

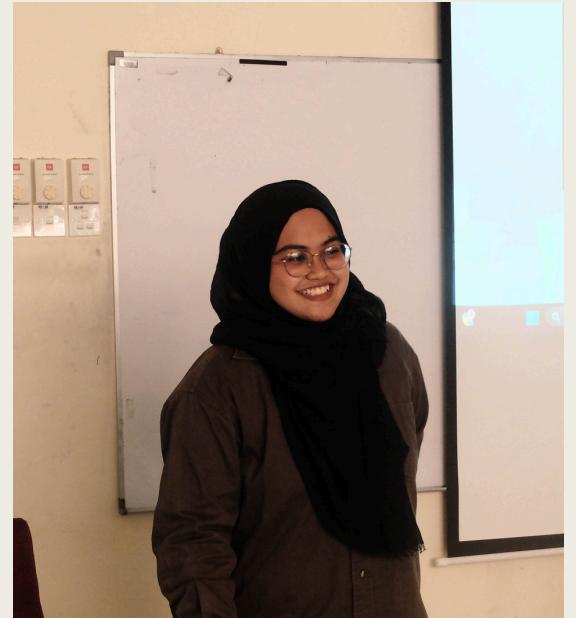
PITCH  
DECK



SHOWHER.EXE

# OUR TEAM

## AVID LEARNERS



Alysha  
Hannani

LEADER



Rihana



Siti  
Hajar



Hannah



Norlydia  
Farniza

# INTRODUCTION

## PROBLEM STATEMENT:



Too focused on price data only

Many quant models rely heavily on OHLC (open-high-low-close), ignoring deeper indicators like on-chain activity or sentiment.



Hard to integrate multiple data sources

On-chain data and sentiment (Twitter, Reddit) are powerful — but difficult to combine and test in a single platform.



Limited flexibility in strategy testing

Existing backtesting tools don't support both rule-based and machine learning-based strategies side-by-side with full customization.



## GOAL OF LIBRARY:

Build a customizable, backtestable trading framework , combines :

- market data
- on-chain indicators
- sentiment analysis

# KEY FEATURES

(01)

Integrates on-chain data, market data, and real-time sentiment

(02)

Supports both rule-based and ML-based strategy generation

(03)

Automatically computes key metrics: Sharpe Ratio,  
Drawdown, Trade Frequency

(04)

JSON/CSV reporting

(05)

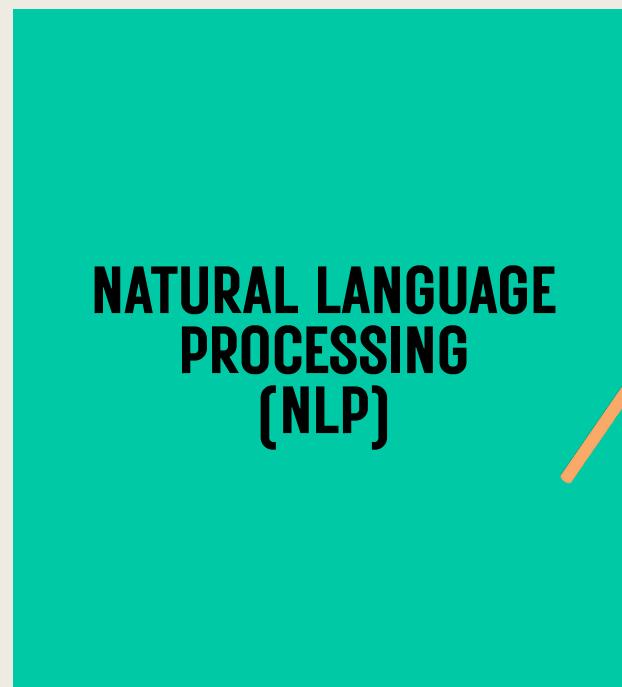
Extendable for multiple asset support (BTC, ETH)

(06)

