NLP-Based Analysis of Annual Reports: Asset Volatility Prediction and Portfolio Strategy Application

Xiao Li[1], Yang Xu[2], Linyi Yang[4], Yue Zhang[3,4] and Ruihai Dong[1,*]

[1] *Insight Centre for Data Analytics, School of Computer Science, University College Dublin, Ireland*
[2] *School of Economics and Management, Beihang University, Beijing, China*
[3] *Zhejiang University, Zhejiang, China*
[4] *School of Engineering, Westlake University, Zhejiang, China*

Abstract

Leveraging recent developments in natural language processing (NLP), we constructed a prediction model using corporate financial annual reports to forecast the stock volatility indicator Beta (β), by analyzing risk discussions. The predicted Beta values were used to construct investment portfolios, whose market performance was then evaluated. Our research demonstrates that the Hierarchical Transformer-based model effectively captures complex risk information from annual reports, leading to improved returns in portfolio simulations. Our motivation arises from the need to better understand and process long, unstructured financial texts like annual reports, which contain crucial yet nuanced risk factors. By utilizing the hierarchical model, we aim to overcome traditional models' limitations in handling such long documents, thereby improving the model's understanding of both sentence-level and document-level contexts. The results highlight the potential of deep learning, particularly hierarchical models, in financial text prediction, and provide a novel perspective on asset management strategies. Compared to the S&P 500 benchmark, portfolios constructed using the predicted Beta values from our model achieved an average return increase of 21% over the same period.

Keywords

Natural Language Processing, Financial Forecasting, Asset Volatility Prediction, Deep Learning, Transformer Models, Risk Assessment, Investment Portfolio

1. Introduction

Financial statements are a key source of information for both internal and external stakeholders, playing a crucial role in market decision-making. It is always analyzed by the investors to assess a company's financial health and market potential. Most papers focus on examining the flow of quantitative information, such as accounting and financial data [1]. Despite the importance of taking advantage of descriptive financial documents, the difficulty in accurately quantifying descriptive information is one of the reasons for the scarcity of studies into how investors understand it.

Thanks to recent breakthroughs in NLP, a rising corpus of literature now employs content analysis to quantify the sentiment and content of descriptive information and learn how the market interprets it. However, the inherent uncertainty of financial markets makes robust predictions challenging. Although NLP models have shown promising results in various financial applications, such as sentiment analysis [2], risk assessment [3, 4], and market forecasting [5], there is still a need for more precise tools in specific financial forecasting tasks. This is particularly true when attempting to extract complex risk indicators from long, detailed financial texts such as annual reports.

Beta (β) is a fundamental measure of financial risk, capturing the volatility of an asset relative to the broader market. It plays a central role in the Capital Asset Pricing Model (CAPM), helping investors assess the sensitivity of individual stocks to market fluctuations, which is crucial for making informed portfolio decisions [6, 7]. Unlike other financial metrics, Beta offers a direct, interpretable link between

AICS'24: 32nd Irish Conference on Artificial Intelligence and Cognitive Science, December 09–10, 2024, Dublin, Ireland

[*] Corresponding author.

xiao.li@ucdconnect.ie (X. Li); yang_xu@buaa.edu.cn (Y. Xu); yanglinyi@westlake.edu.cn (L. Yang); zhangyue@westlake.edu.cn (Y. Zhang); ruihai.dong@ucd.ie (R. Dong)

0009-0005-6418-2354 (X. Li); 0000-0001-9683-1498 (Y. Xu); 0000-0003-0667-7349 (L. Yang); 0000-0002-5214-2268 (Y. Zhang); 0000-0002-2509-1370 (R. Dong)

market behaviour and risk, making it a valuable tool in managing portfolio risk and return strategies [6, 8, 9]. By predicting Beta through textual analysis of financial documents like annual reports, we open new possibilities for aligning qualitative insights with quantitative risk assessments. NLP techniques allow us to extract nuanced risk information embedded in long, complex texts, enhancing traditional financial models that rely on numerical data alone. This approach bridges the gap between descriptive financial disclosures and quantitative metrics, enabling a more sophisticated and data-rich method for predicting asset volatility and improving risk management strategies.

The 10-K annual reports filed by public companies provide detailed financial data and risk factors. In this study, we focus on the "Item 1A: Risk Factors" section, aiming to predict Beta by analyzing the textual content related to company risks. This approach provides a bridge between qualitative risk discussions in financial documents and quantitative financial outcomes. By utilizing these predicted Beta values, we simulate portfolio performance to evaluate the practical applications of this method in real-market conditions.

Our results demonstrate the practicality of combining deep learning techniques, such as Hierarchical Transformer-based models, with traditional financial analysis. The Hierarchical Transformer-based model is particularly suited for processing long and complex texts like financial reports, allowing us to capture risk information more effectively. This method not only improves the accuracy of Beta prediction but also provides new perspectives on asset management and investment strategies by offering a more detailed assessment of financial risk. We further apply the predicted Beta values to construct investment portfolios and simulate their performance in real market conditions. Our results show that portfolios built with the predicted Beta values achieve higher returns and better risk management compared to traditional methods, highlighting the effectiveness of NLP-based financial analysis in practical investment scenarios.

