Portfolio Optimization using Anomalies: A Deep Learning Approach

Aman Foujdar [*], Sandeep Juneja[*], , Atish Kumar [*†]
Nagpurnanand Prabhala [‡], Sidharth Wagle [*]

September 2025

Abstract

The explanatory power of Fama-French factors in capturing stock returns variation is well established in the literature. Furthermore, recent advances in deep learning techniques for time-series forecasting have outperformed regression-based models on a variety of datasets. In this paper, we construct factors related to Indian equity markets leveraging annual firm balance-sheet and daily stock level data. In addition, we propose a novel approach to portfolio optimization by training a deep learning model using Fama-French factors. We select the top 300 firms on the basis of market capitalization and build a long-short trading strategy where the portfolio is rebalanced every month. Our trading strategy generates an economically significant alpha and is easily implementable in practice.

1 Introduction

Estimating the equity risk premium is a topic of interest for academicians and investors alike. Over the years, various risk factors have been identified that have enhanced our understanding of variation in the equity returns. The most prominent set of risk factors has been proposed by Fama-French (Fama and French 1993) (Fama and French 2016). They identified Value, size, investment, and operating profitability to be the most significant features, which explain the stock price movements. (Carhart 1997) introduced momentum as another factor that is widely used by investors. Predicting stock returns is inherently a hard problem due to a low signal-to-noise ratio. Constructing portfolios negates the idiosyncratic changes of individual stocks, which helps in boosting the signal-to-noise ratio (Gu, Kelly, and Xiu 2020). Once the factors have been identified, the key question remains on how to use them in the most optimal manner.

[*] Safexpress Centre for Data Learning and Decision Sciences, Ashoka University, India
[†] Indraprastha Institute of Information Technology, Delhi, India atishk@iiitd.ac.in
[‡] Carey Business School, Johns Hopkins University, USA

Recent developments in the machine learning literature indicate that state-of-the-art deep learning models dominate traditional econometric techniques for time series data(Yong Liu et al. 2023).

In the context of the US and China, studies have shown that deep learning based methods have performed reasonably well in predicting stock returns (Kim, Korajczyk, and Neuhierl 2021) (Hanauer and Kalsbach 2023). A more recent study by (Cakici et al. 2023) extended the work of (Gu, Kelly, and Xiu 2020) (Bali et al. 2020), by testing the methodology for 46 countries, including India. Their dataset contained a yearly average of 1258 firms with 104 features. To maintain consistency across countries, the paper fails to capture country-specific nuances. The Indian equity market has a higher concentration of smaller firms. Hence, using the median as the bifurcation criterion to classify small and big firms is inappropriate (Agarwalla, Jacob, and Jayanth R Varma 2017). The average market capitalization has been around the 90th percentile for Indian firms. Additionally, studies that look at a large number of firms are often harder to replicate for investors.

In this study, we carefully construct factors keeping in mind the characteristics of the landscape of the Indian equities market, along with a trading strategy that is easier to implement. Furthermore, we explore a cutting-edge time series transformer model called iTransformer. Transformers are a class of deep learning models that excel at natural Language Processing (NLP) and vision tasks, but their use for time-series data was limited. Recently, researchers have proposed architectures meant for handling multivariate time-series data and shown solid performance for time-series data as well. The reason for selecting iTransformer has been two-fold: firstly, its performance has been better than its competitors on benchmark datasets by 16-38% (Yong Liu et al. 2023). Secondly, it preserves the architecture of the transformer, which makes it easier to understand as opposed to other models, which significantly alter the architecture. The goal of this paper is also to serve as a comparative benchmark for deep learning techniques in the field of empirical asset pricing.

We curate the top 300 companies listed on the Bombay Stock Exchange (BSE) each month on the basis of market capitalization for trading. Our strategy is simple: we train our models to create a long/short portfolio at the beginning of every month and close the positions at the end of the month. We are essentially trying to create a net-zero investment portfolio. This implies that the sum of portfolio weights should sum to zero, with long and short positions taking positive and negative values, respectively. Since the primary focus of this paper is to make the strategy investment-friendly, we use a loss function that minimizes the negative of the Sharpe ratio. This is critical to ensure that the technique is suitable for real-life applications. Currently, we rely on five Fama-French factors to train the model, namely Value, Size, Momentum, Investment, and Operating Profitability. We obtain an average monthly return of 1.25% with a Sharpe ratio of 2.1.