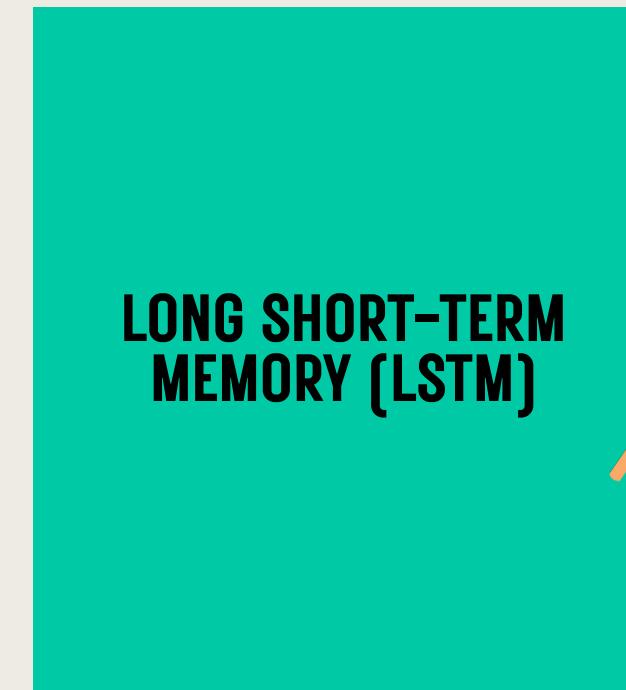
Visual output



# MACHINE LEARNING MODEL



Data Acquisition



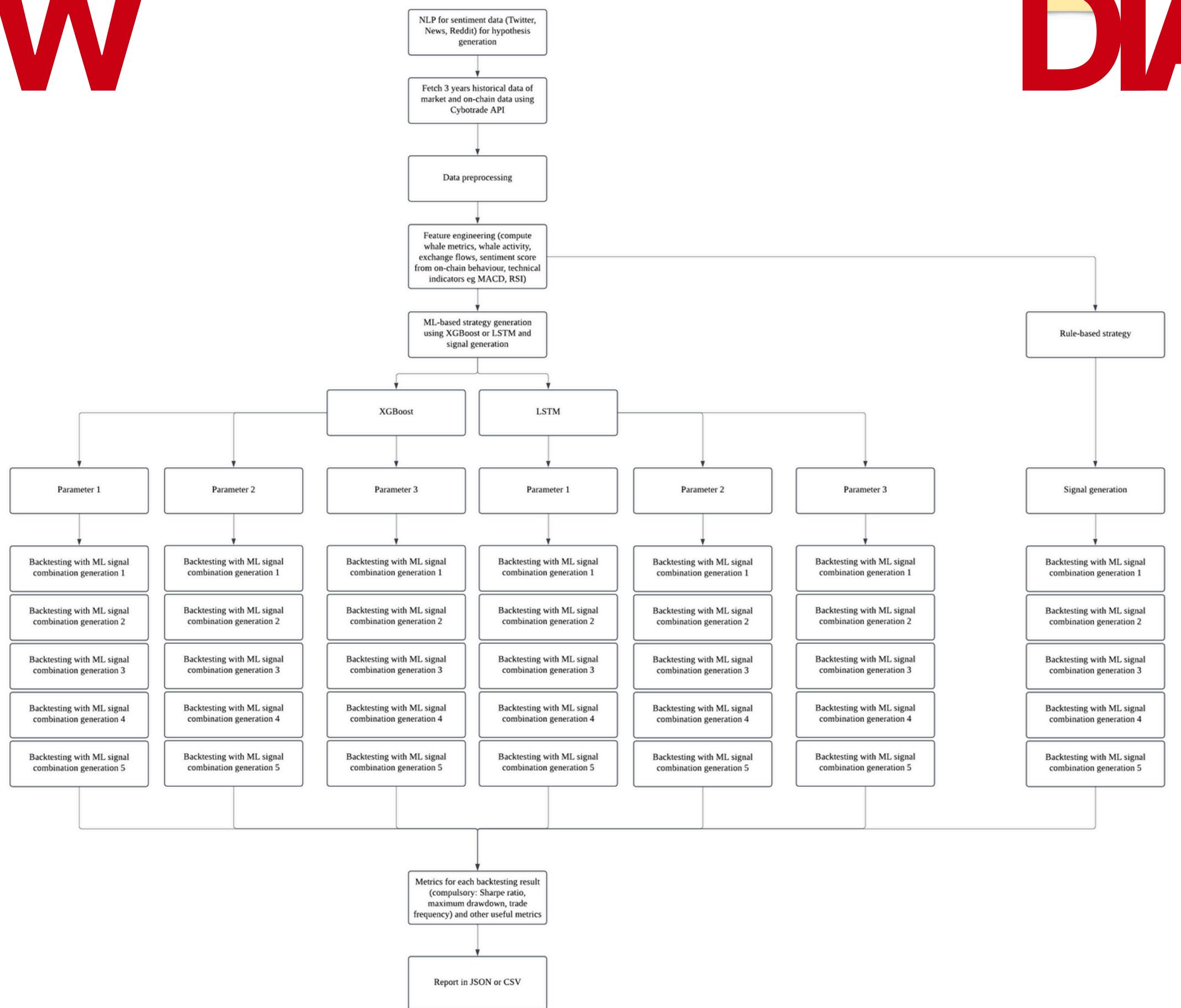
Strategy generator



Signal Generator  
Strategy Generator

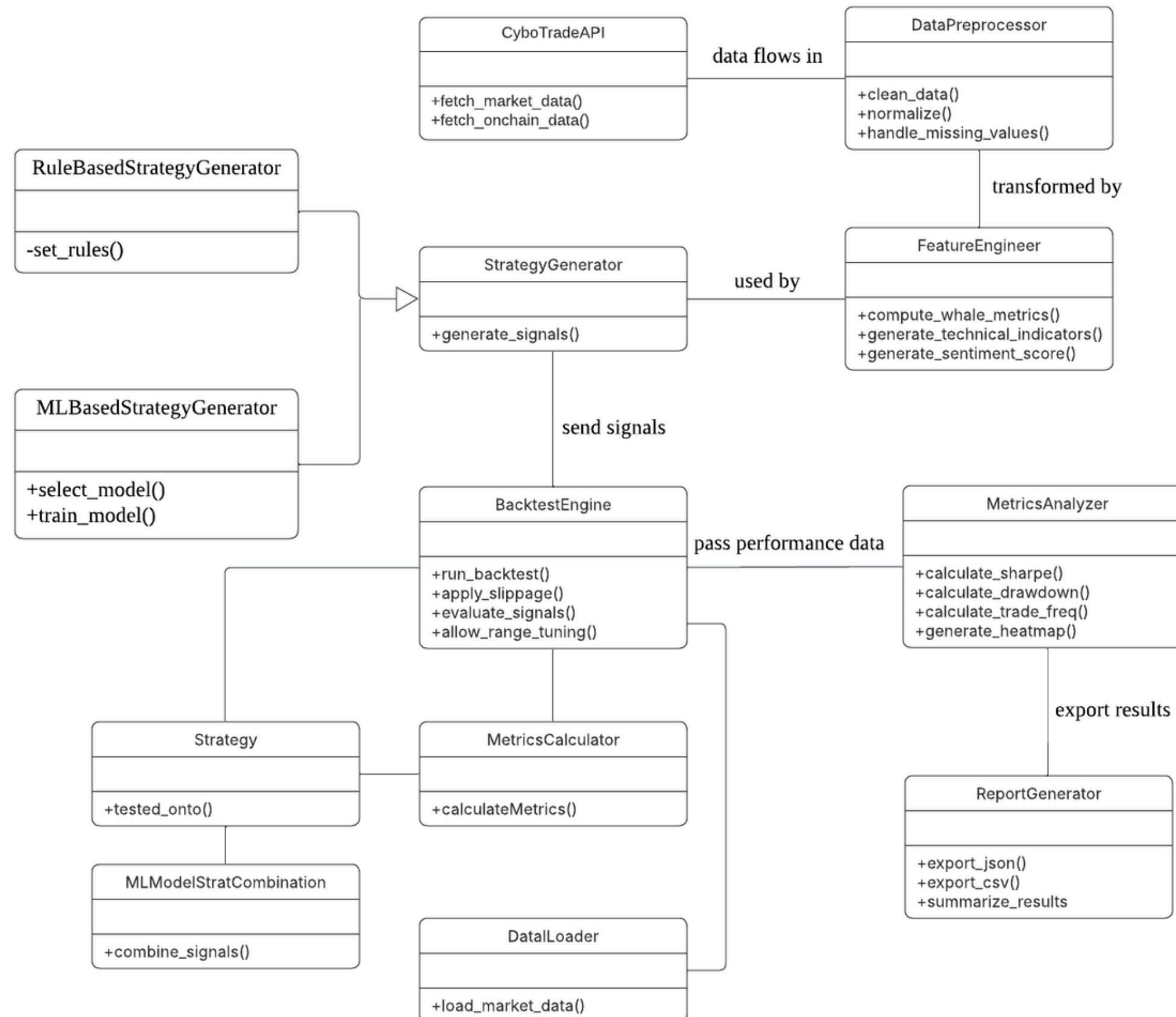
# FLOW

# DIAGRAM



# CLASS

# DIAGRAM



# OUR LIBRARY

```