2. Related Work

2.1. Risk Assessment

Financial risks refer to risks related to finances, such as market risks, credit risks, and operational risks [10]. Among them, the stock investment risk within the market risk, that is, the volatility of stock returns within a certain period, has attracted extensive attention in financial market research [11, 12, 13]. These studies indicate that financial disclosures, such as 10-K annual report [3] and earnings call materials [14], are valuable data sources for financial risk assessments. Given the vast range of financial consequences and arbitrage opportunities that come with stock volatility, accurate projections may contribute to a better understanding of financial markets and higher returns on investment [15]. Furthermore, financial disclosure research can aid in the discovery of each company's possible operational issues, reducing information asymmetry in the investment market to some level [16, 17, 18].

In financial market risk research, researchers generally believe that the stock price is unpredictable [19, 20]. That is because due to numerous influencing factors in the real market, as a result, It is impossible to accurately break down and quantify every risk factor. For example, macroeconomic indicators, market sentiment, company financial health, political events, etc. may have an impact on stock prices [21]. In addition, factors such as investor behaviour, market microstructure, and the international economic environment also have an important impact on stock prices [22]. So it is challenging to analyse stock prices accurately [23]. In the vast majority of cases, stock prices behave as random walks [20]. This unpredictability stems from the market's complexity and dynamics, and even the most advanced models struggle to capture all relevant variables and nonlinear relationships [24].

However, Bernard and Thomas [25] and Sadka [26] found a relationship between stock price volatility and the time of significant events rather than directly predicting stock prices. Therefore, many recent studies are based on this to do volatility analysis and forecasting. This suggests that while stock prices may be random in general, specific events may still have a significant impact on them. For example, events such as company earnings releases, management changes, and major policy adjustments often lead to abnormal stock price fluctuations [21]. These major events often cause market participants to reassess the company's future prospects, triggering dramatic stock price fluctuations[27]. Therefore, many recent studies have used this as a basis for volatility analysis and prediction.

For example, Theil et al. [28] proposed multiple deep learning models to extract text information from 10-K documents and predict stock volatility. By using NLP techniques, these models are able to capture subtle sentiment changes and risk warnings in documents, thereby improving prediction accuracy [29]. In addition, Qin and Yang [14] and Yang et al. [30] discussed extending earnings call analysis to multimodal prediction problems by incorporating text and audio information into the same model. This multimodal analysis approach can more comprehensively reflect management's attitude and market sentiment by combining speech features and language features, thereby providing more accurate predictions.

In addition, some researchers have also focused on other data sources, such as social media and news data, to improve stock price prediction. For example, Bollen et al. [31] found that Twitter sentiment can be used as a proxy variable for market sentiment and has predictive power for short-term market fluctuations. Similarly, Tetlock [2] studied the impact of news media content on the stock market and found that negative news reports are often associated with stock price declines. Therefore, combining multiple data sources and advanced analysis techniques can improve the predictive power of stock price fluctuations to a certain extent.

2.2. Beta Prediction

In financial markets, Beta (β) is a measure of the volatility of an asset or portfolio compared to the market as a whole (usually the S&P 500 index). Traditional methods of calculating Beta are mainly through the Capital Asset Pricing Model (CAPM), proposed by Sharpe [6] and Lintner [8]. They believe that there is a certain linear relationship between asset returns and market returns. Then, subsequent studies found that it is difficult to make actual predictions using a linear regression model such as CAPM [6, 32]. They found that the model relied too much on historical data training, resulting in a significant decrease in its predictive ability when extreme markets occurred.

In order to improve the accuracy of Beta value prediction, Fama and French [9] proposed a three-factor model and Carhart [33] proposed a four-factor model in subsequent studies. Compared with the CAPM model, which only considers company and market returns, the multi-factor model incorporates more market macro and micro variables, such as company size, book-to-market ratio, and momentum. This enhances the models' ability to explain the sources of market risk and demonstrates that these multi-factor models are more accurate in predicting market risk than the CAPM model in practical applications.

With the progress of technology, there has been an increase in research exploring the use of machine learning models for predicting Beta values. The advantage of machine learning is that it can handle large-scale data and can quantify complex data. For example, in early studies, Kim [34] tried to use support vector machines (SVM) and Huang et al. [35], Niu et al. [36] tried to use artificial neural networks (ANN). Their experiments aimed to learn and predict Beta values from stock market dynamics. Based on these research findings, it is evident that machine learning proves to be well-suited for forecasting, within markets due, to its ability to deliver predictions when handling complex and multidimensional datasets. Further studies have shown that machine learning techniques are also widely applicable in stock market index prediction [37], event-driven stock prediction [38] and statistical arbitrage strategies [39]. These studies have proven that machine learning outperforms traditional quantitative models in many aspects of actual market applications. Although machine learning has superior performance in prediction, it has to face the loss of model interpretability due to the complexity of the model structure. At the same time, the problems of overfitting and prediction of the emergency market are also worth considering in future research [40].

3. Methodology

3.1. Data Collection and Preprocessing

The EDGAR database of the U.S. Securities and Exchange Commission (SEC) is one of the primary sources for accessing annual report information (10-K reports). As the main source of publicly filed financial reports required from listed companies in the U.S., the EDGAR dataset provides official records of comprehensive corporate financial performance and risk factors. Expanding on this we utilized the EDGAR-CORPUS created by Loukas et al. [41] which can be accessed publicly on Zenodo. This collection comprises reports of all traded companies from 1994 to 2020 meticulously categorized based on specific elements within the reports. We specifically extracted the reports of companies in the S&P 500 index with a focus on the "Item 1A: Risk Factors" section. As per SEC guidelines, Item 1A is required to outline the risks that the company faces which could have an impact, on its operations. This requirement became mandatory following the SEC's regulatory changes in 2005.[1] Since companies gradually started to include comprehensive risk factors in their reports following this change, we focused our analysis on the Item 1A sections from the annual reports spanning from 2010 to 2020 to ensure the quality and consistency of the data regarding disclosed risks.

