Quantifying Qualitative Financial Announcements

A quantamental pipeline using social-media news, lexicon sentiment, LLM ticker mapping, and ML backtesting

By Kelvin Wang

Advised by Professor Yoonsang Lee

Abstract

Quantitative trading systems have evolved over the past several decades to take advantage of modern machine learning (ML) techniques. This improvement has seen major increases in performance and the rise of “quant” firms in the industry but are often criticized for becoming abstracted from real market context. Trading signals have been reduced to purely numerical signals and have lost their interpretability. In contrast, traditional fundamental analysis emphasizes qualitative events such as CEO changes, regulatory shifts, and financial announcements to inform their decisions.

This project proposes an approach that converts qualitative announcements from influential financial news sources into quantitative features. We scrape financial social media accounts for tweets, compute a finance specific sentiment score, map the content to impacted S&P 500 tickers using a large language model (LLM), attach market features around the event time, and train ML models to predict future price changes. Performance is evaluated both with standard error metrics and with a portfolio-like trading simulation to demonstrate how the model would perform under real world conditions.

Beyond predictive metrics, the design emphasizes the tension between interpretability and performance. Financial stakeholders often need a narrative justification for trades beyond what purely quantitative strategies can easily explain. By attaching trades to specific events with scores that can be easily interpreted in the market context, we can better understand the impact/correlation financial news sources have with actual market change. Additionally, by using an LLM strictly as a structured mapping layer (with strict input and output rather than allowing unconstrained generation), the data pipeline preserves interpretability, minimizes hallucination, and expands semantic coverage.

Keywords: financial NLP, sentiment analysis, ticker mapping, event-driven trading, machine learning, LLMs

1. Introduction

1.1 Quantitative vs Fundamental

Quantitative approaches have become dominant in modern trading due to their ability to scale, systematically test hypotheses, and exploit advances in statistical and machine learning. In particular, the ability of modern quantitative trading systems to execute trades with extreme volume and frequency has allowed them to overtake traditional approaches in prevalence. The common characteristic of the model being detached from the market narrative serves as both an advantage and disadvantage in this sense. Being agnostic about the specific market allows the same general model to be applied across many markets without major customization. But by not being personalized to the specific market, the interpretability of the trading decisions has been mostly lost, they also solely rely on historical numerical features and neglect to incorporate valuable contextual information.

Traditional fundamental approaches, by contrast, react to qualitative information and use mostly human decision-making to inform the final decision. This requires a lot of research about aspects such as leadership changes, earnings guidance, geopolitical events, and policy announcements to then attempt to reason about their mechanisms and impacts. This approach also incorporates historical price changes and financial metrics like quantitative approaches do, but the final decision relies mostly on human intuition and reasoning. As such, these approaches have not taken full advantage of modern advancements in mathematical modeling.

1.2 The ‘quantamental’ approach

The goal of this project is to bridge this gap by applying quantitative ML modeling to qualitative financial announcements. Specifically, we seek to take the textual information used by the fundamental approach and apply the quantitative approach.

This approach of using quantitative analysis on fundamental data has already been implemented in many firms under a branch known as “quantamental” or “quanta.” Quantamental research attempts to combine the strengths of both; to encode qualitative information in a way that can be measured, modeled, and validated by quantitative approaches. They often are focused on official data such as financial news reports, earnings calls, and policy announcements. This project focuses on social media content, particularly information that is more likely to be consumed by the average person.

1.3 Research Questions

RQ1: Can short-form financial news tweets be transformed into structured features that predict short-horizon price movement?

RQ2: Does explicit text to ticker mapping improve learnability compared to sentiment-only features?

RQ3: Which loss/objective functions are most appropriate under noisy, heavy-tailed return labels?

2. Literature Review

Research relevant to this project spans market efficiency, investor attention, media sentiment, and event-driven prediction. Because short-horizon returns are noisy and volatile, the most useful prior works often emphasize either careful text processing and domain specificity, or evaluation frameworks that reflect trading objectives.

Unfortunately, most techniques used by quantamental branches in industry are not published as they contain firm-specific trading secrets and strategies. As a former intern and incoming employee of one of these firms, I have a legal obligation to refrain from implementing any proprietary techniques, but the overall approach of data acquisition, processing, modeling, and evaluation is not firm specific and therefore is implemented in this project.

