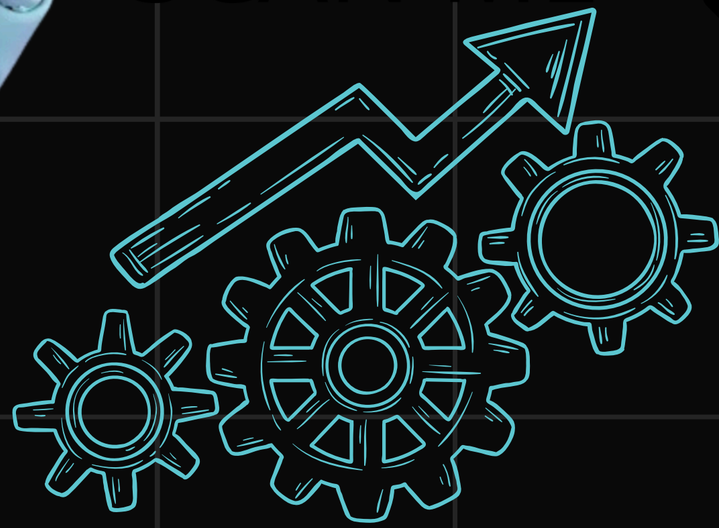
Domain Expert Models

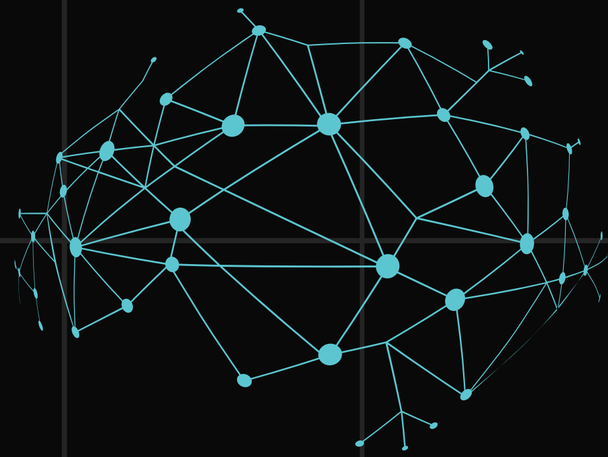
SCAN ME

https://dlweek-hackathon-team-natural-stupidity-domain-expert-models.streamlit.app/



Presented by **Natural Stupidity**

Typically, ML Models go with a “Bigger is Better Approach”



Inspired by SLMs, we went with an ensemble combination of different “Domain-Expert” Models

User Flow

User selects Domain Expert Model

User Inputs Ticker that they are interested in predicting the price

Upon input, backend calls yFinance to pull the Ticker’s recent price action

Sentiment analysis is conducted based on the past 7 days, provides a global perspective on an industry

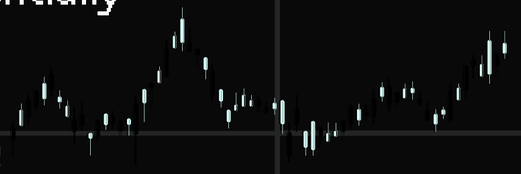
Model takes in Ticker’s recent price action as input, and predicts closing price for the day

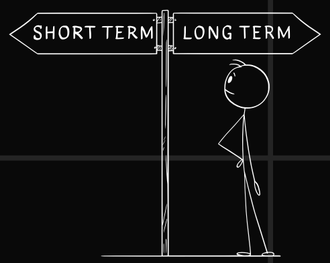
Display key statistics

Playwrite US Modern

Model Pipeline

User Input of Ticker essentially chooses which model to use

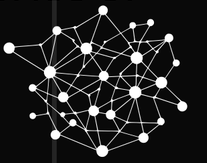




Multiple single variable LSTM models trained on past 3 year data of an individual ticker

Combine this with sentiment analysis model which analyzes the past week's news of how the industry is being valued

These form inputs for a Deep Neural Network, which outputs predicted stock closing price of a specific ticker



Minimizing Risk with Random Forest Classifier

Feature Engineering and Stock Analysis

- Extracts meaningful insights from raw stock data using technical indicators.

