HEC Paris - MIF - Data Analysis in Finance

and

SEC Form 10-K Sentiment-Based Portfolio Construction

March 17, 2025

View Code on GitHub

Report

Nathan Bout, Constance Laverdure, Alexandre Brun

Teacher:

Augustin Landier

Contents

Intro

In recent years, the search for signals that can help predict future stock returns has become an extensive field of research, attracting the interest of academics, asset managers, and hedge funds alike. Traditional quantitative finance has long relied on structured data—such as financial ratios, earnings reports, and macroeconomic indicators to inform investment decisions. However, with the advent of advanced data-mining techniques, pioneered by quantitative analysts, market participants have increasingly turned to alternative data sources to gain a competitive edge. These techniques allow for the identification of hidden patterns and inefficiencies in the market that are not yet fully reflected in stock prices, thereby offering opportunities for excess returns.
A major breakthrough in this field has been the rise of large language models (LLMs), which have significantly expanded the scope of quantitative analysis by enabling the systematic processing of unstructured textual data. Unlike traditional numerical datasets, financial documents such as earnings call transcripts, news articles, analyst reports, and regulatory filings contain rich qualitative information that can influence market movements. By leveraging LLMs, researchers and practitioners can extract meaningful insights from these sources, particularly through sentiment analysis and textual similarity measures.

In this project, we explore the predictive potential of textual analysis in financial markets by applying LLM-based methods to corporate disclosures. Specifically, we focus on 10-K filings. Our study examines the sentiment and textual similarity of historical 10-K filings from firms in the Dow Jones Industrial Average (DJIA) and investigates whether these features can serve as reliable indicators of future stock performance and help develop profitable trading strategies

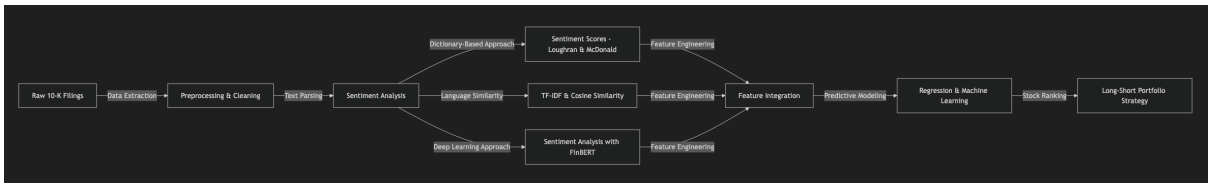

Figure 1: Project pipeline

1 Data Processing

1.1 Data retrieval

In the initial phase of our research, we focused on extracting Item 7 from the 10-K filings of companies listed in the Dow Jones Industrial Average (Dow 30) for each year. The code used to accomplish this task is documented in the Jupyter notebook titled `'filter_dow30_10K.ipynb'`. To facilitate this process, we sourced the list of Dow 30 constituents for each year from publicly available resources on the Internet.

Our approach involved generating a regular expression (regex) pattern tailored to identify and extract Item 7 from the 10-K filings. This method was chosen for its efficiency in parsing large volumes of textual data. However, despite our efforts to refine the regex pattern, we encountered a significant number of extraction failures. These failures can be attributed to several factors, including variations in the formatting of the 10-K filings across different companies and years, as well as the inherent limitations of regex in handling complex and non-standardized text structures.

To address these challenges, we opted to utilize an API to retrieve cleaner and more structured data. Specifically, we employed the SEC API, which provides a certain number of free requests, allowing us to access the necessary data without incurring costs. The code implementing this API-based data extraction is documented in the Jupyter notebook titled `'api_data_extraction.ipynb'`. This approach not only enhances the accuracy and reliability of our data extraction process but also streamlines the workflow by leveraging the standardized data formats provided by the SEC API. By doing so, we aim to minimize errors and ensure the successful extraction of Item 7 from the 10-K filings.

1.2 Signal generation

After retrieving and structuring the dataset, the next step involves transforming raw textual data into meaningful signals that can inform stock return predictions. This process integrates multiple Natural Language Processing (NLP) techniques to extract sentiment and linguistic patterns from corporate disclosures. We implemented the following transforms in `'signal_generation.ipynb'`.

