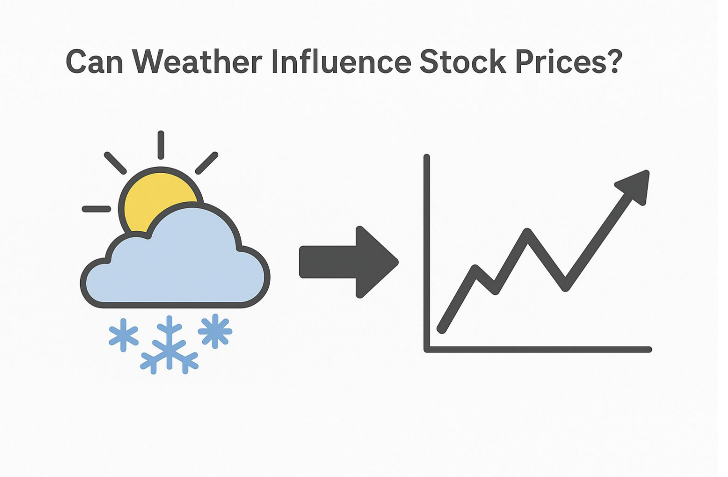
Final Project: Forecasting Stock Trends Using Weather and Market Data with Attention-Based Models

Jingchen Yan, Liyuan Zhang

May 06, 2025

Problem

- **Weather** is often overlooked in financial modeling.
- Certain stocks (e.g., natural gas ETFs, utilities) may respond to weather conditions.
- Can **meteorological data** improve stock trend prediction?
- We aim to build an **attention-based time series** model combining weather and market data.



Algorithmic Solutions

We tried both regression and classification methods

We framed the task as a 3-class classification problem:

- Label = "up", "down", or "flat", according to a certain threshold.
- Based on future 5-day stock price movement

We designed and compared several models:

- Random Forest (baseline)
- LSTM classifier: captures sequential dependencies
- Transformer: models global interactions across features
- Cross-Attention Transformer: separates stock and weather features, enabling weather to attend to price

Models were trained on lagged features from historical price and weather, optionally including future weather forecasts.

Algorithmic Solutions

LSTM Classifier:

- Input: Full feature vector (stock + weather)
- 1 LSTM layer → Global max pooling → Dense → Softmax

Transformer Classifier:

- Input: Full feature vector
- Embedding layer → TransformerEncoder → Average Pooling → Dense

Cross-Attention Transformer:

- Input split into two:
    - Stock features → Stock Encoder
    - Weather features → Weather Encoder
- Cross-attention: Weather attends to Stock
- Fusion → Dense → Softmax

Datasets

Time Range & Alignment

Timeframe: July 2016 – February 2025

Data aligned by trading days using business calendar (pandas `bdate_range`)

Weather data

Source: Meteostat Python API

Features: Daily minimum & maximum temperature, precipitation, snowfall

Locations: New York, Chicago, Dallas, Boston, Minneapolis

Datasets

Stock Market Data

Source: Alpha Vantage API (`TIME_SERIES_DAILY` endpoint)

Tickers: UNG, UNL (natural gas ETFs)

Feature: Daily closing price

Feature: Price and weather data for the previous five days

Target: Stock Price (Regression),

Trend (classification, 3-classes: surge, plunge, or remain stable)

Training Methodology and Experimental Setup

Task: 3-class classification

Features: Lag features from past 5 days of stock + weather data,

Optional inclusion of forecasted weather (future 1–3 days)

Labeling strategy: “up” if price rises more than +5%,

“down” if price drops more than –5%,

Otherwise, “flat”

Training Methodology and Experimental Setup

Model training:

Optimizer: Adam

Loss: CrossEntropyLoss

Epochs: 100 (depending on model)

Device: CPU or Google Colab GPU

Tools & Libraries:

PyTorch, Scikit-learn, Pandas, Meteostat API, Alpha Vantage

Results and Comparison

Core metrics:

- Accuracy
- F1 Score
- Confusion Matrix (focus on up/down)