Model Implementation (Random Classifier Model)

- Preprocessing: Clean & transform data.
- Feature Engineering: Add technical indicators.
- Training: 80/20 split, model learns to predict Bullish (1) or Bearish (0) trends.
- Evaluation: Accuracy improvement through tuning.

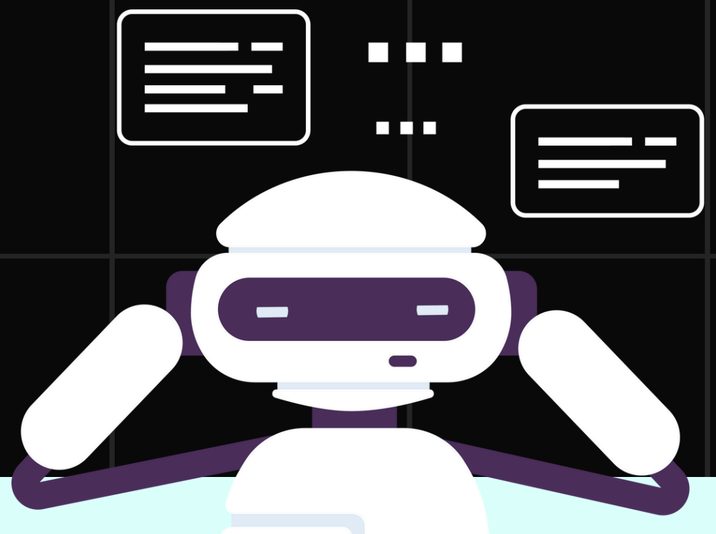
Important Technical Indicators

- Moving Averages: 10-day & 50-day trends.
- Volatility: 10-day rolling standard deviation.
- Momentum: 10-day price change.
- RSI: Measures overbought (>70) and oversold (<30) conditions.
- MACD: Identifies momentum shifts using exponential moving averages.
- Trading Volume: Confirms market activity.

```
Model Accuracy: 0.64

🔍 Classification Report:
              precision    recall  f1-score   support

 Bearish (0)       0.62      0.83      0.71         6
 Bullish (1)       0.67      0.40      0.50         5

    accuracy                           0.64        11
   macro avg       0.65      0.62      0.61        11
weighted avg       0.64      0.64      0.62        11
```

Results

Addressing Overfitting of “Domain Expert” Models



We prioritized Technical Indicators	The Fundamentals of Price Prediction
Implemented Feature Normalization	Prevent one factor from dominating others
Simple Architecture combined with an Ensemble Format	The simplicity prevents overfitting, while ensemble models still provide nuance in predictions