In the preprocessing stage, we first cleaned the extracted descriptions of risk factors. Since the text information is extracted from XBRL or HTML format files [42], we first need to remove the unprocessed HTML format tags. Secondly, the text may contain some tabular data, which we do not need in this experiment, so we remove them together. In addition, we segmented the text into sentences to lay the foundation for the next step of feature extraction.

Due to the highly time-sensitive and coherent nature of annual reports, and the need to finalize the entire company list for portfolio construction, we opted to split the training and test sets based on the year of publication. Specifically, we used reports from 2010 to 2018 for training, comprising 5,835 documents, and selected annual reports from 2019 to 2020 for testing, comprising 914 documents. The year 2019 was chosen as it represents a relatively stable market environment, while 2020, marked by heightened volatility due to the global pandemic, was selected to evaluate the model's performance under more turbulent market conditions. In total, our dataset contains 6,749 annual report documents.

3.2. Task Definition

We formulate the prediction of Beta as a single objective regression problem. We utilized the text content of "Item 1A: Risk Factor" in the 10-K report as the input for the model, which allowed us to predict a firm's Beta for the next n days. The targets were obtained by taking the average value of the next n days. To calculate Beta, we employed the Capital Asset Pricing Model (CAPM), which is a commonly used calculation method:

$$E(R_i) = R_f + \beta_i(E(R_m) - R_f) \tag{1}$$

The CAPM takes into account the risk-free rate of return (R_f), the expected return of the market ($E(R_i)$), the market risk premium ($E(R_m) - R_f$), and the Beta of the stock (β_i). Specifically, the formula for calculating Beta using CAPM is:

$$\beta_i = \frac{E(R_i) - R_f}{E(R_m) - R_f} = \frac{Cov(R_i, R_m)}{Var(Rm)} \tag{2}$$

To calculate a firm's Beta, it is crucial to determine the covariance between the firm's return and the overall market's return, as well as the variance of the market's return. For this study, we chose the S&P 500 index as the market indicator, corresponding to the sources of the companies we collected.

We assess all approaches by employing the mean squared error (MSE) as the primary metric for evaluation (see Equation 3). MSE is widely used for regression tasks because it penalizes larger errors more heavily, making it particularly suitable for scenarios where significant prediction errors need to

[1] https://www.sec.gov/files/rules/final/33-8591.pdf

be minimized [43, 44]. In our case, the observed value of Beta is β_i, and the predicted value is $\hat{\beta_i}$. The MSE is calculated as the average squared difference between these values, providing a straightforward measure of prediction accuracy.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (\beta_i - \hat{\beta_i})^2 \tag{3}$$

3.3. Prediction Models

In this study, we employed XGBoost, a machine learning model commonly used in financial analysis, to validate its adaptability in the context of annual report analysis [45, 46]. XGBoost is renowned for its efficiency and performance in structured data, which is why we selected it as the baseline model for this experiment.

Furthermore, we assessed the applicability of three Transformer-based pre-training models, which are prominently used in the NLP community for their powerful contextual understanding capabilities. These models included BERT [47], RoBERTa [48], and Longformer [49], known for their deep-learning architectures that capture subtle nuances in text data. However, one challenge with these models is their input token limitation, which typically supports fewer tokens (e.g. BERT accept 512 tokens) than the average total word count found in Item 1A of the annual reports, which is approximately 5965 words. This difference required the use of a truncation method [50], where only the initial tokens of the text are fed into the model. While using a larger pre-train model allows us to process longer texts [51], it still leads to the loss of potentially crucial information appearing later in the text or consumes large computation resources to pre-train.

Algorithm 1 Hierarchical Transformer-based Model

```
function PREDICT_BETA(Document)
    Initialize an empty list: Sentence_Embeddings
    Sentences $\leftarrow$ Split_Into_Sentences(Document)
    for each Sentence in Sentences do
        Tokenized_Input $\leftarrow$ BERT_Tokenizer(Sentence)
        Sentence_Embedding $\leftarrow$ BERT_Encoder(Tokenized_Input)
        Append Sentence_Embedding to Sentence_Embeddings
    end for
    Document_Embedding $\leftarrow$ BERT_Encoder(Sentence_Embeddings)
    Beta $\leftarrow$ Fully_Connected_Layer(Document_Embedding)
    return Beta
end function
for each Document in Batch do
    Beta $\leftarrow$ Predict_Beta(Document)
    Store predicted Beta values.
end for
return all collected Betas
```

To mitigate this limitation and effectively manage the extensive content of Item 1A, we explored text processing techniques referenced in studies by Xie et al. [52], Akbik et al. [53], and Sun et al. [54], which focus on dividing long document into multiple small paragraphs and then putting the embedded segments into the model for processing to adapt with the limit on the number of model input tokens. Inspired by these works and referring to Yang et al. [30] paper on the Hierarchical Transformer-based Multi-task model, we implemented a Hierarchical Transformer-based model (based on BERT model), see Algorithm 1. This model is specifically designed to handle longer text segments by splitting the paragraph into sentences in a layered manner that meets the long document of the annual report, thus preserving more information throughout the text.