The remaining paper is structured in six sections. The second section of the paper is the review of the literature. Sections 3 and 4 talk about the data and methodology. Section 5 focuses on the model architecture and training. Section 6 contains the main results of the paper. Sections 7 and 8 are the conclusion and the appendix.

2 Related Literature

The roots of modern portfolio theory can be traced to Markowitz's mean-variance methodology. The key idea behind this framework was to treat assets as random variables and use the expected value and variance to estimate the risk and reward associated with them (Y. Zhang, Li, and Guo 2018). In their seminal work, (Fama and French 1993) identified five factors from the stock and bond markets. These factors, derived from the stock market, are the size and Book Equity/Market Equity (Value henceforth), and exhibited a strong relationship with returns. (Carhart 1997) added the Momentum factor and proposed a four-factor model. Fama-French further supplemented their work by adding two more factors, namely investment and profitability (Fama and French 2016). They used a dividend discount model to justify the addition of the two new factors. Their results showed the five-factor model performs better than the old three-factor model. In the Indian context, the five-factor model has been further extended by adding a human capital factor by (Maiti and Balakrishnan 2018).

Numerous studies have attempted to identify factors that can explain the price movement (Harvey, Yan Liu, and Zhu 2016). However, many of these factors are highly correlated and hence redundant for prediction purposes (Green, Hand, and X. F. Zhang 2017). (Freyberger, Neuhierl, and Weber 2020) showed that these anomalies can also have a non-linear relationship between them, which suggested that traditional empirical techniques are insufficient to capture these relationships. To counter this challenge, (Gu, Kelly, and Xiu 2020) (Karolyi and Van Nieuwerburgh 2020) introduce a machine learning method to overcome the shortcomings of traditional models. They showed that neural networks beat every other standard econometric technique and generate a significant alpha.

(Cakici et al. 2023) did a cross-country study by constructing many firm characteristics and applying machine learning techniques. This study included a total of 46 countries, and it found that although individual models generate significant gains but the combination of models turns out to be the best performing.

3 Data

The firm-level balance sheet and daily stock market variables are sourced primarily from the CMIE Prowess DX database. We extracted the daily stock market variable along with the annual balance sheets of publicly traded companies listed on the Bombay Stock Exchange (BSE). The reason for choosing the BSE over the National Stock Exchange (NSE) has to do with the fact that the BSE has a far higher number of listed companies. The time period for our analysis spans 14 years, from April 2011 to March 2024.

There are a total of 6900 unique firms in our entire dataset. We have restricted our attention to the top 300 firms by market capitalization. The rationale behind choosing the number lies in the fact that they capture close to 90 percent of the total market capitalization and are highly liquid for almost all the years in our dataset. fig. 1 shows the cumulative distribution of the yearly market capitalization of the firms listed on the BSE, and it is evident that the top 300 firms dominate the Indian equities market. The list of top 300 firms is updated every month.

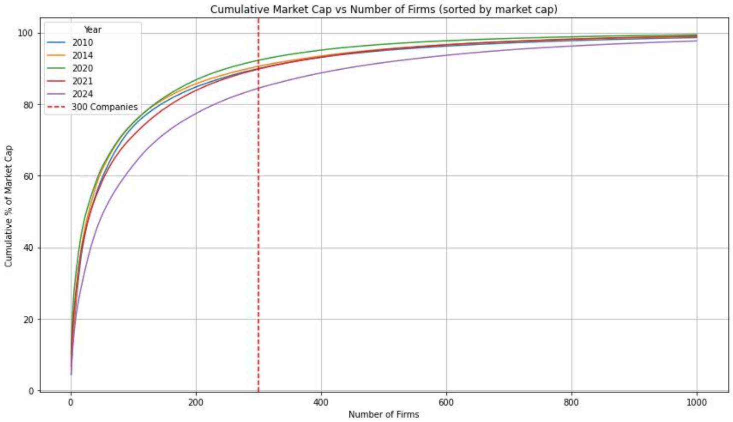


Figure 1: Cumulative Market Capitalization of firms Listed on the BSE

We follow the standard data cleaning procedures to prepare the input data.

