Diagram

This activity diagram illustrates the main steps users follow in the system and how the system reacts to each of those actions, emphasizing the flow from user interaction to system processing and signal generation.

A diagram of a diagram

AI-generated content may be incorrect.

**User**

1. **Login**  
   The user logs in to the system to access trading features.
2. **Decision**After receiving the system’s recommendation, the user makes a decision: Buy, Sell, or just observe (Record Observation).
3. **Buy / Sell / Record Observation  
   The user can choose to:**
   * Buy: Place a buy order.
   * Sell: Place a sell order.
   * Record Observation: Choose not to trade, just observe.

**System**

1. **Open Market**  
   The system opens the main market interface after the user logs in.
2. **Prees Start Sentra Model**  
   The user (or system) starts the Sentra Model to begin data analysis.
3. **Generate Composite Trading Signal**  
   The system combines news analysis and market data to generate a comprehensive trading signal.
4. **Present Recommendation to User**  
   The system shows the trading recommendation (buy/sell/hold) to the user.
5. **Record Trade**  
   If the user acts (buys/sells), the trade is recorded by the system.
6. **Update Portfolio**  
   After a trade, the user's portfolio is updated accordingly.
7. **Display Session Summary**  
   At the end of the session (if the user chooses to stop), the system displays a summary of the trading session.

**BERT**

1. **Analyze Financial News**  
   The BERT model analyzes relevant financial news articles to extract sentiment.

**VWAP Lane**

1. **Track Market Data**  
   The VWAP component tracks market data (like price and volume) in real time.

## 8.4 User-Friendly Interface

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

This interface provides real-time Bitcoin price analysis and smart trading signals, designed to help users determine when to buy or sell based on both market data and financial news sentiment.

* BTC Price (orange line) shows the actual price of Bitcoin over time.
* VWAP (dashed yellow line) is the Volume Weighted Average Price. It acts as a reference line to identify whether the current price is high or low compared to the market average.
* Sentra Signals (green and red dots) are smart indicators:
  + Green dots suggest a buy opportunity.
  + Red dots suggest a sell opportunity.

These signals are generated by combining:

* Technical data (price compared to VWAP),
* Financial news sentiment (from FinBERT or similar models),
* Model confidence levels.

The system shows a "BUY" button when all conditions are met:

1. Price is below the VWAP,
2. News sentiment is positive,
3. The confidence is high (e.g., above 80%).

The interface also includes portfolio details and allows users to execute market orders directly based on the system's recommendations.

This platform represents the practical application of our project, which integrates market indicators and AI-driven news analysis to support better trading decisions.

# 9. Common Problems

## 9.1 Technical Challenges

1. **Labeling Sentiment Data for Supervised Learning**
   * **Problem**: You likely don’t have pre-existing labels saying "this news caused price to go up/down" for all historical financial news relevant to Bitcoin/USD. Directly using price movement as a "label" for news sentiment can introduce noise and misattribution.
   * **Risk**: Wrong or noisy labels lead to a poorly performing or inaccurate model.
   * **Mitigation**: We will address this by:
     + Utilizing **pre-trained FinBERT models** that are already fine-tuned on general financial sentiment datasets. This reduces the need for extensive custom labeling.
     + For domain-specific nuances (crypto/forex), we will explore **transfer learning** with a smaller, carefully hand-labeled dataset specific to BTC/USD news, if necessary.
     + Focusing on **event-study methodology** during backtesting to correlate news events with *immediate* price reactions, rather than relying solely on simplistic labels.
2. **Real-Time Processing**
   * **Problem**: Combining real-time news scraping, sentiment analysis, indicator calculations, and signal generation in a low-latency manner is computationally demanding.
   * **Risk**: Latency will ruin your trading edge, as market opportunities can disappear quickly.
   * **Mitigation**: We will:
     + Optimize the data pipeline for **efficiency and speed**, potentially using asynchronous processing for data fetching.
     + Utilize **cloud-based services** (e.g., AWS Lambda, Google Cloud Functions) for scaling compute-intensive tasks like sentiment analysis, allowing for parallel processing.
     + Process data in **micro-batches** or streams rather than large chunks to reduce latency.
     + Focus on **hourly or 30-minute granularity** for signals initially, rather than tick-by-tick, to manage the processing load.
3. **Backtesting Errors (Lookahead Bias, Overfitting)**
   * **Problem**: If you inadvertently use future data during model training or backtesting, results will be invalid and appear much better than reality. Overfitting occurs when a model performs well on historical data but poorly on new, unseen data.
   * **Risk**: Your model will work on paper but fail catastrophically in real trading, leading to financial losses.
   * **Mitigation**: We will strictly adhere to best practices for backtesting:
     + **Time-series split**: Data will be split chronologically (e.g., train on 2020-2022, test on 2023).
     + **Walk-forward optimization**: Model parameters (e.g., signal weights) will be optimized on a rolling training window and then applied to the subsequent, unseen testing window.
     + **No future data leakage**: Ensure that any data used for training or parameter optimization does not include information from the testing period.
     + **Out-of-sample testing**: Always test the final strategy on completely unseen data.
4. **Integrating FinBERT in Production**
   * **Problem**: FinBERT is a large model (~110M+ parameters) and can be slow to run inference per text in a real-time, high-frequency environment.
   * **Risk**: Real-time processing slows down or crashes, making the system unusable for day trading.
   * **Mitigation**: We will:
     + Use **quantized or distilled versions of FinBERT** if available, which are smaller and faster while retaining most accuracy.
     + Implement **batch processing** for news articles where possible, processing multiple articles at once to utilize GPU/CPU efficiently.
     + Explore using **model serving frameworks** (e.g., TensorFlow Serving, PyTorch Serve) optimized for low-latency inference.
     + Utilize **pre-computation** where feasible (e.g., processing news sentiment every 5 minutes rather than continuously for every new headline if not strictly necessary).