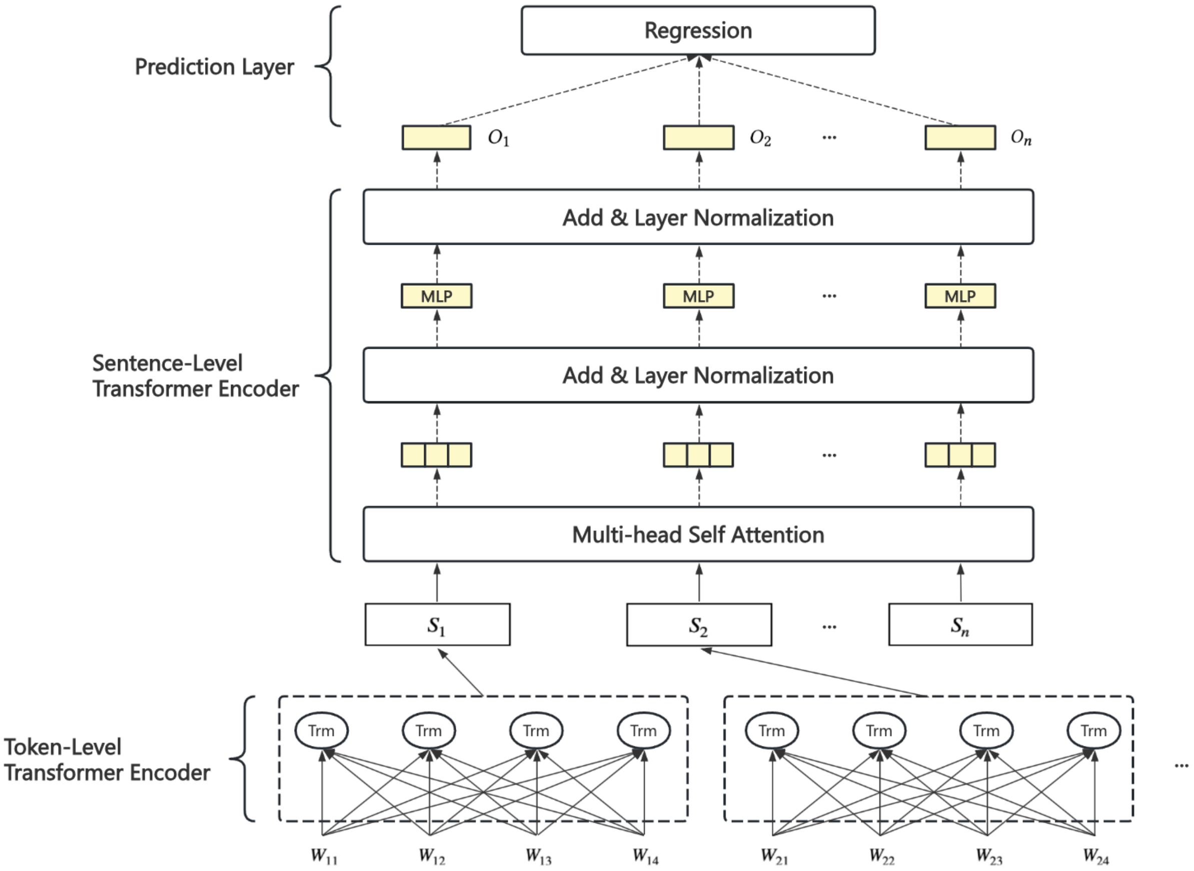
Prediction Layer
Regression
O_1
O_2
O_n
Sentence-Level Transformer Encoder
Add & Layer Normalization
MLP
Add & Layer Normalization
Multi-head Self Attention
S_1
S_2
S_n
Token-Level Transformer Encoder
Trm
W_{11}
W_{12}
W_{13}
W_{14}
W_{21}
W_{22}
W_{23}
W_{24}

Figure 1: Hierarchical Transformer-based Beta (β) Prediction Model

The Hierarchical Transformer-based model (see Figure 1) is applied to Beta (β) prediction from annual report texts through a series of intricate steps. Initially, the input annual report text is preprocessed into sentences and tokenized, forming each sentence into word-level tokens (e.g., $w_{11}, w_{12}, \ldots, w_{1n}$). These tokens are input into the token-level transformer encoder. Each transformer block comprises multi-head self-attention mechanisms and feed-forward neural network layers. The multi-head self-attention mechanism allows the model to focus on different parts of the input tokens in different representation spaces, capturing more complex and diverse dependencies. Simultaneously, positional encodings are added to retain the sequential information of the input tokens, which is crucial for understanding the structure of natural language.

After processing by the token-level encoder, the generated representations (i.e., sentence representations) are input into the sentence-level transformer encoder. The sentence-level transformer encoder has a similar structure, including multi-head self-attention mechanisms and feed-forward neural network layers, but operates at the sentence level. The multi-head self-attention mechanism captures inter-sentence dependencies, allowing the model to understand the context and logical relationships between sentences in the text. After processing through several layers, the output is normalized by addition and layer normalization and processed through multi-layer perceptrons (MLPs), ultimately forming document embedding (e.g., $O_1, O_2, \ldots, O_n$).