- Liquidity: Liquidity in stocks is an important part of any trading strategy; we drop firms that are traded for less than 50 days in a financial year. All the firms in the top 300 stocks fulfil this criterion.
- Anomalous returns: Observations that exhibit anomalous daily returns from the stock market data have been removed. A threshold of upwards of ±40% has been applied, and those days have been removed. There are a total of 104 such instances in our dataset.
- Missing values: In case of missing values for variables available at daily frequency, we drop observations of only that specific date. Specifically,

any firm-year observation was excluded if either the Operating Profit or Investment variable was missing (`NaN`). In most instances, missing Operating Profit values were the result of at least one of its constituent fields being unavailable in the raw dataset.

4 Methodology

This section covers the algorithm behind creating factors using the firm's balance sheet and daily stock prices. We have followed the methodology proposed by Fama-French to calculate each of these factors. The following financial factors were computed on a monthly basis for each firm:

- **Monthly Return**: `bse_returns` measures the daily percentage movement of stocks using the adjusted closing price. To arrive at monthly returns of each stock, we take the geometrically compounded return over the course of the month.

- **Size**: The size factor considers the market capitalization of firms to bifurcate them into 'small' and 'big' categories. A common approach for dividing the firms could be using the median criteria; however, this is not suitable for the Indian market as mentioned earlier. Within the top 300 universe, the average market capitalization lies at the 80th percentile, which is shown in fig. 2; the top 20% firms capture close to 80% of the total market capitalization. Hence, the classification of 'small' and 'big' firms has been made using the above criteria.

- **Momentum**: Defined as the rolling 11-month geometrically compounded return, excluding the current month.

$$\text{Momentum}_t = \prod_{i=t-12}^{t-2} (1 + R_i) - 1 \tag{1}$$

  This formulation ensures the measure reflects only past performance and is available at the beginning of each month, avoiding forward-looking bias. The top 30 % (90 firms) are classified as 'winners' (W) and the bottom 30% (90 firms) are assigned the label as 'losers' (L)

- **Value**: Computed as the ratio of book value per share to the current share price. This ratio serves as a measure of relative valuation, where a higher value indicates that the stock is undervalued relative to its fundamentals.

$$\text{Value} = \frac{\text{Book Value of Equity}}{\text{Market Capitalization}} \tag{2}$$

  Here, the top 30% (90 firms) are called the value (V) stocks, the bottom 30% (90 firms) are the Growth (G) stocks, and the remaining are classified as neutral (N).

- **Operating profitability**: It is a measure to capture the firm's core earnings power relative to its equity base. The formula for operational profit involves subtracting Cost of Goods Sold (CoGS), Selling, General and Administrative (SG&A), and interest expense from the sales revenue and dividing it by book equity of the previous year. This ratio indicates operating performance by excluding financing and non-operating items and focuses on the firm's core business operations. Since the variables required to compute the factor are available only in the balance sheet, this factor is updated annually. The top 30% (90 firms) of the firms are classified as Robust (R) and the bottom 30% (90 firms) are Weak (WE):

$$\text{Operational Profit} = \frac{\text{Sales} - \text{CoGS} - \text{SG\&A} - \text{Interest Expense}}{\text{Book Equity}_{t-1}} \tag{3}$$

- **Investment**: This factor tracks the annual growth in total assets, representing the year-on-year change. Empirically, it has been observed that firms that require higher levels of capital expenditure are less lucrative for investors. The investment factor in asset pricing models is instrumental in capturing investors' expectations about the future earnings prospects of the business. The top 30% (90 firms) firms are considered as aggressive, and the bottom 30% (90 firms) are It is calculated using the following formula:

$$\text{Investment} = \frac{\text{Total Assets}_{t-2} - \text{Total Assets}_{t-1}}{\text{Total Assets}_{t-2}} \tag{4}$$