2.1 Media tone and investor attention

Tetlock (2007) studied the relationship between media content and market outcomes. They used a large corpus of Wall Street Journal "Abreast of the Market" columns to study whether media tone could explain market behavior. They operationalized pessimism via text features and tested association with market returns and trading volume. Results indicated that unusually negative tone forecasted downward pressure and higher future volume. This was consistent with sentiment/attention channels affecting price when incorporation was not instantaneous. In other words, sentiment features were relatively positively correlated in markets where price change was measured with a lag.

These results suggest that for this project, assuming the sentiment scores are evaluated correctly, there should be a correlation with the lagged market price change. In particular, the sentiment

score should consider including polarity (positive and negative scores) and the salience of the content (views, likes, retweets, etc.) can be measured as a feature as a proxy for attention.

2.2 Finance-specific dictionaries and lexical methods

Loughran & McDonald (2011) evaluated the efficacy of a dictionary-based sentiment evaluator on financial documents (primarily legal documents such as 10-Ks). They compared generic sentiment wordlists to a finance-tailored lexicon. Their setup measured whether word-category counts explain returns and other outcomes better than generic sentiment lists. They found that generic dictionaries systematically misclassified financial vocabulary which motivated their creation of a domain-specific dictionary with additional categories such as uncertainty and litigiousness that are better suited for finance-related terms.

This project will be using the Loughran-McDonald lexicon for finance-specific sentiment evaluation alongside the generic VADER sentiment analysis. The Loughran-McDonald lexicon was created primarily for use on legal documents, but many of the added financial terms make frequent appearances in our textual content of interest.

2.3 Social media, mood, and markets

Bollen, Mao, and Zeng (2011) tested whether aggregate Twitter mood could predict broad market movement. Their experiment converted large-scale tweets into daily mood indicators (such as calmness, unrest, etc.) and evaluated predictive relationships with subsequent market index changes. Results found that certain mood dimensions improve forecasts relative to baselines, suggesting that public social-media communication can contain measurable information relevant to predicting market direction. However, later sections highlight the volatility of the information as the effects were found to be highly sample-dependent.

Due to the volatile nature of social media and market reactions, prediction may not always follow consistent patterns. Errors in prediction can be attributed to many things aside from model imperfections: volatility in popular sentiment, external market influences, randomness of human trading, etc.

2.4 Event-driven prediction and entity linking

Ding et al. (2015) framed financial news as structured events with several recorded parameters (associated individuals, directionality of action, sentiment of actions) in a graph-like network. They

then tested event-driven predictors for market movement. Their setup extracted event representations from news and learned predictive graph models. They found that event structure and entity-level alignment had limited improved performance and could only outperform sentiment-only baselines. Although their results suggested limited impact, their setup motivates our text to ticker mapping step in the processing pipeline. Ensuring the entity linking step is correct establishes the premise for supervised learning by making sure the label is attached to the correct stock.

2.5 Microblog information content (ticker-linked tweets)

Sprenger et al. (2014) studied whether stock-related microblogs contained tradable information by focusing on posts that explicitly referenced firms/tickers. The experiment linked microblog sentiments and message volumes to temporal and subsequent stock returns. They found statistically significant relationships, particularly around attention and sentiment, to support the premise that short-form social text can be informative when mapped correctly.

3. Methods

3.1 Data Types

The relevant data types are as follows:

