Can AI Replace Stock Analysts? Evidence from Deep Learning Financial Statements

G. Nathan Dong [*]

September 10, 2025

`

ABSTRACT

We apply a model-free deep-learning technique to equity research by training a neural-network model using financial statement, interest rate and stock return data. This simple AI model outperforms human analysts in predicting equity returns by a large margin. Analysts and AI models are more likely to make similar predictions for firms that are in a better financial position, have higher institutional ownership, and show signs of lower market uncertainty on the day of the forecast. Overall, the results of this research support the notion that AI has the potential to replace human analysts in certain aspects of predicting financial performance.

Keywords: Analyst forecast, Artificial intelligence, Deep learning, Fintech
JEL Codes: C10, C45, G30, G23

[*] Department of Finance, Carroll School of Management, Boston College, Chestnut Hill, MA 02467. Tel: (617) 552-6426. E-mail: nathan.dong@bc.edu.. All errors remain our responsibility.


"All the News That's Fit to Print"
The New York Times
Late Edition
Today, plenty of sunshine, chilly, high 36. **Tonight,** increasing cloudiness, low 29. **Tomorrow,** intervals of snow and rain, little to no accumulation, high 41. Weather map, Page 30.

The Robots Are Coming for Wall Street

Hundreds of financial analysts are being replaced with software. What office jobs are next?

— The New York Times †

INTRODUCTION

Equity research analysts play an important role in the financial market by interpreting public information, discovering new information, and providing recommendations.‡ Part of their job is to utilize their knowledge and analysis of the market to forecast price targets for stocks, which are then used by investors to make informed decisions (Brav and Lehavy, 2003). It involves analyzing financial statement data, studying market trends, and offering educated speculations about the future performance of a particular stock (Beyer et al., 2010).§ However, with the advancements in artificial intelligence (AI) technology, it naturally begs the question of whether AI has the capability to replace this otherwise repetitive and labor-intensive task of forecasting target prices.

The use of AI in predicting performance has attracted considerable attention as an alternative method to traditional equity research. Utilizing advanced algorithms and machine learning techniques, AI systems can analyze vast amounts of data to identify patterns and trends that could be unnoticed by even the most experienced professionals. Specifically, the deep-learning technique based on neural network has been applied to various fields and industries such as entertainment (Igami, 2020), monetary policy (Gorodnichenko et al., 2023),

† This is the title of a NYT article contributed by Nathaniel Popper and published in the New York Times' Sunday Magazine with the headline "Stocks & Bots" on February 28, 2016.

‡ The relative importance of analysts' role in information interpretation versus information discovery has been studied in the accounting literature (Livnat and Zhang, 2012).

§ The focus of this study is on sell-side analysts, rather than buy-side analysts, but they are closely related because sell-side analysts often provide buy-side analysts with in-depth industry knowledge and access to company management (Brown et al., 2016).

macroeconomics (Maliar et al., 2021), microeconomics (Khachiyan et al., 2022), and corporate governance (Erel et al., 2021). Previous studies have shown that AI is certainly capable of analyzing complex datasets and finding non-linear relationships between different financial variables in a very small fraction of time compared to that of a human analyst. This could potentially lead to more accurate predictions with a reduced margin of error. In addition, AI is not influenced by emotions, which sometimes can cloud the judgment of human analysts, thereby eliminating possible biases (Green et al., 2024).

The debate over whether artificial intelligence can outperform human stock analysts in predicting performance remains ongoing. While AI has demonstrated remarkable capabilities in analyzing huge amounts of data with a high level of speed and accuracy and making predictions based on patterns and probabilities, human analysts bring in intuition, experience, and an understanding of market dynamics that cannot be replicated by a machine (Joos et al., 2016). Undoubtedly, AI lacks the intuition and critical thinking skills that human analysts possess because equity research is not just about crunching numbers; it also involves understanding the underlying factors that may be affecting the performance of a stock. This kind of qualitative analysis can never be perfectly replicated by machines (Iskhakov et al., 2020). Indeed, human-defined variables, like competition measures such as HHI, or accounting ratios such as ROA, that has been used by analysts, are often included in the AI deep learning (Cao et al., 2024). The use of pre-specified models or pre-defined ratios in AI deep learning can certainly make the predictions more reliable. Unfortunately, no analysis has attempted to compare the performance of analyst forecasts with that of model-free deep-learning, which involves training a raw neutral network that has no prior knowledge of financial models and accounting ratios.