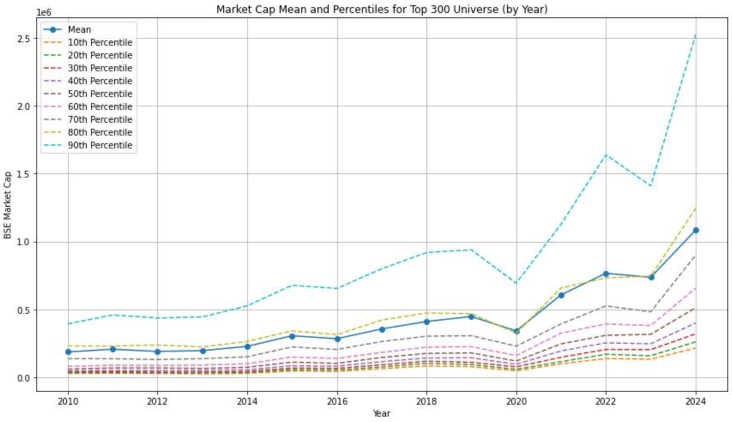


Figure 2: Average and Percentiles of Top 300 Market Capitalization

An important point to highlight here is that to avoid any forward-looking bias, we take a six-month lag to construct variables that require balance sheet variables. For instance, if the financial year ends on 31st March, then we update the factors on 30th September. Once we create these variables, our broader objective is to rebalance our portfolio every month. Since it is a net-zero investment portfolio, the weights of all the stocks should sum to 0. We train the machine learning model to "recommend" which stocks to go long/short on, along with their respective allocations. The model learn the relationship between features and next month's return to arrive at decisions. We will discuss the model architecture in the next section in detail.

5 Model

Transformers are deep learning based models proposed by (Vaswani et al. 2017) that have played a key role in the development of Large Language Models. These models can capture long-term dependencies in data where traditional architectures like Long Short-Term Memory(LSTM) and Convolution Neural Network (CNN) fell short. The most critical part of any transformer model is the self-attention mechanism. Its primary role is to make the model focus on the relevant information and ignore the irrelevant ones.

Despite their tremendous performance on NLP and vision tasks, simple linear forecasting techniques outperformed transformers on time-series data, raising questions about their applicability in this domain (Yong Liu et al. 2023). It has been argued that the vanilla transformer architecture is not suitable for multivariate time-series data. Consequently, various attempts have been made to modify the vanilla transformer architecture to align with time-series data, such as (Kitaev, Kaiser, and Levskaya 2021), (Nie et al. 2023), (Wu et al. 2022).

In the context of our research problem, the traditional approach of constructing an input embedding causes self-attention to focus on the temporal relationships of stocks, ignoring the multivariate correlations. To counter this, (Yong Liu et al. 2023) proposed a novel model called iTransformer; the central idea of this model is to invert the input embedding, which alters the dimensions on which the attention is applied. The major implication of this step is converting the self-attention from a temporal to a variate token view. In other words, each element in the self-attention matrix represents asset-feature scores instead of asset-time. An important point to note is that this approach preserves the architecture of the transformer, unlike other time-series transformer models; the innovation lies only in embedding inversion. Attributing to the structure of iTransformer, the model makes decisions by looking at cross-sectional relationships between assets and features.

5.1 Model Architecture

A training sample is a matrix of N assets and F features over a specified sequence length, T. The asset-feature combination is flattened to obtain a matrix with dimension VXT. For simplicity, one can imagine the entire training data to be matrices with months moving in a sliding window approach. Each batch consists of matrices of the same dimension.