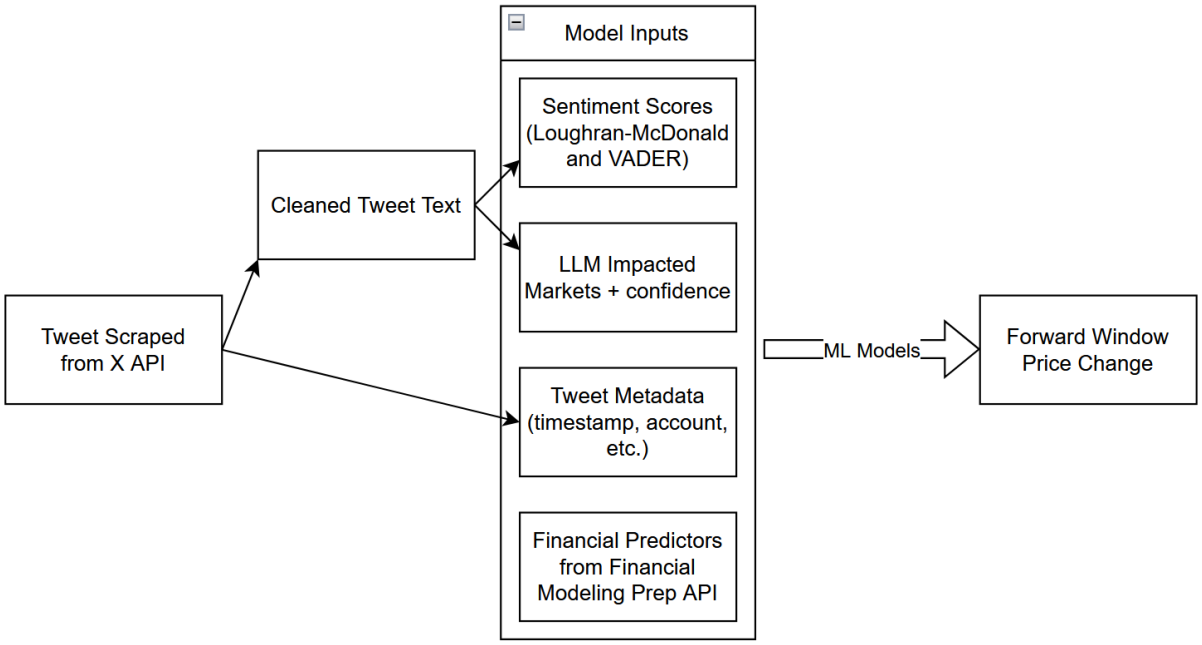
- Tweets and metadata scraped: text, timestamp, account, engagement metrics (likes, retweets, replies, quotes, impressions)
- Sentiment features calculated: Loughran-McDonald categories (Negative, Positive, Uncertainty, Litigious, Strong Modal, Weak Modal, Constraining, Complex) and VADER score
- LLM outputs: list of impacted S&P 500 tickers with a direction label with the sentiment (positive/negative) and confidence score
- Market features: price, total volume, lagged price, delta, and moving averages based on the tweet timestamp; intraday bars for forward-window backtesting

X accounts were selected to represent high-impact financial news sources whose posts are frequently consumed by market participants: @business, @Reuters, @markets, @WSJ, @FT, @MarketWatch, @barronsonline, @SeekingAlpha, @Deltaone, @financialjuice

3.2 Simple pipeline

The pipeline follows the sequence: scrape tweets → clean text → compute sentiment scores → connect content to impacted markets using an LLM → scrape financial predictors at and before the event time → train ML models to predict price change → evaluate accuracy and simulate trading.

A simplified version of the pipeline is depicted below:



3.3 Methodology Details

Additional details of what each processing step entails are included below:

- Tweet scraping: calls X API to pull posts from each account's timeline. Each returned content piece includes the uncleaned text, timestamp, account, and engagement metrics.
- Text cleaning: removing unprocessed characters, removing hyperlinks, and standardizing characters.
- Sentiment calculation: tokenizing cleaned text, removing stopwords, computing Loughran-McDonald category counts, and calculating VADER score.
- LLM stock connector: GPT 5.4 mini is fed a prompt with the cleaned text to output a JSON with tickers, direction, and confidence; progress is checkpointed and recorded to comply with rate limits and for accuracy rerun checks.

- Financial predictors: at the timestamp of the tweet and for an associated stock from the LLM, call the Financial Modeling Prep (FMP) API for price, total volume, lagged price, delta, moving average, and forward window price. For premium-only symbols and legacy S&P 500 tickers, a separate API call is made.
- ML Modeling: regress delta_price (change in price) using sentiment + engagement + market context + LLM confidence and categorical indicators.
- Backtesting: interprets predictions as $1 long/short trades and compute the reward from the forward market move (12-hour window).

3.4 Formal supervised learning problem

Each observation corresponds to a content item i with a time t that is mapped to ticker s. We form a broad feature vector $x_{\{i,s,t\}}$ which contains parameters of sentiment categories, engagement, mapping confidence, and financial market context. The target $y_{\{i,s,t\}}$ is the realized price change over a fixed horizon (delta_price over a 12-hour period). The learning problem is to estimate the function $f(x) \approx y$ that generalizes to future content and market conditions.