The purpose of this paper is to fill this gap by offering an empirical comparison of the accuracy in forecasting 12-month-ahead stock returns by both human analysts and AI model-free predictions.[**] The reason it focuses on predicting stock returns rather than prices is that stock prices are typically non-stationary and often exhibit random walk behavior (i.e., unit root), meaning their statistical distribution changes over time (Campbell et al., 2012). In contrast, stock returns tend to have mean-reverting behavior, and thus more suitable for statistical analysis (Fama and French, 1988). In other words, predicting returns is more robust and reliable than predicting prices, as returns are generally more stationary and less prone to trending behavior (Cochrane, 2008), and on the other hand, price predictability is more sensitive to market conditions and subject to significant estimation errors (Campbell and Thompson, 2008).

In this research, we develop a simple neural-network model and use it to deep learn raw data of historical financial statement, interest rate, and stock return. As Birru, Gokkaya, Liu and Stulz (2022) point out, analyst target prices are typically derived from an analyst's long-term evaluations of a firm's prospects over an extended period. These evaluations are based on a combination of fundamental information and financial models (Birru et al., 2022). Why can't an AI model, which has access to the same financial data, develop its own model to accurately forecast target stock returns 12 months in advance? The first challenge is determining the type of data that should be used to train the AI model to ensure reliable prediction and fair comparison with human analysts. Training a neural-network model using a comprehensive dataset over an extended period of time makes the comparison between humans and AI unfair. In this research, we utilize three sets of historical data: quarterly financial statements, interest rates, and stock returns.

A second challenge is deciding which model to build. If we simply use a pre-built model like the discounted cash flow (DCF) and let the computer to predict stock returns, the

[**] The price forecasted by the analysts, also known as the target prices are predominantly one-year-ahead prices (Brav and Lehavy, 2003).

comparison is no longer between human analysts and AI, but rather a replacement of humans by machines. We prefer a model-free algorithm, which is the estimation of an optimal model that is free-form, meaning it is not constrained by predetermined structures or assumptions. In this paper, we rely on the deep-learning capabilities of neural-network models to explore and interpret data in ways that cannot be achieved through conventional methods. The unpredictable nature of the stock market and external factors make it challenging for both humans and AI to consistently predict future performance accurately. Free-form modeling using neural network offers a more holistic understanding of historical data. This approach allows for more nuanced and dynamic representations of a firm's past performance, potentially leading to deeper insights and more accurate predictions.

This paper makes two contributions to the literature. First, it provides empirical evidence of the deep-learning capability of neural-network models to accurately forecast future returns. It identifies the scope of the data needed to train the model, specifically historical financial statement, interest rate, and stock return data. After a 3,000-epoch training that takes 15 hours and costs less than three dollars in electricity, a basic AI model running on $1,500 computer equipment outperforms human analysts in predicting stock returns 12 months ahead. Secondly, this research examines the underlying economic factors that drive the deviation in predicted target returns between analysts and the AI model. It shows that cross-sectionally, the forecast difference is smaller in firms that are larger and have higher market-to-book, lower leverage, more profits, less cash relative to assets, lower R&D-to-assets, lower return volatility, and higher institutional ownership on the day of the forecast. However, within each firm over time, the disagreement on forecasted returns between human analysts and the AI model is smaller when firm size increases and cash holding decreases while the effects of other factors such as return volatility and institutional ownership remain unchanged. Overall, the results of this research support the notion that AI has the potential to replace human analysts in certain aspects of predicting financial performance.

The remainder of the paper is organized as follows. Section II introduces the neutral-network model and the technical details of model training and prediction. Section III presents the sample data and measurement choices. Section IV analyzes the model performance by comparing the returns between analyst forecasts and AI predictions and conduct robustness tests. Section V investigates the determinants of prediction deviations between analysts and the AI model. Section VI provides a summary and concluding marks.