- Training the iTransformer model involves encoding the input features, which is done by projecting each sample to a higher-dimensional space d; this is the step where the "inversion" takes place. $H \in \mathbb{R}^{B \times V \times d}$ for a batch of size B.
- The attention matrix is split into h heads with width $d_k = d/h$. This is known as the Multi-head attention framework. The idea behind this approach is to learn different aspects of the data through each head.
- The embedding is multiplied by the weight matrix W_i^Q, W_i^K , W_i^V to yield the query (Q_i), Key (K_i), and Value (V_i) vectors respectively.

$$Q_i = HW_i^Q, \qquad K_i = HW_i^K, \qquad V_i = HW_i^V,$$
$$W_i^Q,\, W_i^K,\, W_i^V \in \mathbb{R}^{d \times d_k}.$$

$$\text{Attn}(Q_i, K_i, V_i) = \text{softmax}\left(\frac{Q_i K_i^\top}{\sqrt{d_k}}\right) V_i. \tag{5}$$

Concatenating all the individual heads would help recover the original dimension.

$$\text{head}_i = \text{Attn}\big(QW_i^Q,\, KW_i^K,\, VW_i^V\big), \quad i = 1, \ldots, h, \tag{6}$$
$$\text{MHSA}(Q, K, V) = \text{Concat}\big(\text{head}_1, \ldots, \text{head}_h\big)\, W^O,\ W^O \in \mathbb{R}^{(hd_k) \times d}. \tag{7}$$

LayerNorm is applied on the MHSA output at the token or feature map level across d features to normalize the inputs for stabilizing the training process. This is a standard practice in Transformers to make the learning efficient.

$$H^{(1)} = LayerNorm(H + SelfAttention(MHSA(H))) \tag{8}$$

Post normalization, the output is passed through a Feed-Forward Neural Network (FFNN) with GELU activation function, and the same Layernorm process is repeated again.

$$H^{(2)} = LayerNorm(H^{(1)} + FFNN(H^{(1)}))) \tag{9}$$

The final matrix is then projected to a scalar, producing scores for each at the asset-feature level. These scores are used to map portfolio allocation in each stock. The weights are normalized to make the respective sum of long and short weights to be 1, and the sum of all the weights

$$\sum_{i=1}^{N} w_{b,i} = 0$$

$$\sum_{i=1}^{N} |w_{b,i}| = 1$$

In every epoch, training samples are fed sequentially in the form of matrices discussed above, and the model learns weights corresponding to Query, Key, and Value. The model maps the feature space to the next period's returns by learning network parameters to allocate portfolio weights to each stock. We can calculate the expected weighted return, r_b, by simply multiplying each stock i's weight w_i by its return y_i. We follow the standard definition of the Sharpe ratio, which involves dividing the mean of returns μ by standard deviation σ, to construct a loss function. ϵ is added in the denominator to avoid a zero denominator. The loss function gets updated for each batch, B.

$$r_b = \sum_{i} w_{b,i}\, y_{b,i}.$$

$$\mu = \frac{1}{B} \sum_{b \in B} r_b, \qquad \sigma = \sqrt{\frac{1}{B} \sum_{b \in B} (r_b - \mu)^2}\,.$$

$$\text{Sharpe} = \frac{\mu}{\sigma + \varepsilon}, \qquad \mathcal{L}_{\text{Sharpe}} = -\,\text{Sharpe}.$$

5.2 Training

We leverage these factors to train the iTransformer model; because factor construction requires a two-year lookback, the input data starts from 2013. The model's features are the five factors and their composite score, and the target is the monthly return. Each feature recieves a signal $s_i \in \{-1, 0, +1\}$, where $+1$ denotes the long side, -1 the short side, and 0 neutral. *Small*, *Winner*, *Conservative*, *Robust* and *High* receive the value $+1$. Conversely, *Big*, *Loser*, *Aggressive*, *Weak*, and *Low* get the value of -1 and 0 otherwise. To simplify the investing process, each stock receives an equal portfolio weight. The model is trained for 30 epochs, since the time period of the input data is short and has limited features, the model learns patterns fairly quickly. fig. 3 shows the actual and smoothened training loss of the model. The training time period chosen for the training data was from January 2013 to December 2021. The testing period was two years, starting from January 2023 to December 2024.