## 9.2 Team Challenges

1. **Communication and Coordination**: Ensuring all team members are aligned on project goals, tasks, and deadlines.
   * **Mitigation**: Regular stand-up meetings, clear task assignments using project management tools (e.g., Trello, Jira), and dedicated communication channels (e.g., Slack, Discord).
2. **Skill Set Gaps**: Some team members might be stronger in NLP, while others excel in financial modeling or front-end development.
   * **Mitigation**: Cross-training sessions, pairing up members with complementary skills, and assigning tasks that play to individual strengths while providing opportunities for learning new areas.
3. **Time Management**: Balancing academic commitments with project development, especially when dealing with unforeseen technical issues.
   * **Mitigation**: Setting realistic deadlines, breaking down the project into smaller, manageable sprints, and allocating buffer time for troubleshooting.
4. **Data Access Issues**: Licensing and rate limits for commercial data APIs can pose significant hurdles.
   * **Mitigation**: Proactively researching and securing access to necessary APIs early, exploring open-source alternatives, and implementing robust error handling for API calls.

# References

**[1]** Bloomberg Terminal. Investopedia. Retrieved from <https://www.investopedia.com/terms/b/bloomberg_terminal.asp>

**[2]** AlphaSense – Market Intelligence & Search Platform. Retrieved from <https://www.alpha-sense.com/>

**[3]** Accern – AI-Powered Financial News Analytics. Retrieved from <https://www.accern.com/>

**[4]** Kavout – K Score Predictive Analytics. Retrieved from <https://www.kavout.com/k-score>

**[5]** Anese , P., Gagliardini, P., Pelizzon, L., & Stenbacka, S. (2023). *News-Based Prediction of Stock Movements Using NLP: Evidence from a Large-Scale Dataset*. SSRN. <https://doi.org/10.2139/ssrn.4631351>

**[6]** Bukhari , S. A. R., Shaheen, M., et al. (2023). *Fine-Grained Sentiment Classification of Financial News Headlines Using FinBERT*. SN Computer Science, 5(1). <https://doi.org/10.1007/s10287-023-00454-2>

**[7]** Zhang, H., Liu, Q., & Wu, Y. (2024). *Multi-Source Hard and Soft Information Fusion Approach for Accurate Cryptocurrency Price Movement Prediction*. arXiv preprint arXiv:2409.18895. <https://arxiv.org/abs/2409.18895>

**[8]** Thomas, M., & Chen, K. (2024). *Using Sentiment and Technical Analysis to Predict Bitcoin with Machine Learning*. arXiv preprint arXiv:2410.14532. <https://arxiv.org/abs/2410.14532>

**[9]** Zarattini, C., & Aziz, A. (2023). *Volume Weighted Average Price (VWAP): The Holy Grail for Day Trading Systems*. Concretum Research.

<https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4631351>