II. METHOD OF MODEL-FREE DEEP LEARNING

Deep learning, sometimes referred to as artificial neural networks, is a machine learning technique that can train computer models to do what comes naturally to humans. It is able to handle a large number of input variables and is not affected by multicolinearity. This is because neural networks use multiple hidden layers to process the input variables and capture the relationships between them. Each hidden layer in a neural network learns and extracts different features from the input data, reducing the impact of multicolinearity (De Veaux and Ungar, 1994). In contrast, linear regression models like OLS (Ordinary Least Squares) rely on the assumption of independent predictor variables, and when multicolinearity is present, it becomes difficult to determine which variable is actually contributing to the outcome. This can result in unreliable and inflated parameter estimates, rendering the model less effective in making accurate predictions (Dormann et al., 2013). Additionally, neural networks are able to handle missing data and outliers effectively, which can further contribute to the accuracy of their predictions.

Therefore, we believe that using a neural network to deep-learn a panel dataset of financial statements and other market data for predicting risk and outcomes can be a promising approach that can offer valuable insights for investors and financial analysts. For example, a deep-learning model has been developed to study the non-linear relationship between mortgage risk factors and loan performance. The nonlinearities revealed by the deep-learning model have profound implications for risk management, investment management, and mortgage-backed securities (Sadhwani et al., 2021).[††]

2.1. Neural network model and in-sample data training

In this study, we train a multi-layer neural-network model using a panel data of financial statements, stock returns and interest rates. Training this model is a gradual and repetitive process aimed at enhancing the model's capacity to analyze historical data and generate predictions (Nokhwal et al., 2024). At its core, a neural network is composed of layers of interconnected nodes that collaborate to process input data and produce output (as shown in Figure 1). These layers are essential for the functioning of a neural network, as each layer performs a specific task that contributes to the overall prediction made by the network. The first layer, known as the input layer, is responsible for receiving the financial statement and other market data and passing it on to the next layer. This layer plays a crucial role in the prediction process by establishing the initial values for the network's weights and biases. The next layer is the hidden layer, where most of the complex computations take place. This layer is responsible for extracting features from the input data, which are then passed on to the next layer. The number of hidden layers in a neural network can vary depending on the complexity of the problem being solved. These layers are where the network's learning takes place, as the weights

[††] A similar work can be found in using machine learning to predict loan default using European data (Barbaglia et al., 2023).

and biases are continuously adjusted based on the input data. Finally, the output layer is where the prediction is made. This layer takes the information from the hidden layers and produces the final output, which is the predicted 12-month stock returns (to be compared with the actual returns). It is important to note that we use five layers (one input, three hidden, and one output layers) in the neural-network. The choice of this number is completely arbitrary because we want to keep the model as simple as possible while maintaining the level of sophistication needed to make the predictions as accurate as possible. In the robustness section, we will evaluate the performance of a 4-layer model and 6-layer model.

[Insert Figure 1]

During the training process, the neural-network layers undergo a series of transformations known as forward propagation, which ultimately enhance the model's capability to accurately predict the stock returns. At the beginning of the training process, the weights and biases of the neural network layers are randomly initialized. These weights and biases determine the strength of the connections between neurons and the threshold at which they fire. As the training progresses, the model is presented with a set of in-sample training data, and the input data is fed forward through the layers of the neural network. As the data passes through each layer, the outputs of neurons are calculated using the current weights and biases. The output of the final layer (predicted 12-month-ahead returns) is then compared to the desired output (realized stock returns), and the difference between the two is measured using a loss function.

2.2. Loss function and optimizer

The Mean Absolute Error (MAE) function is used to compute the error or loss of the model. Compared to other statistical loss functions, MAE is more robust to outliers, helping reduce the impact of these outliers on the model's predictions (Campbell et al., 2012). The MAE loss function directs the optimizer to minimize the average absolute differences between the predicted and the actual 12-month-ahead stock returns. More specifically, the optimizer, a mathematical function based on the concept of gradient descent (i.e., the iterative reduction of the loss function by following the gradient in a greedy manner), adjusts the weights of each neuron using the gradients. Its function can be as simple as subtracting the gradients from the weights. More complex optimizers can be faster and more efficient, dramatically improving model performance. We chose the Adaptive Moment Estimation (ADAM) optimizer, which utilizes adaptive optimization methods, for our model training because it consistently converges faster and has become the default optimization algorithm in many social science research fields. The learning rate of the Adam optimizer in our models is set to 0.0001 in Pytorch, an open-source machine-learning library using the Python programming language (Stevens et al., 2020). Specifically in the Python code, the MAE loss function and the ADAM optimizer with automatic mixed precision (AMP) and a cosine scheduler with warm restarts:

```
loss_fn = L1Loss(reduction='mean')
optimizer = Adam(params=model.parameters(), lr=0.0001)
scaler = GradScaler(enabled=True)
scheduler = CosineAnnealingWarmRestarts(optimizer, T_0=int(epochs/2))
```