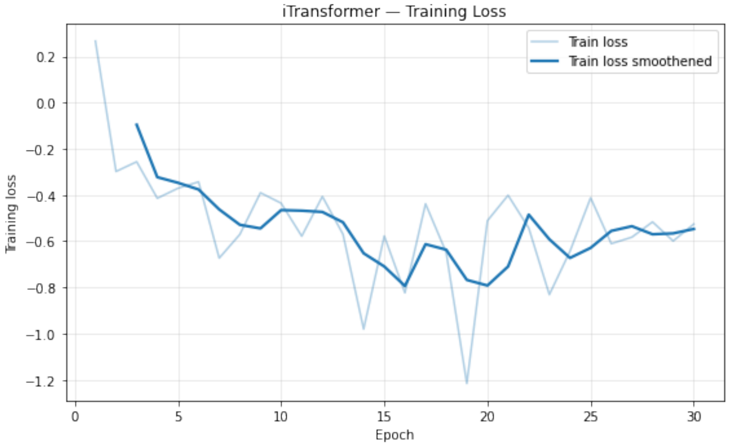


Figure 3: Training Loss

6 Results

In this section, we discuss the main results from the paper. The Indian equity markets have given decent results to investors. The NIFTY50 index, which comprises the top 50 companies of India, grew by 11% CAGR in the past 15 years. During the same period, small and mid-cap firms witnessed a sharp rise in their stock prices. fig. 4 shows the buy-and-hold returns of equal-weighted portfolios for all the listed firms in the BSE since 2013. The stocks belonging to the “small” in size and “winner” in the momentum factor surpass every portfolio. This portfolio has provided a 38% CAGR over 11 years; substantially higher than the NIFTY50 index. This difference shows the potential of combining Fama-French factors for trading. Firms belonging to the Small-Robust portfolio and the Small-Conservative also provide strong returns with a CAGR of 31% and 28%, respectively. Empirical evidence suggests that equal-weighted portfolios tend to do better than value-weighted ones, as equal-weighted portfolios assign higher weights to smaller firms (Gu, Kelly, and Xiu 2020). Despite the high returns, it is often not a feasible strategy to execute due to the sheer number of stocks and also due to the illiquid nature of many of these stocks.

Historically, momentum is considered to be the best performing factor in the Indian markets (Agarwalla, Jacob, and Jayanth Rama Varma 2014), which

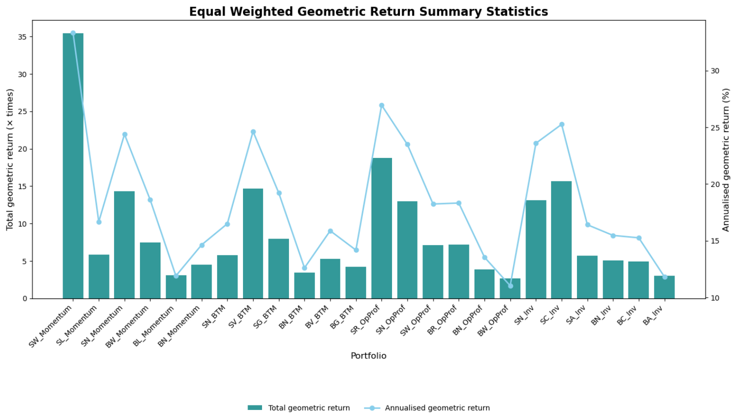
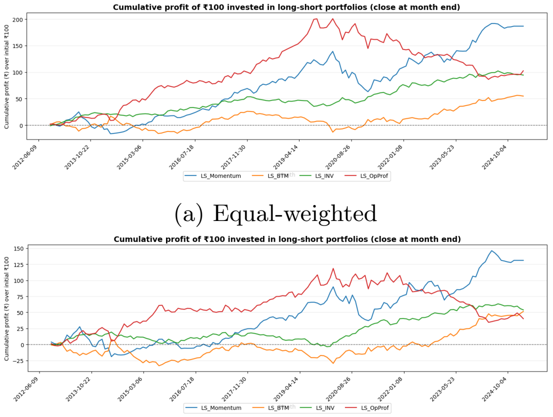


Figure 4: Equal weighted Buy-and-hold Returns for all firms

can be seen in fig. 5. Interestingly, the momentum factor has given substantial returns to investors post the COVID-19 pandemic. We also observe a trend reversal with respect to the operational profitability factor; prior to 2020, it was the best-performing factor, but it has underperformed since then. Figure 2 shows the equal and value-weighted returns of various portfolios. As mentioned earlier, these long/short portfolios have been created using the 80-20 split. These returns show how each of the factors performs individually. Concretely, at the beginning of each, a long/short portfolio is created with an initial investment of Rs. 100 at the start of every month, and positions are closed at the end of the month. This process is repeated every month.