1.2.1 Sentiment analysis using a dictionary-based approach

One of the core pillars of our signal-generation process is the sentiment analysis of the Item 7 section of 10-K filings. Research has consistently shown that the tone and sentiment embedded within these narrative sections can provide valuable information that goes beyond traditional financial metrics—such as earnings surprises, accruals, and operating cash flows.[Feldman et al. 2008]

By incorporating textual sentiment into our analysis, we are able to capture qualitative signals that help explain future stock performance. To do this, we rely on the specialized Loughran and McDonald dictionary—a lexicon engineered specifically for financial texts. This dictionary improves upon general

sentiment dictionaries by correctly categorizing words in a financial context, where terms might have different connotations compared to everyday language.

1.2.2 Language similarity analysis

Beyond simple sentiment metrics, analyzing the similarity of language between consecutive Item 7 filings offers another powerful approach to signal generation. Research presented in the paper "The Positive Similarity of Company Filings and the Cross-Section of Stock Returns" [Matúš 2020] indicates that low positive similarity between filings can predict outperformance. The underlying theory suggests that significant changes in language—particularly in the usage of positive terms—may signal important internal developments. In contrast, companies that employ highly similar positive language across filings might be experiencing stability or stagnation. To quantify these linguistic changes, we compute a similarity score between two successive filings. Our approach is implemented using a cosine similarity measure on TF–IDF representations of the texts.
In our signal generation framework, this similarity score is used as an additional feature alongside traditional sentiment metrics. The hypothesis is that a lower similarity (especially in terms reflecting positive sentiment) may capture subtle yet significant shifts in a company's narrative that can precede future outperformance. By integrating this feature into our predictive model, we aim to better account for the qualitative changes in corporate disclosures over time.

1.2.3 Sentiment analysis using BERT

While our dictionary-based approach provides a solid foundation for sentiment extraction, advanced NLP techniques can uncover deeper insights from financial texts. Recent developments in machine learning, particularly transformer-based models, have enabled more sophisticated approaches to text analysis, allowing for the capture of nuanced context and domain-specific language characteristics. One promising methodology involves using BERT (Bidirectional Encoder Representations from Transformers) models, which have shown remarkable performance in various NLP tasks. Specifically, we employ FinBERT, a BERT model fine-tuned for financial text analysis, to interpret the narratives within the Item 7 sections of 10-K filings. Research has demonstrated the potential of BERT-based models in outperforming dictionary-based methods; for example, studies analyzing Chinese MD&A sections using BERT showed superior accuracy and recall compared to traditional sentiment analysis approaches[Fedorova et al. 2022]. Unlike dictionary-based methods, BERT effectively captures contextual subtleties, making it ideal for analyzing complex financial disclosures where word meaning varies by context.
To operationalize this analysis, we implement a function that processes the text of an Item 7 filing using FinBERT. Because these documents can be lengthy, we first divide the text into manageable chunks of 512 tokens. For each chunk, the FinBERT pipeline produces a sentiment label (e.g., `positive`, `negative`, or `neutral`) along with an associated confidence score. We then map these labels into

numerical scores and weight each by its confidence score. The overall sentiment score for the document is computed as the average of these weighted scores, ensuring that each part of the text equally contributes to the final sentiment measure. This BERT-based approach is advantageous because it leverages the rich contextual embeddings of the model. Unlike simple bag-of-words or dictionary-based methods, FinBERT dynamically interprets the meaning of words based on their context, thereby allowing us to capture nuanced shifts in sentiment that may signal significant corporate developments. Consequently, integrating this method into our signal-generation framework broadens the scope of sentiment analysis in capturing both overt and subtle cues from financial disclosures.

1.3 Data Integration and Feature Engineering

The project combines textual sentiment signals with market performance data to construct a comprehensive dataset for predicting company returns. It leverages sentiment extracted from previous-year reports alongside key market indicators to explain the subsequent year's performance, using the predicted expected return as the basis for testing a trading strategy. All related code for this process can be found in `trading_strat_data.ipynb`, which merges the required sources, performs feature engineering, and saves the intermediate datasets (as CSV files) and generated plots in a folder it creates named `trading_strat_data`.