The purpose of using a cosine scheduler with warm restarts is to adjust the learning rate according to a cosine function, and at the same time to periodically reset the optimization process with updated initial conditions. The combination of a cosine annealing and warm restarts can help the model escape local minima by periodically increasing the learning rate. The goal of training a neural network is to minimize the loss function by adjusting the weights and biases of neurons in each layer. This is done through backpropagation, where the error (i.e., the value from the loss function using MAE) is propagated backward from the output layer to the hidden layers, and the weights and biases are updated accordingly. Backpropagation allows the network to learn from its mistakes in the current epoch and improve its predictions in the subsequent epoch. This process is repeated multiple times (epochs), with the weights and biases being adjusted after each iteration, until the network reaches a point where the error is minimized, indicating that the predictions of returns are accurate. As the weights and biases are updated, the neural network layers undergo a process called learning. Neurons that receive similar inputs most of the time will have their connections strengthened. Neurons that receive rare inputs will have their connections weakened. This process of learning helps the network to become more specialized in recognizing patterns and making accurate predictions. Automatic mixed precision (AMP) is used by the optimizer to reduce memory usage and speed up computations without compromising the model's accuracy by combining two levels of numerical precision—single-precision (32-bit) and half-precision (16-bit) floating-point formats—during the training process (Kotipalli et al., 2019). In the Python code, this is specified as:

```
for epoch in range(3000):
    model.train()
    optimizer.zero_grad()
    with torch.cuda.amp.autocast(enabled=True, dtype=torch.float16):
        y_pred = model(x_train).squeeze()
        loss = loss_fn(y_pred, y_train)
    scaler.scale(loss).backward()
    scaler.step(optimizer)
```

```
        scaler.update()
        scheduler.step()
```

2.3. Regularization technique

As training progresses, the layers of neurons become more efficient at extracting important characteristics from the data. Additional fine-tuning is made by modifying the weights and biases to give more importance to specific features that are more valuable for making accurate prediction of returns. This process is commonly referred to as feature learning. In addition to the tuning of the weights and biases during training, we also apply a regularization techniques called dropout with a dropout rate of 10% to avoid over-fitting by inducing the model to learn more general patterns from new data (Srivastava et al., 2014). During training, the model will take the output from its previous layer, randomly select 50% of the neurons, zero them out, and then pass the output to the next layer, effectively ignoring them. Essentially, this involves randomly dropping a half of the neurons in the network, requiring the other neurons to take over and represent the information needed to make the predictions for the remaining neurons. This may thus force the network to become less sensitive to the exact weights of neurons, potentially reducing the chances of over-fitting the training data and ultimately improving the model's accuracy. In the Python code, dropout regularization can be specified as below, where the dropout rate is 10%:

```
def __init__(self):
    self.dropout = nn.Dropout(p=0.1)
def forward(self, x):
    return self.linear5(self.dropout(self.relu(self.linear4(...
```

2.4. Out-of-sample prediction and total costs of training

Once the neural network has been trained, it is then used to make predictions on the out-of-sample panel dataset. The new dataset (financial statement and other market data that are one calendar quarter later than the in-sample data) is fed into the network, and the forward propagation process begins. The data is processed through the layers of neurons until it reaches the final layer, where the prediction is produced. Of course, the accuracy of the prediction depends on the quality of the training data and the complexity of the neural network. In essence, a trained neural network makes predictions by leveraging its capacity to identify patterns and relationships in the data it receives, along with the knowledge it has acquired during the training process.