During the regression prediction process, in the prediction layer (see Figure 2), the final hidden states outputs (e.g., $O_1, O_2, \ldots, O_n$) are used as inputs for the regression model. These hidden states are processed through a dropout layer, where the dropout layer randomly sets some hidden states to zero, introducing regularization effects to prevent overfitting. The output processed by the dropout layer is

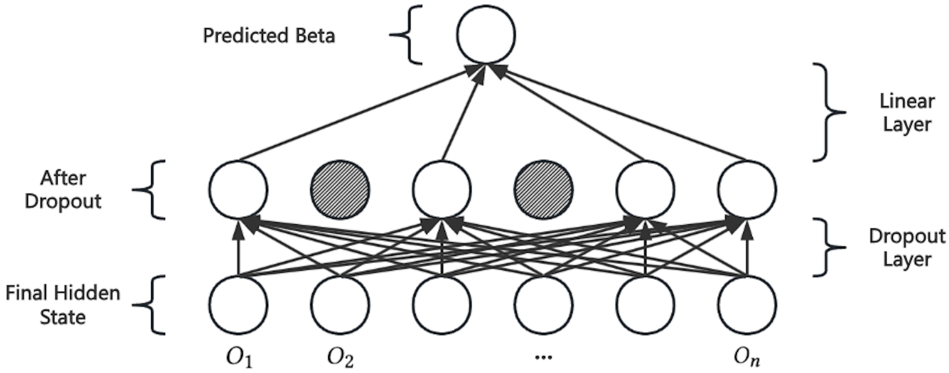

Figure 2: Regression Layer in Beta Prediction: Model Output with Fully Connected Layer

input into a linear layer, which combines this information through weighted aggregation to generate the prediction. The output of the linear layer represents the predicted value of the target variable (Beta), which is used to understand the stock volatility or risk of the company based on the annual report text information.

This method fully leverages the powerful capabilities of transformer models in processing natural language, capturing complex dependencies in textual data. From the word level to the sentence level, the transformer model extracts and aggregates information layer by layer, generating deep representations that can be used for financial predictions. Through this hierarchical representation and regression prediction, the model can avoid the problem of data loss caused by pre-trained models for long text training, and more accurately extract key information from annual report texts that impact the company's stock volatility, thereby improving the accuracy and reliability of Beta predictions.

3.4. Portfolio Construction

Constructing a portfolio that aligns with investment goals and risk tolerance is a critical task in finance. In this paper, we find that utilizing predictive analytics, particularly the predicted Beta values derived from the analysis of annual report texts using a Hierarchical Transformer-based model, can significantly enhance the strategic allocation of assets. This section outlines the methodology employed to build a portfolio based on the Beta predictions, aiming to optimize risk-adjusted returns.

First, we extract the predicted Beta value from the prediction results of each model and sort them according to the predicted Beta value. Second, we extract 20 companies from this prediction result to build a portfolio. According to the principle of asset investment, in order to reduce the risk brought by the portfolio, we adopt a hedging strategy, that is, to take out the 10 companies with the largest Beta values and the 10 companies with the smallest Beta values to form a portfolio. In this way, even when the stock market falls sharply, it can ensure that the assets of the portfolio will not shrink significantly, that is, reduce the portfolio's downside risk.

Then, we optimized the portfolio by constructing the capital market line (CML) to determine the weight of each stock in the portfolio. First, we determined the efficient frontier in the capital market line through CML. Second, through Monte Carlo simulation, we found the portfolio weight that maximizes the return under unit risk, that is, the maximum Sharpe Ratio, see Equation 4. Where R_p is the expected portfolio return, R_f is the risk-free rate, and σ_p is the risk of the portfolio), suggesting that risk-adjusted returns work best.

$$\text{Sharpe Ratio} = \frac{R_p - R_f}{\sigma_p} \tag{4}$$

From the prediction results (see Table 1), we can find that the model has the strongest performance

Table 1
Beta Prediction Using 10-K Report “Item 1A: Risk Factor” Section

Models	**Mean Square Error (MSE)**						
	n=3	n=7	n=15	n=30	n=60	n=90	n=180
XGBoost + TF-IDF	9.53087	1.43068	0.83950	0.30795	0.17176	0.13650	0.09951
BERT (bert-base-uncased)	9.36212	1.40890	0.86511	0.33718	0.18489	0.15794	0.12033
RoBERTa (roberta-base)	9.33685	1.39831	0.82897	0.33769	0.18527	0.15515	0.11957
Longformer (longformer-base-4096)	9.40540	1.43685	0.89168	0.32439	0.17902	0.14855	0.12384
Hierarchical Transformer-based	9.27465	1.41573	0.84309	0.32341	0.17346	0.12015	**0.09634**

in predicting the Beta value of the next 180 days. Therefore, in the portfolio simulation, we choose to monitor the cumulative returns within 180 days after the annual report is released for evaluation. Since the annual reports are mostly released around March, we choose the date of the last company to release the annual report among the 20 selected companies as the observation starting point and compare the cumulative returns of the portfolio in these 180 days with the cumulative returns, comparing to the S&P 500 market. The cumulative income for the portfolio was calculated using Equation 5:

$$R_c = \prod_{i=1}^{180}(1 + \sum_{j=1}^{20} R_{ij}w_{ij}) - 1 \tag{5}$$

Where i represents the days since the start of the observation period, j denotes each of the 20 companies in the portfolio, R represents the return, and w is the weight obtained from the previous CML calculations. This structured approach not only tested the real-world applicability of our Beta predictions but also provided insights into how these predictions could be employed strategically in investment portfolio formulation.

4. Results and Discussion

4.1. Beta Prediction

The experimental results presented in Table 1 compare the effectiveness of various models in predicting Beta values at different future time windows — 3, 7, 15, 30, 60, 90, and 180 days — following the release of annual reports, using the “Item 1A: Risk Factor” section. While the prediction accuracy improved with longer horizons, especially for the 180-day window, the shorter-term forecasts (3-day and 7-day horizons) demonstrated significantly higher error rates. This suggests that the models are more suited to long-term volatility predictions, possibly due to the market’s ability to digest and respond to the risk factors disclosed in annual reports over time.