For each year and every company in the DJIA Index, the merged dataset is constructed as follows:

- **Sentiment Signals:**
  - We extract sentiment scores from the narrative sections of 10-K filings.
  - The signals include: **sentiment_score{Positive}**, **sentiment_score{Negative}**, **sentiment_score{Polarity}**, **sentiment_score{Subjectivity}**, **similarity_score**, and **nlp_result**.
  - These forward-looking indicators are sourced from the previous year's filings (e.g., sentiment values for 2011 are derived from the 2010 `10_K_info.txt` file).
  - In the codebase, these metrics appear at the end of each `10_K_info.txt` file, which are arranged by year and company inside the folder `api-data-signal`.
- **Market Regressor – DJIA Return:**
  - To capture overall market performance, the model incorporates the prior year's DJIA return as an additional regressor.
  - For any given year i, the DJIA return from year $i-1$ is used to help explain performance in year i.
  - We hypothesize that a company's statements in year i become more informative when combined with market return data from year $i-1$.

- **Company Sector:**
  - Each company is classified by its industry sector to capture industry-specific effects on returns and to enhance the explanatory power of the statements.
  - The sectors include:
    * Information Technology
    * Health Care
    * Consumer Discretionary
    * Financials
    * Industrials
    * Energy
    * Communication Services
    * Consumer Staples
    * Materials
- **Company Identifier:**
  - The company name is included as a regressor to capture company-specific linguistic patterns over time.
- **Target Variable – Company Returns:**
  - The target variable is the actual annual return of each company, computed from historical market data, which serves as the outcome for our regression model.

Summary: For each company in year i (i from 2011 to 2024), the merged dataset includes the following predictors:

- Sentiment metrics extracted from 10-K filings in year $i-1$,
- The DJIA return from year $i-1$,
- The company's sector classification,
- The company identifier (name) to capture firm-specific linguistic patterns.

The target variable is the actual market return for the company in year i. This design ensures that all predictors are available prior to the prediction period, making the dataset ideal for regression modeling and trading strategy analysis.

2 Modeling, Trading Strategy, and Backtesting

2.1 Overview

In this section, we introduce a predictive modeling framework and a long-short trading strategy to evaluate the effectiveness of our sentiment-based signals. An Ordinary Least Squares (OLS) regression model forecasts future stock returns, which then guide the formation of a long-short portfolio. We employ an expanding window approach from 2011 to 2024, retraining the model each year as new data becomes available, and placing particular emphasis on performance from 2021 onward.

All code for the regression, trading strategy, and results is in `trading_strat_modelling.ipynb`, which computes regression outputs, predicts returns (ranking them by expected and actual values), and saves all outputs in the `trading_strat_data` folder for each prediction year.

2.2 Predictive Model Specification

We fit an Ordinary Least Squares (OLS) regression model where each observation corresponds to a *company-year* pair. For year t and company j, the predictors (all taken from year $t-1$) include:

- **Sentiment Metrics:** Sentiment features extracted from the previous year's 10-K filings:
  - `sentiment_score_positive`,
  - `sentiment_score_negative`,
  - `sentiment_score_polarity`,
  - `sentiment_score_subjectivity`,
  - `similarity_score`,
  - `nlp_result`.
- **Market Indicator:** The DJIA return from year $t-1$, reflecting overall market trends.
- **Sector Information:** One-hot-encoded indicators for the company's industry sector.
- **Company Identifier:** One-hot-encoded indicators for each firm, allowing the model to learn company-specific effects.

The *target variable* is the realized annual return $R_{j,t}$ of company j in year t. Formally, the model can be written as:

$$
\begin{aligned}
R_{j,t} = \beta_0 & \\
& + \beta_1 \times \text{sentiment\_score\_positive}_{j,t-1} + \beta_2 \times \text{sentiment\_score\_negative}_{j,t-1} \\
& + \beta_3 \times \text{sentiment\_score\_polarity}_{j,t-1} + \beta_4 \times \text{sentiment\_score\_subjectivity}_{j,t-1} \\
& + \beta_5 \times \text{similarity\_score}_{j,t-1} + \beta_6 \times \text{nlp\_result}_{j,t-1} \\
& + \beta_{\text{DJIA}} \times \text{DJIA}_{t-1} + \sum_{m=1}^{M} \beta_{\text{sec},m} \text{Sector}_{m,j} \\
& + \sum_{n=1}^{N} \beta_{\text{comp},n} \text{Company}_{n,j} + \epsilon_{j,t},
\end{aligned}
$$

where:

- DJIA_{t-1} is the DJIA return in year $t-1$,
- $\text{Sector}_{m,j}$ is a one-hot indicator for the m-th sector to which company j belongs,
- $\text{Company}_{n,j}$ is a one-hot indicator for the n-th company, and
- $\epsilon_{j,t}$ is the error term.