Table 1 lists the equipment and software used for this research. Because the operating system (Ubuntu) and programming tool (Python) are open-source, which means they are free and available for everybody to use for any purpose, the total costs of this research mainly come from the hardware equipment ($1,450), with half of it allocated to an Nvidia GPU ($700). On the other hand, while each training process includes 3,000 epochs and takes about 15 hours to finish, the electricity consumption ($2.52) is minuscule compared to the vast amount of energy used in mining cryptocurrencies (Dong, 2024) and training large language model (LLM) models like the ChatGPT (Yang et al., 2024).

[Insert Table 1 Here]

III. SAMPLE DATA

The primary source of data for training the neural-network model is a combination of three sources: Compustat, CRSP, and IBES. The quarterly and year-end financial statement data from the Compustat 10Q and 10K reports from 2010 to 2023 are linked to the CRSP monthly stock file and the IBES-CRSP Link (Beta) from the WRDS to calculate the month-end market value and ensure that each firm-quarter observation includes at least one analyst's forecast from the IBES database. This generates a panel dataset with 175,718 firm-quarter observations. It is then merged with the IBES Detail History's Price Target, and the forecast period is restricted to 12 months. Both the stock price on the forecast day and the actual realized price after the 12-month period are collected only if the price level is lower than $1,000 to prevent potential distortions caused by outliers. The initial stock price and the 12-month realized price are used to calculate the realized return, and similarly, the initial price and the analyst's 12-month target price are used to calculate the analyst-forecasted return. In addition, four historical stock returns (3-month, 6-month, 9-month, and 12-month) prior to the forecast day are also included. Because there are multiple analysts covering the same firm in the same quarter, the sample size increases to 284,305.

One of the tools that equity research analysts often utilize to predict future stock price is the DCF model. The cost of capital is a key input in DCF analysis, as it directly impacts the present value of the future cash flows. We can simplify the estimation of a firm's cost of capital by using a risk-free rate and a risky rate as proxies for the rate that a company pays on its funds. We collect two types of interest rate data from the Federal Reserve Economic Database maintained by the Research Division of the Federal Reserve Bank of St. Louis. The market yield on U.S. Treasury securities at a 10-year constant maturity (TCMNOM-Y10) is used to represent the risk-free rate, while the Moody's Seasoned Baa Corporate Bond Yield (DBAA) serves as a proxy for the cost of capital associated with higher investment risk.

To construct the training data for the neural-network model in deep learning, each row (firm-quarter) of the panel dataset includes the historical stock returns (on the day of the forecast, 3-month prior, 6-month prior, 9-month prior, 12-month prior, and 12-month later), interest rates (10-year risk-free and 1-year Baa-grade), and financial statements with data from the four prior quarters. In the Compustat 10Q/10K dataset, there are 598 unique accounting variables, many of which have missing values in almost every reporting period. After dropping the mostly blank-value accounting variables, there are 154 variables that have non-missing values at least half of the time. The detailed list of accounting variables included in the training data and their definitions can be found in Table 2. The training data consists of a total of 167 columns, which include firm identification (2 columns), time (3 columns), stock returns (6 columns), interest rates (2 columns), and financial statement items (154 columns).

[Insert Table 2 Here]

During the training, the Python code will also locate three prior quarters of financial statement data (154 × 3 = 462) for each firm-quarter observation and append them to the end of each row, increasing the number of variables to 629 inside the Python and Nvidia GPU's memory. The entire dataset is dynamically split into the in-sample dataset for training and the out-of-sample dataset for predictions over time. For example, to make a 12-month prediction in the first quarter of 2022 (Q1), the neural network is trained for 3,000 epochs using the four consecutive quarterly financial statements from 2020 (Q1, Q2, Q3, and Q4). After the model is trained, it uses the quarterly financial statement data from 2022 Q1 to predict stock returns. Because it is a 12-month prediction, it is forecasting the returns for the first quarter of 2023 (Q1). This predicted stock returns will be compared with the analyst-forecasted returns and the realized stock returns in Q1 of 2023.