(a) Equal-weighted

(b) Value-weighted

Figure 5: Portfolio performance by weighting scheme.

The model produces an annualized return of 16% for the test period with a Sharpe ratio of 2.1. The maximum loss in any given month in the test period is 1.66%. fig. 6 shows the cumulative returns over the two-year test period. Despite the smaller data size, we believe these results serve as preliminary evidence of the capability of transformer-based architectures.

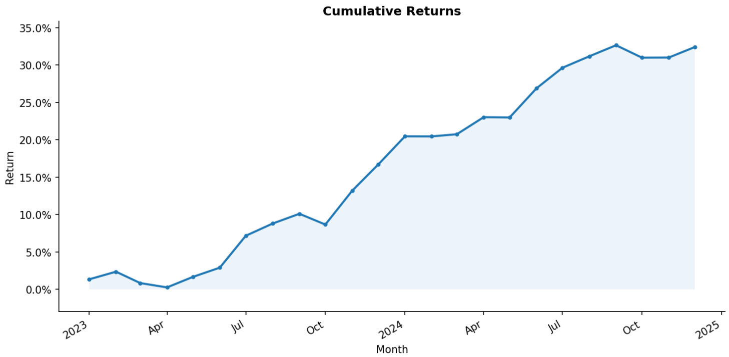


Figure 6: Cumulative Returns for Top 300 stocks: Model Output

7 Conclusion

This paper deals with using time-series transformer-based models for monthly rebalancing of portfolios using Fama-French factors. The preliminary evidence shows that the model is capable of generating excess returns for investors even when restricting the stock universe to 300. We plan to extend the data starting from 2000, and we also incorporate more anomalies that are established in the literature. There are other deep learning architectures that we intend to explore in further iterations of this paper.

References

Agarwalla, Sobhesh Kumar, Joshy Jacob, and Jayanth R Varma (2017). “Size, value, and momentum in Indian equities”. In: *Vikalpa* 42.4, pp. 211–219.

Agarwalla, Sobhesh Kumar, Joshy Jacob, and Jayanth Rama Varma (2014). “Four factor model in Indian equities market”. In: *Indian Institute of Management, Ahmedabad Working Paper* 2013-09, p. 05.

Bali, Turan G et al. (2020). “Predicting corporate bond returns: Merton meets machine learning”. In: *Georgetown McDonough School of Business Research Paper* 3686164, pp. 20–110.

Cakici, Nusret et al. (2023). “Machine learning goes global: Cross-sectional return predictability in international stock markets”. In: *Journal of Economic Dynamics and Control* 155, p. 104725.

Carhart, Mark M (1997). “On persistence in mutual fund performance”. In: *The Journal of finance* 52.1, pp. 57–82.

Fama, Eugene F and Kenneth R French (1993). “Common risk factors in the returns on stocks and bonds”. In: *Journal of financial economics* 33.1, pp. 3–56.

— (2016). “Dissecting anomalies with a five-factor model”. In: *The Review of Financial Studies* 29.1, pp. 69–103.

Freyberger, Joachim, Andreas Neuhierl, and Michael Weber (2020). “Dissecting characteristics nonparametrically”. In: *The Review of Financial Studies* 33.5, pp. 2326–2377.

Green, Jeremiah, John RM Hand, and X Frank Zhang (2017). “The characteristics that provide independent information about average US monthly stock returns”. In: *The Review of Financial Studies* 30.12, pp. 4389–4436.

Gu, Shihao, Bryan Kelly, and Dacheng Xiu (2020). “Empirical asset pricing via machine learning”. In: *The Review of Financial Studies* 33.5, pp. 2223–2273.

Hanauer, Matthias X and Tobias Kalsbach (2023). “Machine learning and the cross-section of emerging market stock returns”. In: *Emerging Markets Review* 55, p. 101022.