This specification incorporates multiple sentiment features, market and sector dynamics, as well as firm-specific effects. All predictors reflect data available prior to the forecast period, aligning the model with real-world investment scenarios. Notably, we have deliberately limited our feature set to sentiment-based metrics, the DJIA return, and basic company identifiers (name and sector) to isolate the contribution of our NLP-driven sentiment analysis in predicting returns.

2.3 Training and Prediction

1. **Temporal Split:** We sort the data by year to ensure that the model only uses information from years up to $t-1$ when predicting returns for year t.
2. **One-Hot Encoding:** Categorical variables (e.g., *Company*, *Sector*) are converted to dummy variables so that the regression model can capture company- and sector-specific effects.
3. **Model Fitting:** We fit the OLS model on the training subset (e.g., 2011–2020) to estimate the coefficients.
4. **Out-of-Sample Prediction:** The fitted model is then used to predict returns for the next year (e.g., 2021). We compute performance metrics such as the mean squared error (MSE) on the test set, including the specific prediction year.

2.4 Long-Short Trading Strategy

Once the model generates predicted returns for a given year t, we implement a long-short portfolio:

- **Ranking:** Companies are ranked by their predicted returns from highest to lowest.
- **Portfolio Construction:**

  Long Positions Allocate half of the portfolio, equally weighted, to the top quartile of companies (those with the highest predicted returns).

  Short Positions Allocate the remaining half, equally weighted, to the bottom quartile of companies (those with the lowest predicted returns).

- **Strategy Return:** The strategy's annual return is calculated as the difference between the average realized return of the long positions and the average realized return of the short positions.

2.5 Rolling Backtest (Expanding Window)

TO SIMULATE A REAL-WORLD SCENARIO WHERE NEW DATA BECOMES AVAILABLE EACH YEAR, WE ADOPT AN EXPANDING WINDOW APPROACH:

1. **Initial Training (2011–2020):** We train on historical data from 2011 to 2020, then generate predictions for 2021.
2. **Roll Forward:** We expand the training set to include data through 2021, then predict for 2022. This process is repeated up to 2024.
3. **Performance Evaluation:** For each prediction year, we record the strategy's return and compare it to the DJIA return for the same year. We also compute the excess return of the strategy relative to the DJIA.

2.6 Results and Visualization

To finalize the analysis, we compute and visualize the performance of our long-short trading strategy against the DJIA over the forecast years 2021–2024. The results are presented as follows:

- **Annual Strategy Returns:** For each year, the strategy return is computed as the difference between the average return of the top-quartile (long) positions and that of the bottom-quartile (short) positions.
- **Benchmark Comparison:** For each simulation year, the strategy's performance is compared with the DJIA return of *that* year
- **Average Performance:** The overall performance across the test period is summarized by averaging the annual strategy returns and DJIA returns, as well as computing the average excess return (strategy return minus DJIA return).

Figure 2 illustrates the annual returns for both the trading strategy and the DJIA, with annotations for the exact percentage values.

Table 1: Long-Short Strategy vs. DJIA (2021–2024)

Year	Strategy Return	DJIA Return	Excess Return
2021	29.28%	20.23%	9.05%
2022	-40.18%	-9.40%	-30.78%
2023	61.29%	13.74%	47.55%
2024	23.03%	12.88%	10.15%
Average	18.36%	9.36%	8.99%