3.5 Error function hypothesis

Return distributions tend to exhibit heavy negative tails. This is because market reactions are more likely to show drastic negative change in the short-term when presented with a content piece. Squared error objectives are susceptible to being influenced by these large moves and can lead to unstable fits. Robust objectives (such as Huber and absolute error) reduce sensitivity to outliers. Quantile regression changes the prediction from mean to quantiles which can improve stability at the cost of underfitting extremes. We hypothesize that given the nature of market reactions and the already unstable nature of the data/model, the robust objectives and quantile regression are likely to demonstrate higher returns in the simulated trading.

4. Data Collection and Processing

Tweets were collected using a paid X API subscription. Each account was capped at pulling 500 tweets to control cost and rate limits. After scraping, the tweets ranged from beginning on April 1st, 2025 to May 1st, 2026. Tweets were scraped with frequency depending on their creation date to get a better spread of representative data. Generally, the sampling gets much denser as the creation date approaches the present. This is because when attempting to create a training set that will best represent the testing set, there are two main criteria we sought to fulfill. Firstly, we want to include

more recent tweets as market sentiment is highly influenced by recent news. Secondly, we want to include data that occurred on the same date range within the year. This is motivated by industry theory that markets have cyclical habits. For instance, markets will act similarly during certain seasons every year or during the holiday seasons. On a smaller scale, industry models will even account for the day of the week and even time during the day. These shifts in tendency are due to things like the approaching weekend or end of trading hours encouraging traders to close their open positions. Unfortunately, we did not have the resources to scrape enough data to meaningfully examine smaller influences like day of the week or time of day, but by at least including data from at least a year ago, the model can include influences stemming from the annual cycle of habits.

Fields scraped include tweet text, created_at timestamp, username, and public metrics of likes, retweets, replies, quotes, and impressions.

Text processing involved removing URLs, mentions, hashtag symbols, punctuation, and extra whitespace to improve token matching, reducing noise, and removing proper noun references. Sentiment is computed using the Loughran-McDonald dictionary across its categories. The VADER score is also calculated.

For each sample, market metrics are scraped using the Financial Modeling Prep (FMP) API. A 12-hour forward window is used for the backtest to reflect short-horizon reaction time to news while still allowing enough time to capture delayed assimilation and spillover reactions.

4.1 Time alignment and choosing the 12-hour horizon

Choice of horizon is a modeling decision and has both statistical and economic implications. Too short of a window can be dominated by microstructural noise and immediate repricing, failing to capture enough of the market's reaction. Too long of a window increases the chance of capturing confounding and unrelated news. A 12-hour window was decided as a pragmatic compromise. It attempts to capture the market's reaction within the same trading day (or through the overnight session) while reducing sensitivity to minute-level noise.

4.2 LLM reliability controls

The GPT 5.4 mini model by OpenAI was prompted to connect textual pieces to impacted markets. Due to the risk of LLM output hallucinations, the connector is constrained to return a strict JSON and provide a confidence score. A low temperature setting is used to reduce the chance of creative hallucinations. The LLM's output can only output from a selected set of tickers composing the S&P

500 current and historical constituents. Operationally, once an LLM output is received, the pipeline filters out low-confidence mappings, limits the number of tickers used, and checkpoints progress for manual review. These controls are important because entity-linking errors have been shown to directly corrupt labels used in supervised learnings which have extremely detrimental effects on performance, as demonstrated by previous research.

5. ML Model and Experiment Design

We tested multiple model families and objectives to explore robustness under heavy-tailed, noisy labels. Loss functions include squared error (MSE), absolute error (MAE), Huber loss (robust), and quantile loss (median regression).

Rather than employ the standard train/test split of random selection, we employed a chronological split to avoid look-ahead bias. In other words, our testing set is all the data from before the cutoff (from April 1st, 2025 to April 29th, 2026) and our testing set is all the data after the cutoff (from April 29th, 2026 to May 1st, 2026). For clarification, even though a data sample in the testing set is after the cutoff, it still uses the financial metrics immediately before it for prediction. For instance, a testing point on April 30th, 2026 will still use historical financial metrics immediately before it to predict the 12-hour forward window.