The predictive accuracy was quantified using MSE, revealing a diverse range of effectiveness among the models evaluated, which included XGBoost, BERT, RoBERTa, Longformer, and the Hierarchical Transformer-based model. Among these, the Hierarchical Transformer-based model demonstrated superior performance for long-term horizons (90 and 180 days), which we attribute to its ability to capture complex dependencies within the risk-related text of annual reports. This model’s hierarchical structure allows for a better understanding of both sentence-level and document-level contexts, making it more effective in capturing nuances of financial risks embedded in long texts.

In contrast, XGBoost’s performance was relatively stable over all time horizons, showcasing its robustness with the best results observed at the 180-day mark. RoBERTa is similar to BERT in terms of performance, especially in the medium to long term. However, because the data is lost due to truncation during training, it does not show better performance than XGBoost. The Longformer model, which is adept at handling extensive text, excelled in short-term predictions and remained competitive over longer periods, a reflection of its architectural benefits for handling lengthy documents. Interestingly, the Hierarchical Transformer-based model, while not performing as well in the short-term predictions,

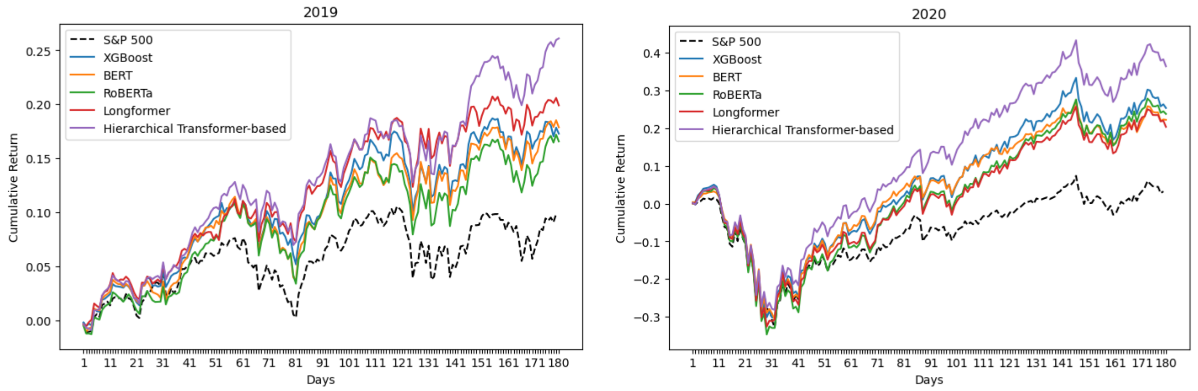

Figure 3: Portfolio Simulation Using the Predicted Beta (β)

demonstrated superior performance for the longer 90 and 180-day horizons. Its hierarchical approach seems particularly well-suited for understanding the complex structure and content of annual reports, enabling it to make more accurate longer-term predictions.

As the prediction horizon extends to 180 days, all models appear to improve in accuracy, suggesting that the market's reaction to the information contained in annual reports becomes clearer and more substantial over time. This trend indicates that market efficiency might increase as information is progressively assimilated by market participants.

In summary, the Hierarchical Transformer-based model exhibited outstanding performance in long-term forecasting, which may be beneficial for long-term investment strategies. However, for investors focusing on the short-term, the Longformer and XGBoost models might be more preferable. The choice of model ultimately depends on the investor's strategy and the desired prediction timeframe, balancing the immediacy of prediction needs against the value of accuracy.

4.2. Portfolio Simulation

Figure 3 shows the cumulative returns for 2019 and 2020 across various forecast models in portfolio construction and evaluated against the S&P 500 benchmark. The simulation results for both years clearly indicate that the portfolios constructed using the predicted Beta values consistently outperformed the S&P 500, particularly in periods of market stability.

In the graph of the year 2019, we observe that after a period of convergence in model performance, all models began to outperform the S&P 500 benchmark. Throughout the year, the Hierarchical Transformer-based model, especially over extended periods, demonstrated a notable lead, signifying its stronger predictive capabilities and suggesting that it may more effectively capture long-term market trends. The performances of XGBoost, BERT, and RoBERTa were fairly similar. The Longformer model, while competitive, exhibited slightly more volatility.

In contrast, the graph of the year 2020 presented a different scenario. During significant market fluctuations, the strategy of selecting hedged stocks worked, resulting in cumulative returns that did not significantly fall below the S&P 500 benchmark. As the market began to recover, the Hierarchical Transformer-based model distinguished itself with a robust upward trajectory, surpassing the benchmark and suggesting that its advantage may be attributed to its nuanced understanding of complex risk factors detailed in annual reports. The other models also showed recovery, indicating that the predicted Beta values offer some guidance for market investment.

Comparing the two years, it's clear to see that the Hierarchical Transformer-based model is more effective at adapting and recovering from market fluctuations than other models. There's a strong correlation in performance trends among the models. However, the advanced structure of the hierarchical model seems to enable it to utilize available information more effectively, particularly in turbulent and long-term investment market conditions. While traditional and NLP-based models can capture market dynamics to a certain extent, the Hierarchical Transformer-based model leads to superior investment portfolio performance due to its approach to integrating the context and structure of financial texts, especially in the face of economic uncertainties. This performance makes it an appealing model for investors seeking robust long-term strategies.