Harvey, Campbell R, Yan Liu, and Heqing Zhu (2016). “… and the cross-section of expected returns”. In: *The Review of Financial Studies* 29.1, pp. 5–68.

Karolyi, G Andrew and Stijn Van Nieuwerburgh (2020). “New methods for the cross-section of returns”. In: *The Review of Financial Studies* 33.5, pp. 1879–1890.

Kim, Soohun, Robert A Korajczyk, and Andreas Neuhierl (2021). “Arbitrage portfolios”. In: *The Review of Financial Studies* 34.6, pp. 2813–2856.

Kitaev, Nikita, Lukasz Kaiser, and Anselm Levskaya (2021). “Reformer: The Efficient Transformer”. In: *International Conference on Learning Representations*.

Liu, Yong et al. (2023). “itransformer: Inverted transformers are effective for time series forecasting”. In: *arXiv preprint arXiv:2310.06625*.

Maiti, Moinak and A Balakrishnan (2018). “Is human capital the sixth factor?” In: *Journal of Economic Studies* 45.4, pp. 710–737.

Nie, Yuqi et al. (2023). “A Time Series is Worth 64 Words: Long-term Forecasting with Transformers”. In: *The Eleventh International Conference on Learning Representations*.

Vaswani, Ashish et al. (2017). “Attention is all you need”. In: *Advances in neural information processing systems* 30.

Wu, Haixu et al. (2022). “TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis”. In: *The Eleventh International Conference on Learning Representations*.

Zhang, Yuanyuan, Xiang Li, and Sini Guo (2018). “Portfolio selection problems with Markowitz’s mean–variance framework: a review of literature”. In: *Fuzzy Optimization and Decision Making* 17.2, pp. 125–158.

8 Appendix

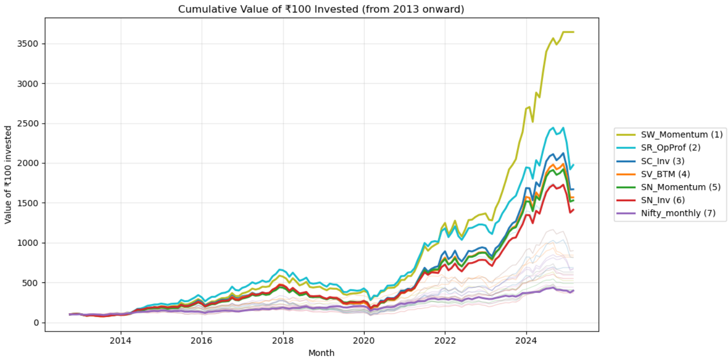


Figure 7: Buy-and-hold Returns: All Stocks

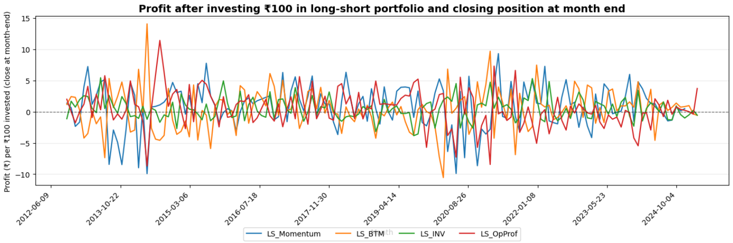


Figure 8: Equal-weighted long/short portfolio: Monthly Returns

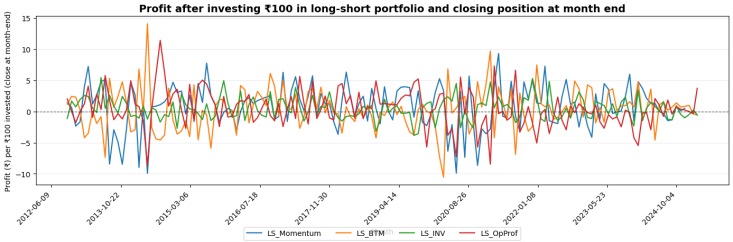


Figure 9: Value-weighted long/short portfolio: Monthly Returns