IV. EMPIRICAL ANALYSIS

4.1. Evaluating AI predictions of stock returns (benchmark models)

Once the neural network has been trained, it can be used to make predictions on new data that it has not seen before. The input data is fed into the network, and the forward propagation process begins. The data is processed through the layers of neurons until it reaches the final layer, where the prediction is produced. The model compares the predicted returns with the actual returns and calculates the MAE of the prediction. This helps improve the calibration of the model in the next epoch of the training process. After 3,000 epochs of training with the dropout regularization techniques using the in-sample dataset, the model makes predictions based on the out-of-sample data. Of course, the accuracy of the prediction depends on the quality of the training data and the complexity of the neural network. The main neutral model used in this research has a total of five layers, and the input size of the first layer is set to the number of variables in the in-sample dataset.

In the first model, we only use data from the financial statement, so the input size is four times the number of accounting variables (154 × 4 = 616). In the Python code, this is specified as:

```
self.linear1 = nn.Linear(616, 4096)
self.linear2 = nn.Linear(4096, 8192)
self.linear3 = nn.Linear(8192, 4096)
self.linear4 = nn.Linear(4096, 2048)
self.linear5 = nn.Linear(2048, 1)
```

It is noted that the choice of the number of epochs (3,000), the number of layers (5) and the size of each layer (2048, 4096, and 8192) in the neutral-network model is entirely arbitrary. As a robustness check, we will also train a 4-layer model with 2,000 epochs and another 6-layer model with 4,000 epochs.

In order to evaluate the quality of this AI model's performance in predicting returns, we can compare the 12-month predicted return (R_{AI}) with the realized stock return after 12 months (R_{Realized}) by estimating an accuracy function in the following equation:

$$Accuracy_{\mathrm{AI}} = \left| R_{\mathrm{AI}} - R_{\mathrm{Realized}} \right| \tag{1}$$

The measure is the absolute difference between the realized stock return and the AI-predicted return. A lower value of this measure indicates a higher ability to predict future performance. Similarly, we can define the accuracy of analyst forecasts by comparing the target price-derived return (R_{Analyst}) with the actual stock return (R_{Realized}) as:

$$Accuracy_{\text{Analyst}} = \left| R_{\text{Analyst}} - R_{\text{Realized}} \right| = \left| \frac{P_{\text{Analyst}} - P_0}{P_0} - R_{\text{Realized}} \right| \tag{2}$$

In Table 3 Panel A, the average accuracy of analyst forecasts using financial statement data from 2010 to 2023 is reported in the first row of column (1). The statistically significant accuracy measure (0.4266) and the standard error (0.0008) suggests that, on average, the analysts set inaccurate target prices. This is generally consistent with some earlier findings indicating that analysts have limited abilities to provide accurate target price forecasts (Bradshaw et al., 2013), but it does not necessarily agree with other studies (Bilinski et al., 2013). However, the focus of this research is on the out-of-sample prediction ability of the AI model after being trained on the in-sample data. The second row of Column (2) reports the accuracy measure for the AI model (0.2523), and the difference between the accuracy of analyst forecasts and that of AI predictions (0.1743) is statistically significant at the one-percent level with a standard error of 0.0008. In other words, the AI model trained on financial statement data shows a much higher ability to forecast the 12-month-ahead stock returns than human analysts do.

[Insert Table 3 Here]

The poor accuracy of analyst forecasts deserves attention because both human analysts and the AI model are exposed to the same historical accounting data. The significant deviation in accuracy suggests that either human analysts were incompetent or biased when studying companies with these characteristics. However, it is also likely that the announcement of the analyst's target price has caused market over-reaction, leading to a change in the subsequent market price. This is consistent with the evidence that analysts create value for firms under their coverage by enhancing their investor recognition rather than by monitoring or reducing information asymmetry (Li and You, 2015). In other words, the analyst may have made an accurate forecasts, but his or her target price has caused a significant drift in the realized price over the next 12 months (Dechow and You, 2020).