# Train the model with explicit signals
specific_signals = ['EMA_20', 'EMA_50', 'RSI_14', 'MACD_Line', 'MACD_Signal', 'Bollinger_Upper', 'Bollinger_Lower']
backtester.train_model(lookback=20, forward_return_days=5, explicit_signals=specific_signals)
```

**Simplified training, only need to adjust parameters**

```
from backtester.backtester import MLBacktester

# Step 1: Initialize
backtester = MLBacktester()
```

```
# Find top combinations based on test data performance
backtester.find_top_combinations(n=5)
```

**simple usage**

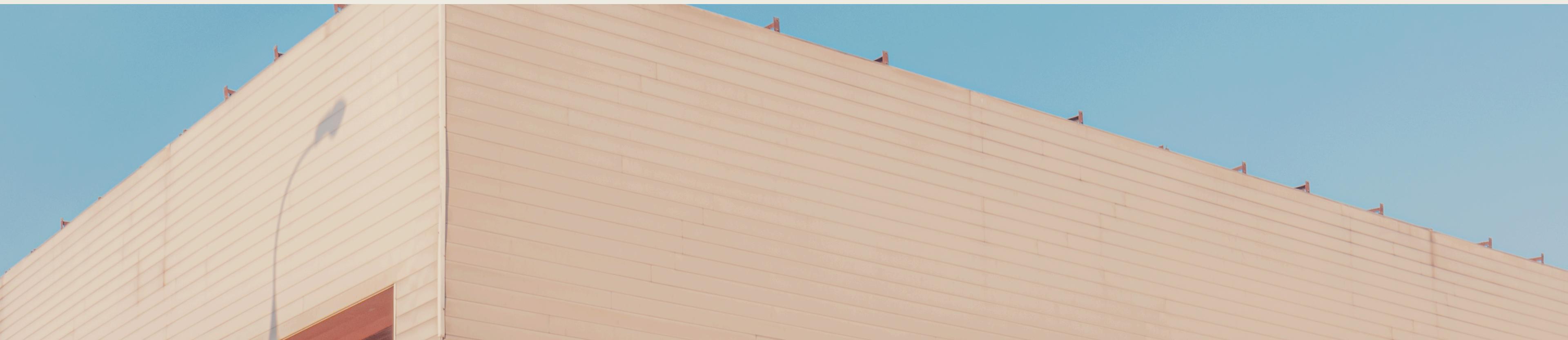
**Choose how many strategies to be generated with  
XGBoost ML model**