5. Conclusion

The paper mainly discusses the combination of NLP technology and traditional financial analysis, that is, by extracting the risk analysis from the 10-K annual report to predict the Beta value of the relationship between the company and the market. By employing deep learning, we demonstrate the extraction of market-related asset volatility predictions from descriptive information. Our results demonstrate that Hierarchical Transformer-based models possess significant capabilities in long-term Beta prediction. Compared with the traditional transformer model, the design of the hierarchical model is better able to capture the expression of risks in the 10-K report. In investment portfolios constructed by predicting Beta values, deep learning predictions can bring higher returns than traditional quantitative data-based investment portfolios. This provides a more comprehensive view of underlying market behaviour and investment risks.

Furthermore, we found that although financial forecasting is promising in the field of NLP, it is not without challenges. These challenges arise as financial reporting becomes more complex over time and companies' descriptions of risks change. Understanding these new changes in financial reporting will be a significant challenge for existing models. However, with the birth of large language models (LLM), the understanding of financial reporting will bring new changes. In future work, a model focused on annual reports can be trained through a large language model. This helps predict market reactions and company performance more accurately. At the same time, we can also analyze the influence of external factors, such as geopolitical risks or global macroeconomic events, on financial forecasts. By expanding the dataset to include such external variables, we hope to create more sophisticated risk models that integrate textual information from annual reports with other sources of market-relevant data. This multi-source approach will enable us to build a more comprehensive financial forecasting system that can better capture the complexities of modern financial markets.

Acknowledgments

This research was conducted with the financial support of Taighde Éireann – Research Ireland, through the Insight Research Ireland Centre for Data Analytics at University College Dublin. The study also gratefully acknowledges financial support from the China Scholarship Council (CSC), which played a crucial role in supporting the researcher's development and facilitating international collaboration.

References

[1] M.-J. Kim, D.-K. Kang, Ensemble with neural networks for bankruptcy prediction, Expert systems with applications 37 (2010) 3373–3379.

[2] P. C. Tetlock, Giving content to investor sentiment: The role of media in the stock market, The Journal of finance 62 (2007) 1139–1168.

[3] S. Kogan, D. Levin, B. R. Routledge, J. S. Sagi, N. A. Smith, Predicting risk from financial reports with regression, in: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, 2009, pp. 272–280.

[4] H. A. Javaid, Ai-driven predictive analytics in finance: Transforming risk assessment and decision-making, Advances in Computer Sciences 7 (2024).

[5] S. Yıldırım, D. Jothimani, C. Kavaklıoğlu, A. Başar, Classification of" hot news" for financial forecast using nlp techniques, in: 2018 IEEE International Conference on Big Data (Big Data), IEEE, 2018, pp. 4719–4722.

[6] W. F. Sharpe, Capital asset prices: A theory of market equilibrium under conditions of risk, The journal of finance 19 (1964) 425–442.

[7] J. Y. Campbell, T. Vuolteenaho, Bad beta, good beta, American Economic Review 94 (2004) 1249–1275.

[8] J. Lintner, The valuation of risk assets and the selection of risky investments in stock portfolios and capital budgets, in: Stochastic optimization models in finance, Elsevier, 1975, pp. 131–155.

[9] E. F. Fama, K. R. French, Common risk factors in the returns on stocks and bonds, Journal of financial economics 33 (1993) 3–56.

[10] P. Jorion, et al., Financial risk manager handbook, volume 406, John Wiley & Sons, 2007.

[11] V. Vapnik, S. Golowich, A. Smola, Support vector method for function approximation, regression estimation and signal processing, Advances in neural information processing systems 9 (1996).

[12] T. Loughran, B. McDonald, When is a liability not a liability? textual analysis, dictionaries, and 10-ks, The Journal of finance 66 (2011) 35–65.

[13] C. K. Theil, S. Štajner, H. Stuckenschmidt, Word embeddings-based uncertainty detection in financial disclosures, in: Proceedings of the First Workshop on Economics and Natural Language Processing, 2018, pp. 32–37.

[14] Y. Qin, Y. Yang, What you say and how you say it matters: Predicting stock volatility using verbal and vocal cues, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, pp. 390–401.

[15] S.-H. Poon, C. W. J. Granger, Forecasting volatility in financial markets: A review, Journal of economic literature 41 (2003) 478–539.

[16] P. P. Pompe, A. Feelders, Using machine learning, neural networks, and statistics to predict corporate bankruptcy, Computer-Aided Civil and Infrastructure Engineering 12 (1997) 267–276.

[17] F. Mai, S. Tian, C. Lee, L. Ma, Deep learning models for bankruptcy prediction using textual disclosures, European journal of operational research 274 (2019) 743–758.

[18] T.-K. Chen, H.-H. Liao, G.-D. Chen, W.-H. Kang, Y.-C. Lin, Bankruptcy prediction using machine learning models with the text-based communicative value of annual reports, Expert Systems with Applications 233 (2023) 120714.

[19] E. F. Fama, K. R. French, Size and book-to-market factors in earnings and returns, The journal of finance 50 (1995) 131–155.

[20] E. F. Fama, Random walks in stock market prices, Financial analysts journal 51 (1995) 75–80.

[21] D. M. Cutler, J. M. Poterba, L. H. Summers, What moves stock prices?, volume 487, National Bureau of Economic Research Cambridge, Massachusetts, 1988.

[22] R. J. Shiller, et al., Do stock prices move too much to be justified by subsequent changes in dividends? (1981).

[23] Z. Ye, Y. Qin, W. Xu, Financial risk prediction with multi-round q&a attention network., in: IJCAI, 2020, pp. 4576–4582.