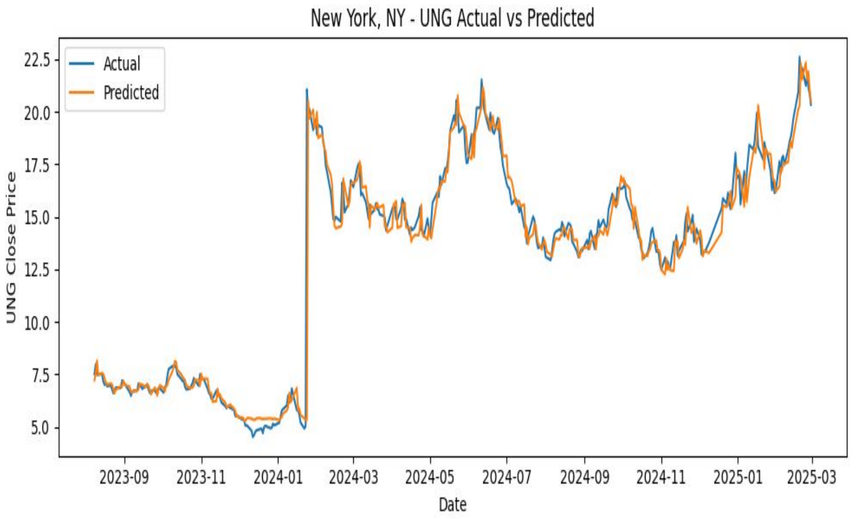
Key priority:

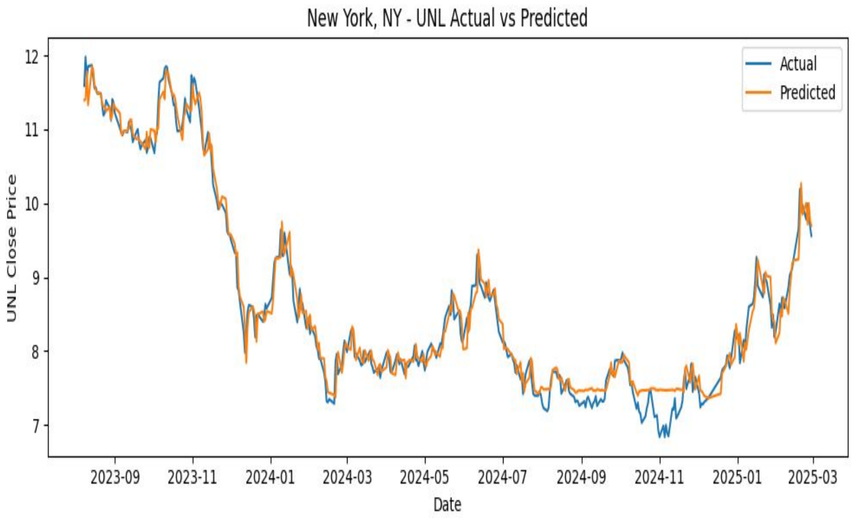
- Detecting **sharp upward or downward movements**

- Correctly identifying “up” and “down” is more valuable than maximizing overall accuracy

Results and Comparison

Regression: not the best way for this task





Results and Comparison

Classification

Evaluation without weather

	MLP	LSTM	Transformer	Cross atten
Accuracy	0.39	0.39	0.38	0.32
precision(up/down)	0.39/0.47	0.42/0.40	0.44/0.38	0.00/0.30
recall(up/down)	0.09/0.23	0.21/0.40	0.06/0.40	0.00/0.32
f1-score(up/down)	0.16/0.30	0.28/0.40	0.11/0.39	0.00/0.31