# UNIQUE SELLING POINTS

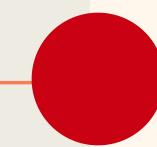
More than one ML model used for strategy generation, allowing more diverse data input

Considers sentiment data to understand market behaviour by integration of NLP in hypothesis generation

Further optimization of strategies using ML models by iteration to find better parameters based on market

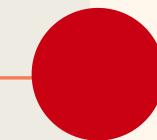


# CHALLENGES AND SOLUTIONS



## Ensuring Realistic Backtesting Conditions

We addressed lookahead bias by generating target variables based on future price movements shifted appropriately and by splitting our data into distinct training, testing, and forward testing periods.



## Avoiding Overfitting to the Backtesting

We implemented a separate forward testing period, distinct from the training and initial testing data. This allows us to evaluate the strategy's performance on data the model has never seen during training or parameter tuning, providing a more realistic assessment of its potential future performance.

# METRICS

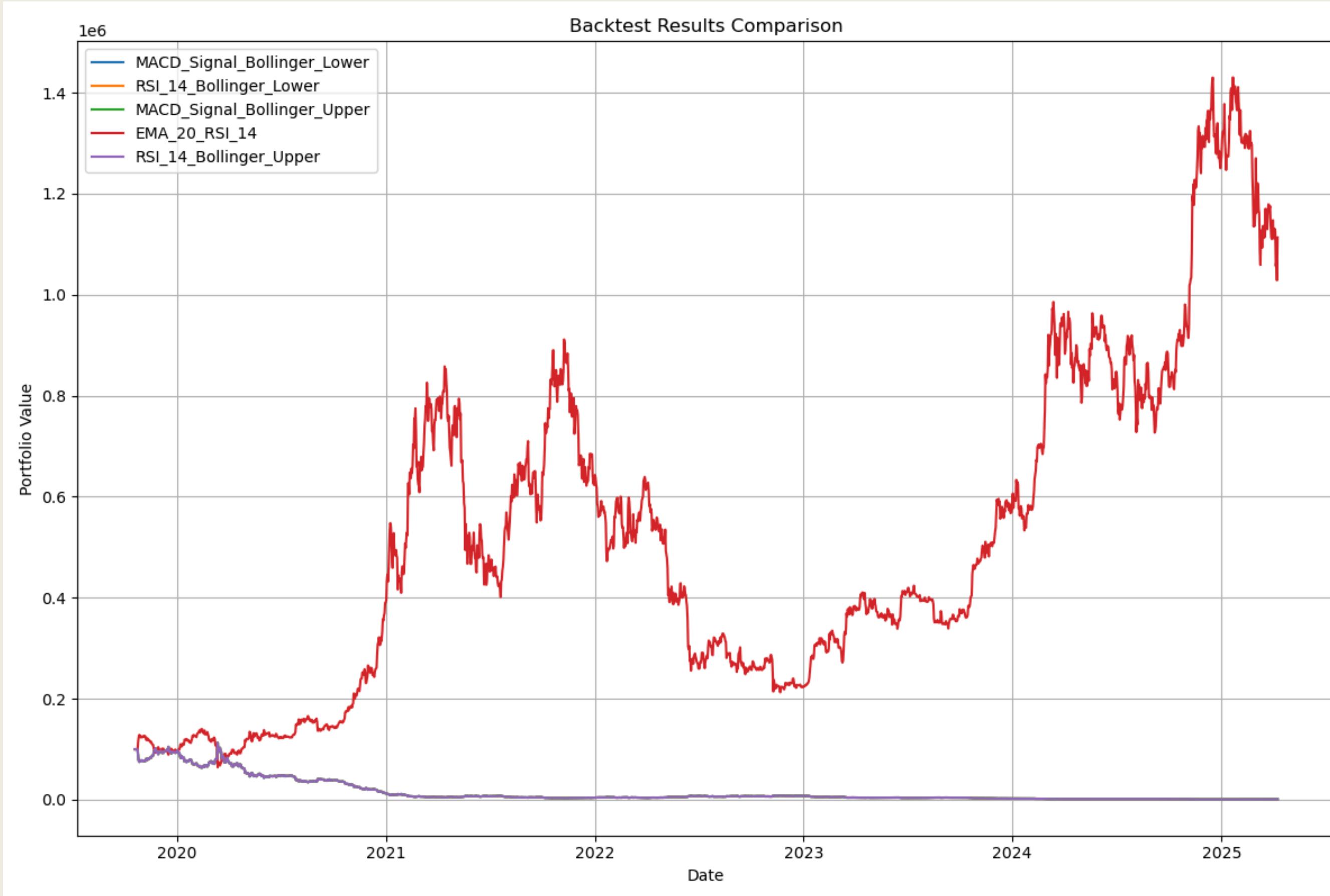
```
# Find top combinations based on test data performance
backtester.find_top_combinations(n=5)

Test feature matrix shape: (1975, 30)
Test target vector shape: (1975,)
Model performance on test dataset: accuracy = 0.80
Top 5 signal combinations based on test dataset performance:
1. MACD_Signal + Bollinger_Lower (Performance Score: 0.0821)
2. RSI_14 + Bollinger_Lower (Performance Score: 0.0809)
3. MACD_Signal + Bollinger_Upper (Performance Score: 0.0715)
4. EMA_20 + RSI_14 (Performance Score: 0.0711)
5. RSI_14 + Bollinger_Upper (Performance Score: 0.0704)
```