This has the advantage of mimicking real-world conditions where we use all available data to make a future prediction and is helpful due to the limited size of the available data. There is the disadvantage of the testing possibly not fulfilling the same temporal conditions as the training set which would imply the model being less applicable, however, this does reflect how trading systems work in the industry.

Accuracy was measured using MAE, RMSE, and R^2. Directional accuracy is reported as the trading simulation rewards strongly depending on the sign of the move. To reflect portfolio deployment, a $1-per-trade simulation computed the P&L curve over the test window. The trading simulation was adjusted such that the notional/size of the trade is capped at $1 but can be less if the confidence of the prediction is lower.

5.1 Model Hyperparameters

Generally, the hyperparameters for each model follow the Python/sklearn standard default values. For clarity, they are described in the following table with a brief rationale:

Model	Hyperparameters	Rationale
Ridge	`alpha=1.0`	This is the default L2 shrinkage rate as it's considered a reasonable starting point. The actual values for some of the variables (such as the one-hot encoded tickers) can vary wildly without much contextual meaning making regularization needed.
Elastic Net	`alpha=0.001, l1_ratio=0.2, max_iter=6000`	This used a much smaller alpha than Ridge and a slightly smaller L1 component since we want to strike a balance between each regularization param.
Huber	`Default loss (MSE), max_iter=400`	We use the default loss since we are mainly concerned with the Huber regression model itself. The max_iter is raised a bit from the default of 100 since there were several sklearn warnings regarding convergence.
Random forest	`n_estimators=400, n_jobs=-1, Default depth and min_samples_leaf`	400 trees is a common standard for having enough complexity for variance reduction while still being tractable for the test set. Adding more trees could smooth prediction, but it likely wouldn't impact final performance too drastically. Depth and min_samples_leaf were left at default as there was no motivation to adjust them too much.
Gradient Boosting	`sklearn default n_estimators, learning_rate, max_depth`	The primary goal of including gradient boosting is to compare how the loss functions compare to each other. Tuning the parameters for each would likely see significant improvements in performance, but they are not the focus of the experiment.

Table 1- Model Hyperparameters

6. Experiment Results and Analysis

The total number of validated training samples that the models used to train was 24,092. The total testing size was 6,024. Despite the training sample tweets ranging over a much longer time period (~1 year) than the testing period (~4 days), the training set is proportionally much smaller than the testing set. This was intentionally implemented as explained earlier to make sure the training set would include samples that occurred on the same date range as the testing period, but a year ago.

Different combinations of models and loss functions were simulated to evaluate the effectiveness of each.