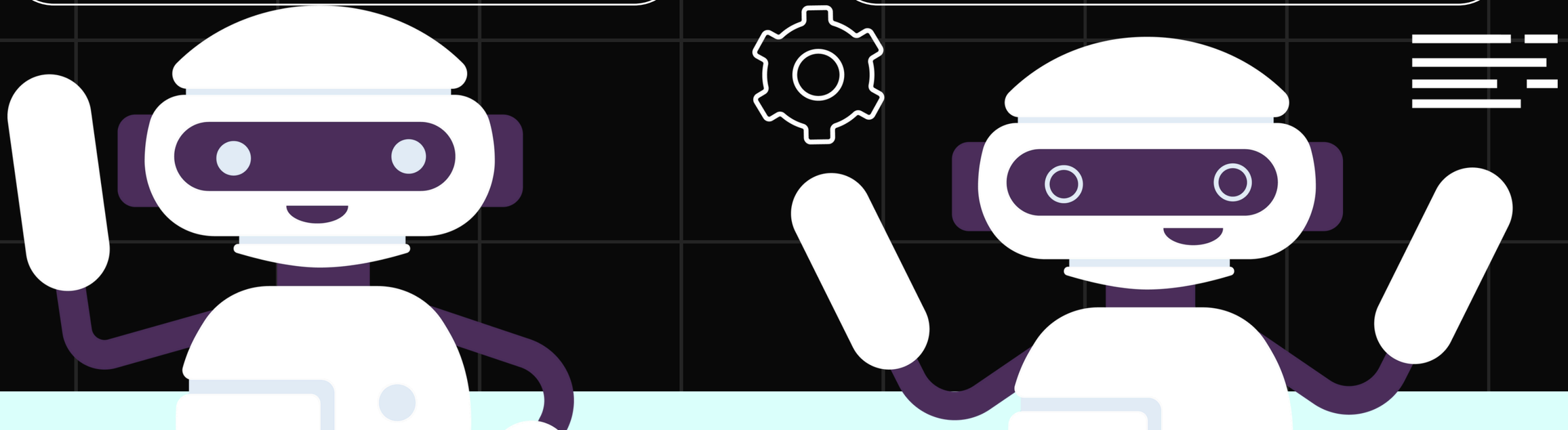
Deep Neural Network

The Endpoint of the Ensemble System

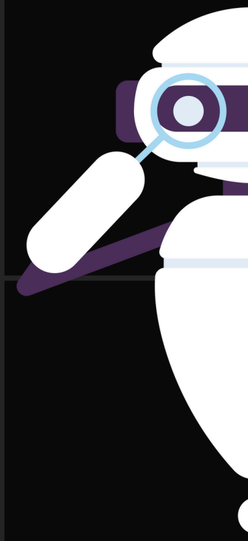
With all the different models contributing, the DNN aggregates all their results to provide one final input.

What is Deep Learning?

A subset of machine learning using neural networks with many layers, capable of learning representations of data with activation functions.



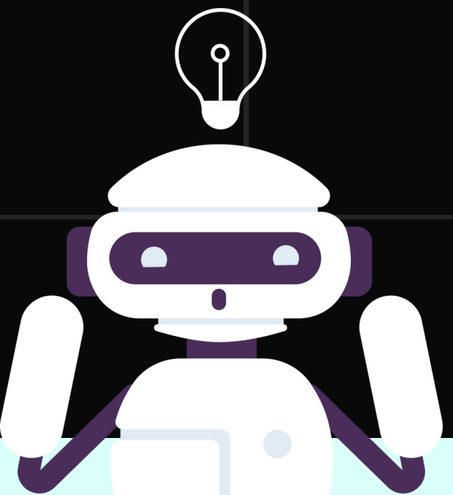
Data Scraping



Data Quality

AI systems heavily rely on quality data for training and decision-making.

We used crawlers to gather social media data from Financial Subreddits and Telegram Stock Channels, before cleaning and processing the data for training sentiment analysis models

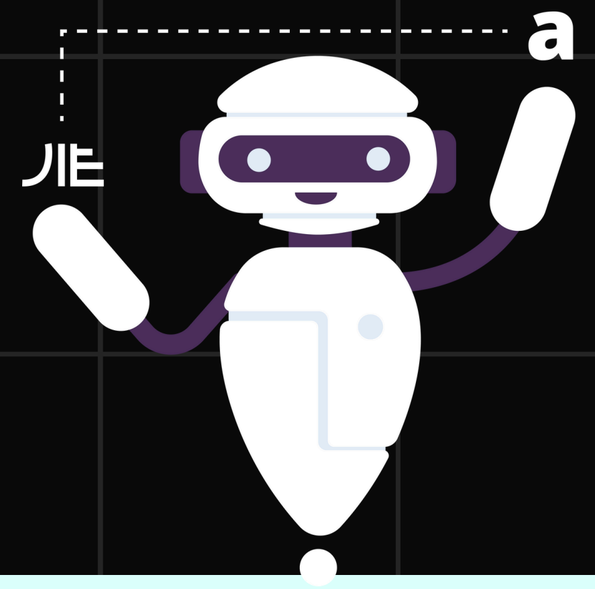


Sentiment Analysis

```
'Exciting new AI chip announced by NVIDIA! Stock price expected to surge.' - Score: 0.99
'Tech stocks are crashing after disappointing earnings reports.' - Score: -0.40
```

We used a RoBERTa model, optimized for Twitter Sentiment, on our Telegram and Reddit datasets.

Our sentiment analysis model was too large for github - with 340 Million Parameters



https://huggingface.co/raz-1412/tech-stock-sentiment-analysis/tree/main

Thank you