**Strategy combination 4 works best with Sharpe ratio of 0.84**

```
Backtest 1 completed: MACD_Signal_Bollinger_Lower
Total Return: -99.12%
Sharpe Ratio: -0.84
Max Drawdown: -99.35%
Trades: 0.5
Win Rate: 48.95%
-----
Backtest 2 completed: RSI_14_Bollinger_Lower
Total Return: -99.12%
Sharpe Ratio: -0.84
Max Drawdown: -99.35%
Trades: 0.5
Win Rate: 48.95%
-----
Backtest 3 completed: MACD_Signal_Bollinger_Upper
Total Return: -99.12%
Sharpe Ratio: -0.84
Max Drawdown: -99.35%
Trades: 0.5
Win Rate: 48.95%
-----
Backtest 4 completed: EMA_20_RSI_14
Total Return: 1013.39%
Sharpe Ratio: 0.84
Max Drawdown: -76.67%
Trades: 0.5
Win Rate: 51.05%
-----
Backtest 5 completed: RSI_14_Bollinger_Upper
Total Return: -99.12%
Sharpe Ratio: -0.84
Max Drawdown: -99.35%
Trades: 0.5
Win Rate: 48.95%
```

# METRICS



Strategy in red moves upwards, indicating that the strategy would still be relevant over the years

**POSSIBLE  
FUTURE  
ENHANCE-  
MENT**



- USE RICHER BACKTEST METRICS (E.G. ADD CALMAR RATIO, SORTINO RATIO, WIN/LOSS RATIO, PROFIT FACTOR)
- IMPLEMENT ROLLING WINDOW BACKTESTING TO SIMULATE LIVE TRADING ENVIRONMENTS
- ADDING MORE SENTIMENT DATA FROM VARIOUS SOURCES (E.G. SPEECH-TO-TEXT FEATURE FOR TWITTER SPACE, PODCAST, TIKTOK VIDEOS)
- BETTER TRAINING FOR ML BACKTESTING



**LET'S CREATE  
SOMETHING  
AMAZING TOGETHER**