Results and Comparison

Classification

Evaluation with weather

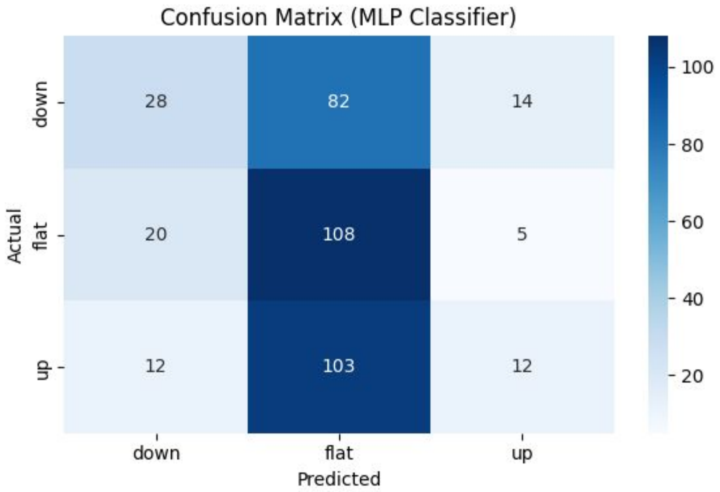
	MLP	LSTM	Transformer	Cross atten
Accuracy	0.44	**0.48**	0.42	0.44
precision(up/down)	0.40/0.40	**0.47/0.51**	0.42/0.38	0.41/0.44
recall(up/down)	0.40/**0.42**	0.31/0.40	**0.43**/0.31	**0.47**/0.28
f1-score(up/down)	0.40/0.41	0.38/0.44	0.43/0.34	**0.44**/0.34