In the second model, the AI model is trained on interest rate data. Interest rates are a crucial element of a country's monetary policy and are closely monitored by analysts because they can impact the cost of borrowing and spending. This, in turn, can affect stock prices and returns as companies may adjust their financing and investment decisions based on interest rates. For instance, when interest rates are low, companies may be more inclined to borrow and invest in expanding their business. This can lead to potential growth and an increase in stock prices. Moreover, equity research analysts also pay attention to central bank decisions on interest rates, as any changes in these rates can significantly impact the economy and the stock market. If the central bank raises interest rates, it can result in higher costs for businesses, potentially leading to slower economic growth and a decrease in stock prices. On the other hand, a decrease in interest rates may signal a strong economy and result in increased consumer spending, which could have a positive impact on future stock returns. Equity research analysts utilize interest rate data not only to monitor interest rates on a macro level but also to analyze specific companies. If a company has a high level of debt, an increase in interest rates may raise their cost of borrowing, ultimately affecting their earnings and stock returns. Conversely, a company with a strong balance sheet and low debt may be less affected by changes in interest rates.

The second row in Table 3 Panel A reports the difference in accuracy measures between analyst forecasts and AI predictions. While the AI model that deep learns from interest rate data also outperforms human analysts with a statistically significant difference (0.1326), the difference is actually smaller than the previous model that learns from financial statement data. In the third model, it is trained on historical stock returns including 3-month, 6-month, 9-month, and 12-month returns prior to the day of the forecast. The model performs poorly compared to the first and second models, but it still outperforms human analysts. The difference in prediction accuracy between the analyst forecast and the third model trained on prior stock returns is 0.0646 with a statistical significance at the one-percent level (third row in Table 3 Panel A).

The fourth model is trained on both financial statement and interest rate data and the fifth model is trained on both financial statement and stock return data. The accuracy of the fourth model (0.2538) and that of the fifth model (0.2508) are very similar to the measure of the first model which uses only the financial statement data (0.2523); however, they are statistically different to the accuracy measures of both the second model trained on interest rate (0.2940) and the third model trained on stock return (0.3620).‡‡

The sixth model does not use financial statement data; instead, it is trained on interest rate and stock return data. Its accuracy measure (0.3422) is closer to the accuracy of the third model trained on stock return (0.3620) than the second model on interest rate (0.2940). Still, the accuracy difference between this model and the stock return model is statistically significant with a t-statistic of 25.35 from the independent two-sample t-test (with equal sample sizes and variances). Finally, when we include all three types of data (financial statement, interest rate, and stock return) in the training dataset, its prediction accuracy (0.2475) is the best (with the

‡‡ For example, the independent two-sample t-test (with equal sample sizes and variances) between the first and fifth model has a t statistic of 2.343, whereas the t-statistic of the test between the second and fourth models is 56.85.

difference of 0.1791) among all models reported in Table 3, and in fact, the improvement in accuracy is statistically significant compared to the second-best model which is the fifth model that is trained on financial statement and stock return data (0.2508).[§§]

It is noteworthy that the largest difference between the accuracy of analyst forecasts and AI predictions does not necessarily indicate that the model with three different data sets is better than the ones with one or two sets of data. The answer depends on the model's goodness of fit. A high mean difference value (i.e., improvement in accuracy) presented above can be misleading if the model's predictions are inconsistent or only accurate for specific subsets of data.

4.2 Evaluating AI model's goodness of fit

We first calculate the mean squared error (MSE) which is the average squared difference between the realized return and the predicted return:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(R_{\text{Realized}} - R_{\text{Predicted}})^2 \tag{3}$$

The first table in Table 3 Panel B reports the MSE for analyst forecasts, the MSE for AI predictions, and the difference between the two. It is indeed the last model that is trained on all three types of data (financial statement, interest rate, and stock return) in the seventh row has the lowest MSE (0.1157) among all models. The model with two types of data (financial statement and stock price) has a similar MSE (0.1200) and it is also significantly smaller than the MSE of analyst forecasts (0.3560).

We also estimate the Predicted R-squared, which is the proportion of variance in the realized stock prices that can be explained by the variance in the out-of-sample predicted

[§§] The independent two-sample t-test with equal sample sizes and variances between the fifth and seventh model has a t statistic of 5.834.

returns:

$$Predicted\ R^2 = 1 - \frac{\sum_{i=1}^{N}(R_{\text{Realized}} - R_{\text{Predicted}})^2}{\sum_{i=1}^{N}(P_{\text{Realized}} - \bar{P}_{\text{Predicted}})^2} \tag{4}$$