6.1 Model/Loss variation results

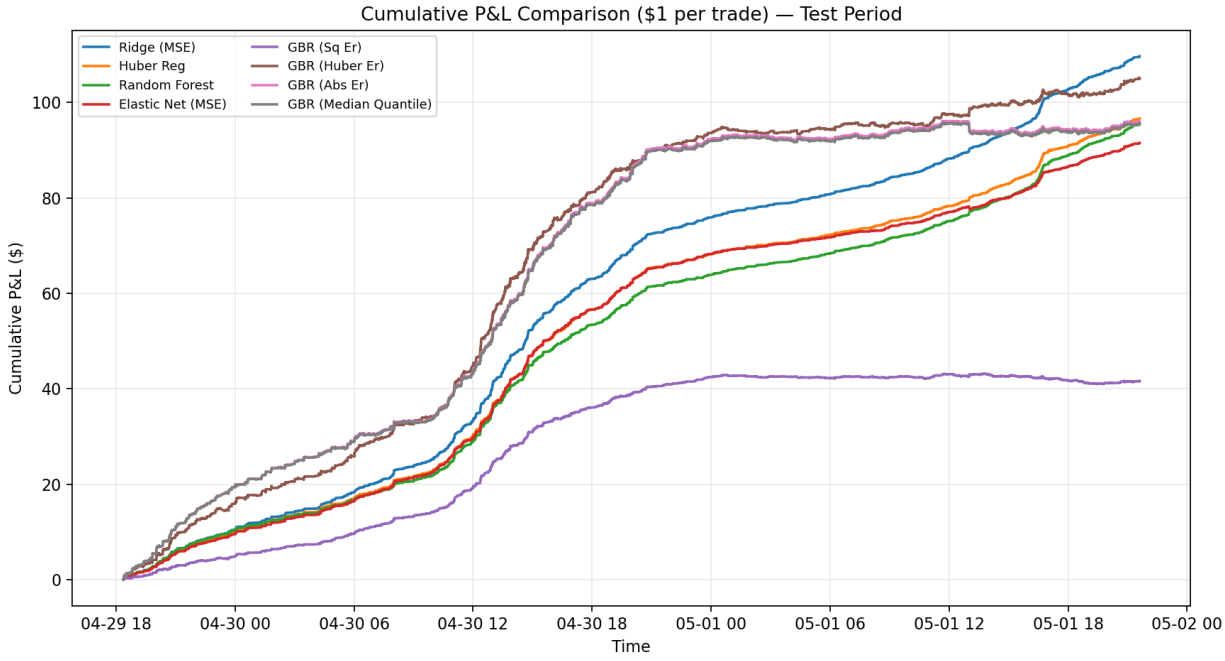
Model	MAE	RMSE	R^2	Directional Accuracy	Cumulative P&L	Number of Trades
Ridge (MSE)	0.604068	1.215221	0.987584	0.978752	109.690379	6024
Huber Reg	1.945144	3.907249	0.871648	0.861554	96.659922	6024
Random Forest	2.466855	7.628208	0.510778	0.902058	95.456306	6024
Elastic Net (MSE)	3.320995	7.139159	0.571496	0.826029	91.532769	6024
GBR (Sq Er)	4.628407	10.271002	0.113075	0.606076	41.572226	6024
GBR (Huber Er)	0.698954	1.441092	0.960192	0.940192	104.98912	6024
GBR (Abs Er)	2.055212	4.172302	0.809812	0.818033	95.908190	6024
GBR (Median Quantile)	2.466855	7.628208	0.510778	0.902058	95.445295	6024

Table 2- out-of-sample performance on the test period for each model/loss configuration, along with the final cumulative P&L from the $1-per-trade simulation.

As we can see, ridge regression using MSE consistently led in error metrics and in final P&L. The gradient boosting model using Huber loss also performed closely behind ridge across all metrics.

Gradient boosting using Huber loss was the strongest robust-linear candidate by MAE with gradient boosting using absolute error following close behind it.

It was surprising to see Ridge regression perform so well, especially considering that this training set turned out to be quite volatile. This is demonstrated by the fact that Gradient boosting using squared error performed by far the least well. This suggests that the training set contained outliers that heavily influenced or deviated from what the testing set exhibited. So, the high performance of the Ridge regression could either be explained by this testing set being particularly appropriate for Ridge, or by the Ridge capturing the more nuanced interactions of market trading.



Graph 1- P&L of each model/loss configuration over the testing period.

Here we can visually see the performance of each model/loss configuration across the short testing period. We can see that the trading returns are relatively stable until around April 30th noon where there was a surge of profits across all models. After this influx, some of the models taper off while others continue trading and seeing higher profits.

6.2 Nuanced interpretation of metric trade-offs

Error metrics (MAE/RMSE) evaluate magnitude accuracy, while trading simulation performance depends primarily on the sign and distribution of predicted returns. There is obviously a strong correlation between each, but a model with slightly worse RMSE can outperform in P&L if its

directional accuracy is better and does not produce large wrong-way bets. Conversely, a model could have excellent RMSE by predicting small magnitudes near zero but not generating large profitable trades due to missing the signs of high-impact moves or having low confidence. We can see this as even models with very different MAE and RMSE values can still come to very similar P&L values. For instance, when comparing GBR (Abs Er) and GBR (Median Quantile), Abs Er has better metrics for MAE, RMSE, and R^2, but because Median Quantile has better Directional Accuracy, it can make up for the difference and come to a very close final evaluation.

6.3 Mathematical intuition behind loss-function behavior

Huber loss transitions from quadratic to linear beyond a threshold δ. This robustness limits the influence of large residuals and can improve performance when the label y has outliers. Absolute error directly optimizes the median and can be stable, but it tends to underfit when the conditional mean contains an exploitable signal, it often doesn't lend itself to complexity needed to capture that signal. Quantile loss generalizes MAE to other quantiles and is useful for risk-aware predictions, but it doesn't align well for mean-return objectives. All these aspects can see different results and effects depending on the volatility and contents of the testing set, which will vary widely over time and over different social events.