Results and Comparison

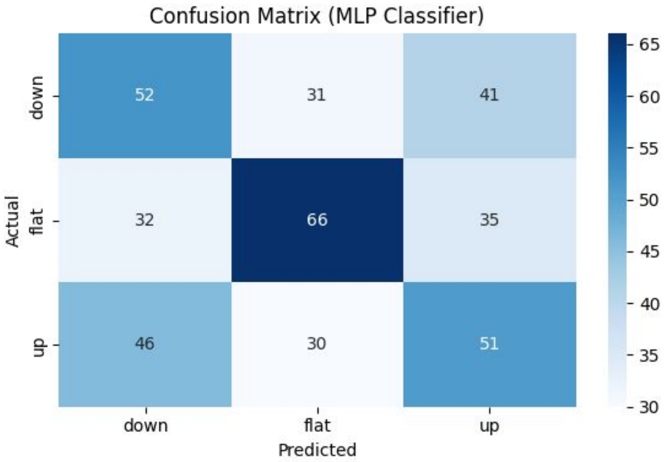
Classification

MLP with and without weather features

without weather features



with weather features

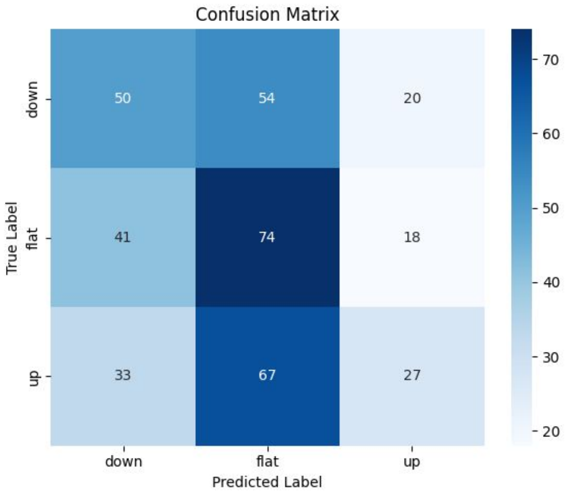


Results and Comparison

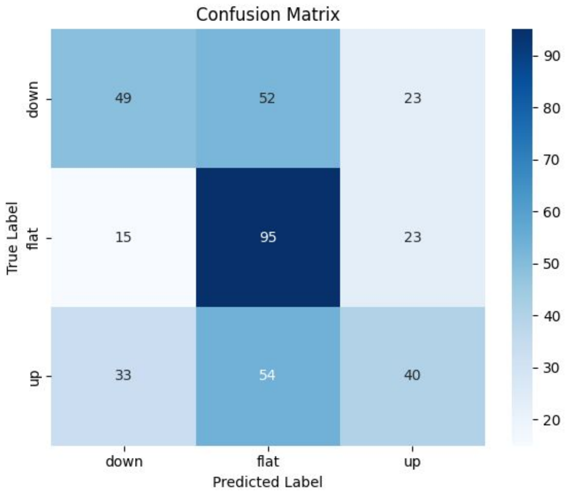
Classification

LSTM with and without weather features

without weather features



with weather features

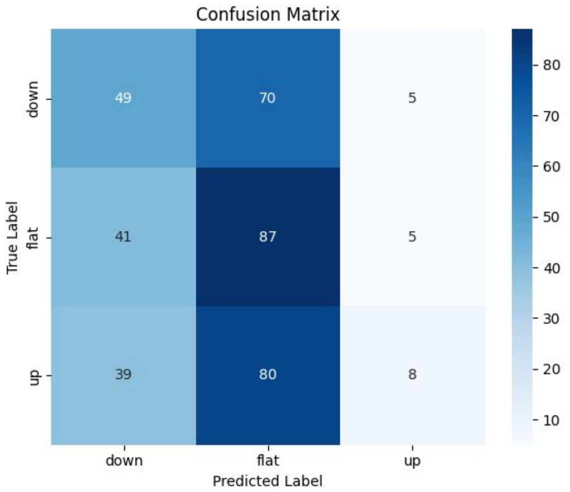


Results and Comparison

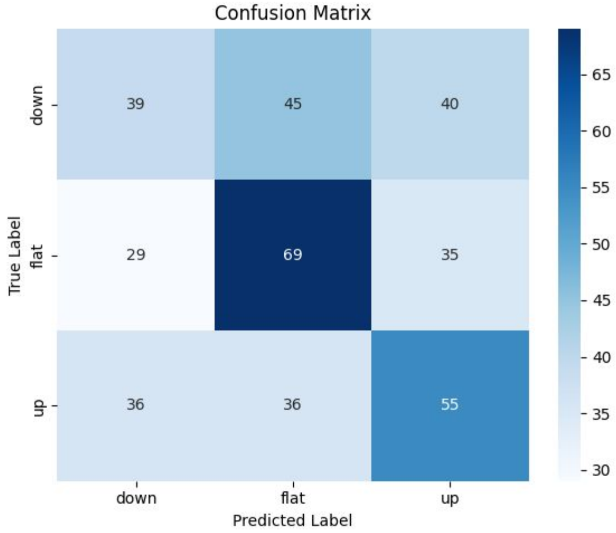
Classification

Transformer with and without weather features

without weather features



with weather features

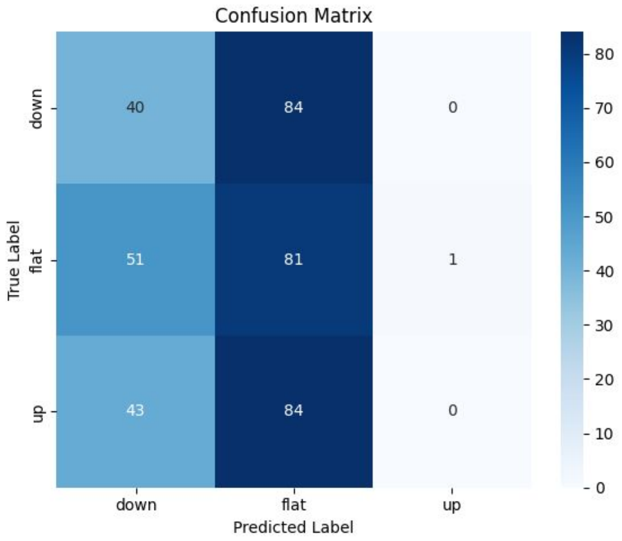


Results and Comparison

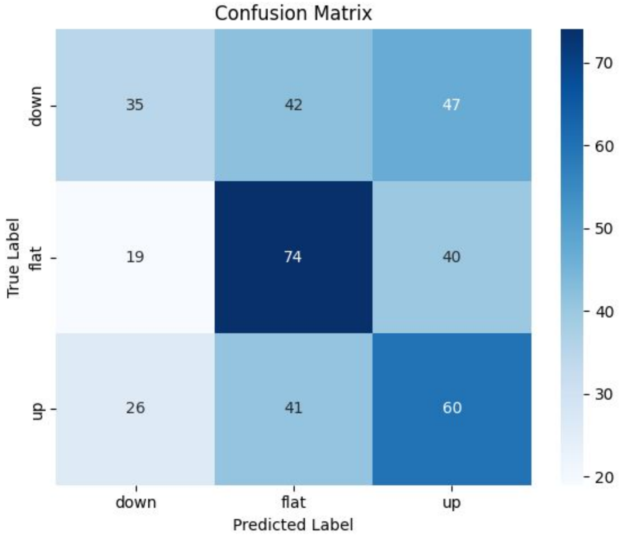
Classification

Cross Attention Transformer with and without weather features