[24] R. J. Shiller, Irrational exuberance: Revised and expanded third edition (2015).

[25] V. L. Bernard, J. K. Thomas, Post-earnings-announcement drift: delayed price response or risk premium?, Journal of Accounting research 27 (1989) 1–36.

[26] R. Sadka, Momentum and post-earnings-announcement drift anomalies: The role of liquidity risk, Journal of Financial Economics 80 (2006) 309–349.

[27] D. Kong, L. Gao, Explaining stock price movements: Is it news or noise?, Journal of Business Finance & Accounting 38 (2011) 579–605.

[28] C. K. Theil, S. Štajner, H. Stuckenschmidt, Explaining financial uncertainty through specialized word embeddings, ACM Transactions on Data Science 1 (2020) 1–19.

[29] S. Hansen, M. McMahon, A. Prat, Transparency and deliberation within the fomc: a computational linguistics approach, Quarterly Journal of Economics 133 (2018) 801–870.

[30] L. Yang, T. L. J. Ng, B. Smyth, R. Dong, Html: Hierarchical transformer-based multi-task learning for volatility prediction, in: Proceedings of The Web Conference 2020, 2020, pp. 441–451.

[31] J. Bollen, H. Mao, X. Zeng, Twitter mood predicts the stock market, Journal of Computational Science 2 (2011) 1–8.

[32] M. C. Jensen, et al., Studies in the theory of capital markets, The Journal of Finance (1972).

[33] M. M. Carhart, On persistence in mutual fund performance, The Journal of finance 52 (1997) 57–82.

[34] K.-j. Kim, Financial time series forecasting using support vector machines, Neurocomputing 55 (2003) 307–319.

[35] W. Huang, Y. Nakamori, S.-Y. Wang, Forecasting stock market movement direction with support vector machine, Computers & operations research 32 (2005) 2513–2522.

[36] L. Niu, X. Xu, Y. Chen, An adaptive approach to forecasting three key macroeconomic variables for transitional china, Economic Modelling 66 (2017) 201–213.

[37] J. Patel, S. Shah, P. Thakkar, K. Kotecha, Predicting stock market index using fusion of machine learning techniques, Expert systems with applications 42 (2015) 2162–2172.

[38] X. Ding, Y. Zhang, T. Liu, J. Duan, Deep learning for event-driven stock prediction, in: Twenty-fourth international joint conference on artificial intelligence, 2015.

[39] C. Krauss, X. A. Do, N. Huck, Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the s&p 500, European Journal of Operational Research 259 (2017) 689–702.

[40] Y. Peng, M. H. Nagata, An empirical overview of nonlinearity and overfitting in machine learning using covid-19 data, Chaos, Solitons & Fractals 139 (2020) 110055.

[41] L. Loukas, M. Fergadiotis, I. Androutsopoulos, P. Malakasiotis, Edgar-corpus: Billions of tokens make the world go round, arXiv preprint arXiv:2109.14394 (2021).

[42] L. Loukas, M. Fergadiotis, I. Chalkidis, E. Spyropoulou, P. Malakasiotis, I. Androutsopoulos, G. Paliouras, Finer: Financial numeric entity recognition for xbrl tagging, arXiv preprint arXiv:2203.06482 (2022).

[43] L. Cao, F. E. Tay, Financial forecasting using support vector machines, Neural Computing & Applications 10 (2001) 184–192.

[44] T. Chai, R. R. Draxler, et al., Root mean square error (rmse) or mean absolute error (mae), Geoscientific model development discussions 7 (2014) 1525–1534.

[45] A. A. Ali, A. M. Khedr, M. El-Bannany, S. Kanakkayil, A powerful predicting model for financial statement fraud based on optimized xgboost ensemble learning technique, Applied Sciences 13 (2023) 2272.

[46] B. Quinto, Next-generation machine learning with spark: Covers XGBoost, LightGBM, Spark NLP, distributed deep learning with keras, and more, Apress, 2020.

[47] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, CoRR abs/1810.04805 (2018). URL: http://arxiv.org/abs/1810.04805. arXiv:1810.04805.

[48] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized BERT pretraining approach, CoRR abs/1907.11692 (2019). URL: http://arxiv.org/abs/1907.11692. arXiv:1907.11692.

[49] I. Beltagy, M. E. Peters, A. Cohan, Longformer: The long-document transformer, CoRR abs/2004.05150 (2020). URL: https://arxiv.org/abs/2004.05150. arXiv:2004.05150.

[50] A. Merchant, E. Rahimtoroghi, E. Pavlick, I. Tenney, What happens to bert embeddings during fine-tuning?, arXiv preprint arXiv:2004.14448 (2020).

[51] J. Ainslie, S. Ontanon, C. Alberti, V. Cvicek, Z. Fisher, P. Pham, A. Ravula, S. Sanghai, Q. Wang, L. Yang, Etc: Encoding long and structured inputs in transformers, arXiv preprint arXiv:2004.08483 (2020).

[52] Q. Xie, Z. Dai, E. Hovy, T. Luong, Q. Le, Unsupervised data augmentation for consistency training, Advances in Neural Information Processing Systems 33 (2020) 6256–6268.

[53] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, R. Vollgraf, Flair: An easy-to-use framework for state-of-the-art nlp, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations), 2019, pp. 54–59.

[54] C. Sun, X. Qiu, Y. Xu, X. Huang, How to fine-tune bert for text classification?, in: China national conference on Chinese computational linguistics, Springer, 2019, pp. 194–206.