6.4 Reliance by feature group (linear models only)

The table aggregates absolute coefficient masses for each feature family. Coefficients shown below are on a standardized feature scale, so they sum to 1.0 per model. Higher value implies the model relies more on that coefficient.

Model	Sentiment	Engagement	Market	LLM	Categorical	Other
ridge_mse	0.2100	0.0001	0.6086	0.0000	0.1813	0.0000
elasticnet_mse	0.1280	0.0015	0.4968	0.0001	0.3737	0.0000
huber_regression	0.1680	0.0000	0.5373	0.0000	0.2946	0.0000

Table 3- coefficients of each feature family. Aggregated values are the sum of the absolute values of their constituents.

We can see that all models primarily rely on market metrics for their predictions. This makes sense as they are the closest indicators for prediction, which is why quantitative trading strategies rely mostly on them. Interestingly, the categorical coefficients were also quite influential for prediction

while LLM confidence and other coefficients had effectively no impact. The categorical family contains parameters such as one-hot encoded ticker and username. The high coefficient suggests that certain one-hot encoded variables contain strong signals for prediction. Contextually, this would mean that knowing what the specific ticker is or where the content came from can provide varying performance.

Additionally, we can see a general trend in the reliance on the Sentiment, Market, and Categorical coefficients with the performance of the models. As the models decrease in performance based off Cumulative P&L, the reliance on market and sentiment decrease while the reliance on categorical increases. This would suggest that market and sentiment are better predictors of price change than categorical.

6.5 Sentiment coefficient weights (linear models only)

Coefficients shown below are on the same standardized feature scale from above, as such they do not sum to 1.0 as they are derived from the above table. A positive coefficient means higher values of that sentiment category correspond to an increase in the model's predicted delta_price, holding other features fixed. Magnitude indicates relative reliance within the linear model.

Model	Feature	Coefficient	Absolute Coefficient
elasticnet_mse	Uncertainty	0.127300	0.127300
elasticnet_mse	Negative	0.080400	0.080400
elasticnet_mse	Positive	-0.067000	0.067000
elasticnet_mse	vader_compound	-0.056012	0.056012
elasticnet_mse	Litigious	0.046900	0.046900
elasticnet_mse	Constraining	0.043550	0.043550
elasticnet_mse	vader_neg	0.037145	0.037145
elasticnet_mse	Complexity	0.032160	0.032160
huber_regression	Uncertainty	0.176700	0.176700

huber_regression	Negative	0.111600	0.111600
huber_regression	Positive	-0.093000	0.093000
huber_regression	vader_compound	-0.077748	0.077748
huber_regression	Litigious	0.065100	0.065100
huber_regression	Constraining	0.060450	0.060450
huber_regression	vader_neg	0.051559	0.051559
huber_regression	Complexity	0.044640	0.044640
ridge_mse	Uncertainty	0.224200	0.224200
ridge_mse	Negative	0.141600	0.141600
ridge_mse	Positive	-0.118000	0.118000
ridge_mse	vader_compound	-0.098648	0.098648
ridge_mse	Litigious	0.082600	0.082600
ridge_mse	Constraining	0.076700	0.076700
ridge_mse	vader_neg	0.065419	0.065419
ridge_mse	Complexity	0.056640	0.056640

Table 4- coefficients of each sentiment category. Categories are from the Loughran-McDonald dictionary unless they state vader in the front.

Across all model types, we see that positive categories have a negative coefficient while the negative categories have a positive coefficient. This is the same under VADER where vader_compound (which aggregates general positive sentiment) is negative and vader_neg is positive. This makes sense as a delta_price is defined as price during the event timestamp minus the price at end of the 12-hour forward window. So, a positive event that suggests that the price would increase would lead to a negative delta_price.

7. Discussion

7.1 Answering research questions

RQ1: Can short-form financial news tweets be transformed into structured features that predict short-horizon price movement?

We have demonstrated that textual content can be transformed into quantities that are useful for short-term prediction. The models still had mostly reliance on the financial metrics, which makes sense as the quantification of the textual content was likely to lose some of its signal. Market reactions also depend on much more than just social media announcements, which financial metrics are more likely to capture.

RQ2: Does explicit text to ticker mapping improve learnability compared to sentiment-only features?