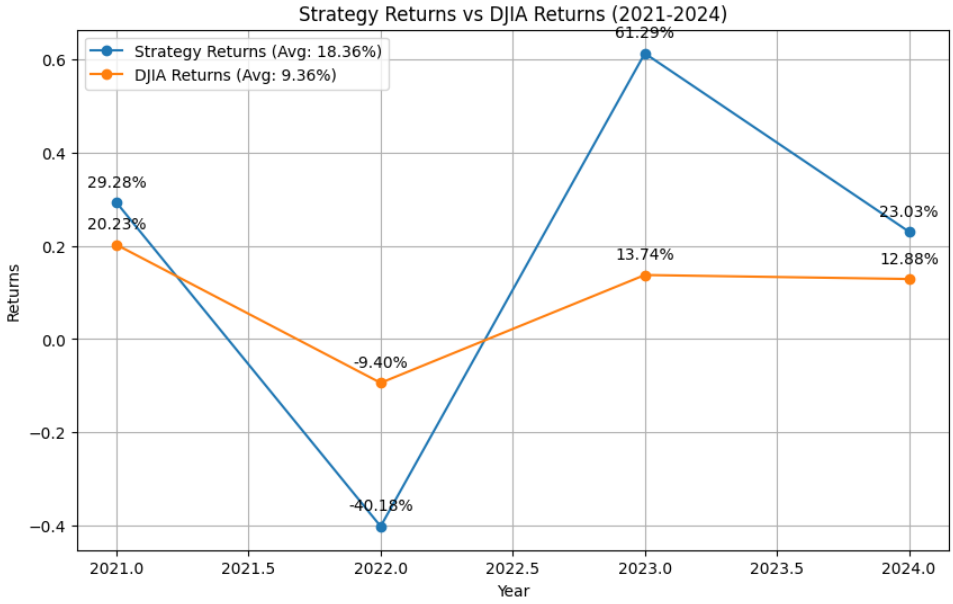

Figure 2: Annual Strategy Returns vs. DJIA Returns (2021–2024)

2.6.1 Interpretation

- **Market Phases (2021–2024):**
  - **2021:** Following the initial COVID-19 shock and extensive stimulus measures, markets maintained their upward momentum, resulting in stable strategy gains.
  - **2022:** Aggressive rate hikes aimed at controlling inflation caused a substantial drawdown, but were followed by
  - **2023:** A swift rebound as inflation pressures subsided, allowing the strategy to recover.
  - **2024:** Both the broader market and the strategy returned to more typical, steady conditions.

  Strategy Outperformance:
  - Over the entire 2021–2024 period, the long-short strategy maintained higher average returns than the DJIA by roughly 9%.

  - While performance dipped in 2022—mirroring broader market losses—the strategy continued to demonstrate an overall advantage due to its combination of long and short positions, coupled with sentiment-driven stock selection.

- **Data Limitations and Validation:**
  - We worked with approximately 420 data points (14 years × 30 companies), limiting the feasibility of more extensive validation (e.g., k-fold cross-validation).
  - Instead, we chose to balance data availability for model training with the need for testing (the multi-year backtest of the trading strategy).
  - With a larger dataset, walk-forward validation or similar time-sensitive methods could be used more effectively.

- **Overall Conclusion:**
  - A sentiment-oriented approach, augmented by basic market indicators and firm-level attributes, can effectively inform a long-short trading strategy.
  - Despite the constraints imposed by limited data, our results suggest that these sentiment-driven signals can outpace the market by a meaningful margin.

Conclusion

In brief, our framework integrates sentiment-based features from 10-K filings, prior-year market returns, and sector information to predict future company returns. An OLS model is trained using an expanding window (e.g., training on 2011–2020 to predict 2021, then 2011–2021 to predict 2022, etc.), and the predicted returns guide a long-short strategy that ranks stocks by expected performance. We backtest this strategy from 2021 to 2024, comparing its performance to the DJIA. Overall, our sentiment-driven approach provides a tangible advantage over the benchmark, delivering an average annual excess return of roughly 9%. Despite experiencing a brief drawdown in 2022, the strategy ultimately rebounded, highlighting the value of forward-looking sentiment information in capturing return dynamics that go beyond traditional market factors.

References

Fedorova, E. et al. (2022). “Impact of MDA sentiment on corporate investment in developing economies: Chinese evidence”. In: URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4884740.

Feldman, Ronen et al. (2008). “The Incremental Information Content of Tone and Sentiment in Management Discussion and Analysis”. In: URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1126962.

Matúš (2020). “The Positive Similarity of Company Filings and the Cross-Section of Stock Returns”. In: URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3690461.