without weather features



with weather features

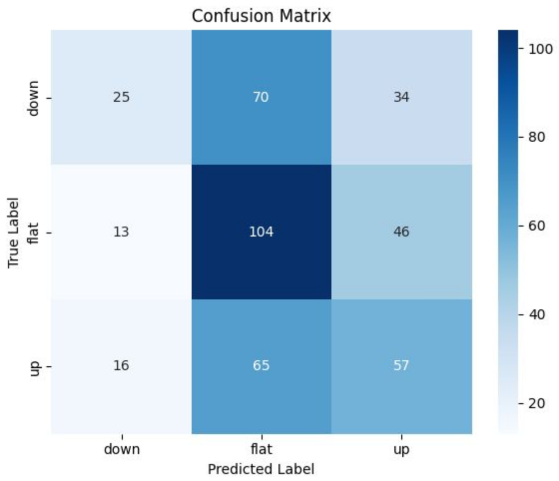


Results and Comparison

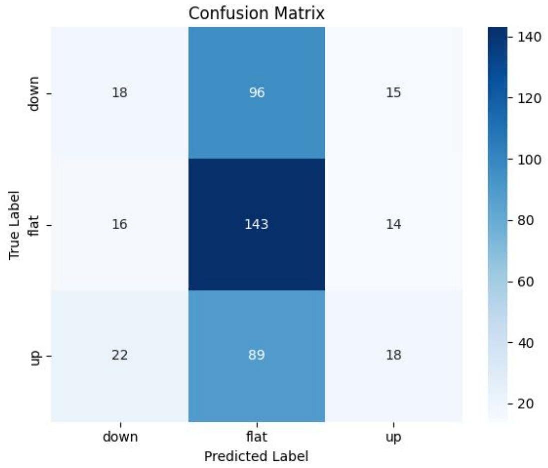
Classification

Use Other cities

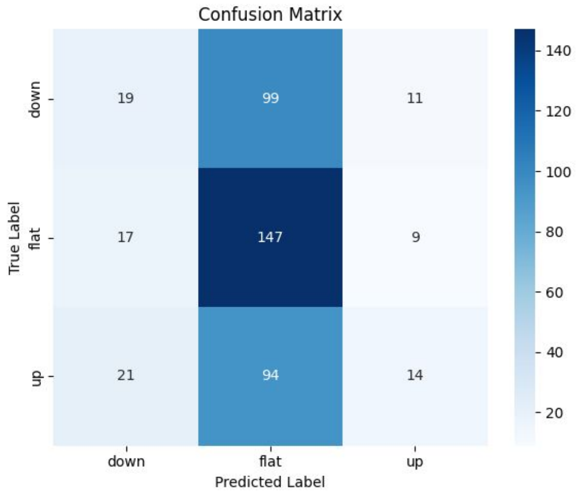
Minneapolis



Dallas



Boston



Lessons Learned

Regression-based models can fit curves well, but may fail to capture future shifts in price. Even when sharp changes are predicted after they happen, the MSE can still be low—this limits practical value.

Models trained with weather features consistently outperform those trained on price data alone.

Among all tested cities, New York’s weather data led to the most accurate stock movement predictions.

Weather data offers complementary signals that improve trend detection—especially for weather-sensitive assets.