The use of the LLM mapping step assisted in creating a more focused supervised learning approach. Rather than broadly connecting a textual piece to many markets or just overall market movements, we could specifically attach the piece to one market and analyze its effect more closely. This helped reduce noise from broader markets.

RQ3: Which loss/objective functions are most appropriate under noisy, heavy-tailed return labels?

Generalizing the results of the performance metrics, we can see that robustness is generally beneficial in improving prediction/P&L. However, performance is likely highly sample-specific as the best performing model was still Ridge regression using MSE which does not contain strong robustness. As such, Huber loss is likely still the best general loss function, but MSE does prove useful in specific conditions.

7.2 Fitting this approach in real workflows

In practice, this pipeline approach serves as a research prototype for a 'quantamental signal identifier.' In industry, this signal would likely not be directly traded systematically, but instead would trigger human review, prompt risk alerts, or serve as an input into a larger multi-signal portfolio model. Simply put, the predictions made in this project are likely too risky, volatile, and unstable to trade reliably by themselves. The core value of converting unstructured narratives into structured quantitative signals is still valuable beyond financial applications.

7.3 Limitations

Data volume was intentionally capped because of API costs and rate limits, and the X API in particular was expensive relative to the scope of this project. LLM-based ticker mapping can hallucinate or misclassify symbols; using low temperature and confidence filtering reduces that risk but does not remove it. Tweets are also inherently noisy and sometimes inaccurate, so poor outcomes may reflect both model limitations and weak or unreliable signal quality in social media rather than the modeling stack alone. The backtest is deliberately simplified: transaction costs, slippage, bid–ask spreads, and portfolio-level constraints are not modeled. Finally, the 12-hour return window assumes that no major confounding events occur in the same interval—such as earnings releases, macro announcements, or sector-wide shocks unrelated to the tweet—which is difficult to guarantee in practice. That is one reason robust error functions (for example Huber loss) are preferred, since they can limit the influence of outliers driven by such confounding moves.

7.4 Future extensions

Future work could expand historical depth and add more accounts, and an ensemble or voting scheme across sources could be used to confirm events before they are traded. Once API costs are controlled and mapping reliability improves, the pipeline could move from simulation toward paper trading or live trading. It would also be valuable to replicate common industry practice by adding sources such as earnings calls and Federal Reserve communications and comparing how the same models respond to each channel. On the modeling side, classification-first objectives (predicting direction rather than magnitude) and confidence-weighted position sizing are natural next steps to align optimization more closely with how P&L is actually earned.

7.5 Academic context of quantamental research

Many quantitative firms maintain 'quantamental' research groups that combine systematic models with fundamental signals. These approaches often use specialized proprietary feeds (earnings calls, macro releases) rather than mass social media sources. Public research remains limited because production-grade methods are typically proprietary. More open research in this area would benefit academia by clarifying which approaches generalize and how contextual reasoning can be incorporated without sacrificing rigor. This would apply beyond the context of financial markets and broadly toward the quantification of qualitative data, the structuring of unstructured information.

References

Bollen, Johan, et al. "Twitter mood predicts the stock market." *Journal of Computational Science*, vol. 2, no. 1, Mar. 2011, pp. 1–8, https://doi.org/10.1016/j.jocs.2010.12.007.

Ding, Xiao, et al. "International Joint Conference on Artificial Intelligence ." AAAI Press, *Deep Learning for Event-Driven Stock Prediction*, 2015, pp. 2327–2333.

Loughran, Tim, and Bill McDonald. "When is a liability not a liability? textual analysis, dictionaries, and 10-KS." *The Journal of Finance*, vol. 66, no. 1, 6 Jan. 2011, pp. 35–65, https://doi.org/10.1111/j.1540-6261.2010.01625.x.

Sprenger, Timm O., et al. "Tweets and Trades: the Information Content of Stock Microblogs." *European Financial Management*, vol. 20, no. 5, 29 May 2013, pp. 926–957, https://doi.org/10.1111/j.1468-036x.2013.12007.x.

Tetlock, Paul C. "Giving content to investor sentiment: The role of media in the stock market." *The Journal of Finance*, vol. 62, no. 3, 8 May 2007, pp. 1139–1168, https://doi.org/10.1111/j.1540-6261.2007.01232.x.