Surprisingly, the model trained on financial statement and stock return data has the best prediction ability, as suggested by the highest value of the Predicted R-squared (0.453) in the fifth row in the second table in Table 3 Panel B, followed by the model that includes all three types of data with a Predicted R-squared of 0.270 (the seventh row). The model that is solely trained on financial statement data has a Predicted R-squared of 0.223, which is still better than the Predicted R-squared of the analyst forecast (-1.247). It is important to note that a neural-network model trained on more features (e.g., the seventh model with financial statement, interest rate, and stock return data) can sometimes perform worse in predictions than one with fewer features in the training dataset (e.g., the fifth model with financial statement and stock return data) due to several factors, and it can be attributed to several factors, primarily overfitting and the quality of the additional features. Overfitting occurs when a model becomes too complex and starts to fit the training dataset witch has a large number of features too closely, capturing noise and outliers rather than the underlying pattern, especially if the additional features are not relevant or are highly correlated with existing features (Hawkins, 2004). This can lead to a model that performs well on the training data but poorly on unseen training data; however, it may not be the major cause because our model has already incorporated the dropout regularization technique (Goodfellow et al., 2016). The quality of the additional features is another potential cause when the new features added to the dataset (i.e., interest rate data) may not provide any new information that is useful for the model to learn from. If interest rate data are noisy or irrelevant, they can act as a source of confusion for the model, making it harder for it to identify the underlying patterns in the data (Guyon and Elisseeff, 2003). It is in fact a very likely factor given that the second model trained on interest rate data has a Predicted R-squared of only 0.06 (the second row).

Overall, these results are encouraging, in the sense that the model seems to be well trained and able to capture the dynamics of the data with out-of-sample predictions not too far from the realized returns. Nonetheless, it is crucial to examine the statistical means across different years rather than solely relying on the overall mean value of accuracy because focusing on the overall mean can mask year-specific variations and trends in prediction accuracy.

4.3. Breakdown of prediction accuracy comparison

When we examine the differences in accuracy between analyst forecasts and AI predictions at the calendar quarter level, AI predictions using only financial statement outperform human analysts forecasts in all 13 years, as shown in Columns (1)–(2) in Table 4. The improvements in forecast accuracy are more significant in more recent years (e.g., 0.23 in 2018, 0.22 in 2019, 0.30 in 2021, and 0.27 in 2022). For the model that is trained on all three types of data, the accuracy improvements of AI predictions over human forecasts follow a similar pattern, as shown in Columns (1)–(3). The absence of trend or seasonality in both cases suggests that the accuracy measure is randomly distributed across the years.

[Insert Table 4 Here]

4.4. Larger training sample using 16 prior quarters

The efficacy of neural networks in predictive modeling can be significantly influenced by the size of the training sample. While it is feasible (and of course, fast) to train our neural-network model on a short period of data (i.e., four prior quarters in our model training exercises), a more extensive historical dataset, presumably with more features, is likely to enhance prediction accuracy because it can provide the model with a more comprehensive understanding of the underlying patterns and relationships (Hansen and Salamon, 1990). However, there is a trade-off between the quantity of training data and the risk of overfitting. As the volume of data used for training increases, so does the likelihood of the model becoming overly specialized to the training data, thereby compromising its ability to generalize to new, unseen data (Qi and Zhang, 2001).

[Insert Table 5 Here]

In this section, we train the same model using financial statement, interest rate, and stock return data of 16 prior quarters to assess the robustness of our models. The two-sample paired t-tests are reported in Table 5. While all AI models trained on various combinations of historical data outperform human analysts, the only model shows some improvement in prediction accuracy over the benchmark model is the one trained on stock return data. The accuracy improvement over human analyst forecasts is 0.0652, whereas the improvement of the 4-quarter model is 0.0646 (reported in Table 3). It is important to note that the sample size using 16 prior-quarter data is reduced from 284,305 to 276,439 given the requirement that each firm in the sample much have 4-year historical data. In the rest of six models, the models using the small sample outperform the benchmark model using the large sample. It appears that the costs of overfitting outweigh the benefits of data quantity in our models. For example, the accuracy improvement (over human analysts) in the model trained on financial statement, interest rate, and stock return data of prior four quarters is 0.1791, and the improvement in the model on 16-

quarter data is 0.1769.[***]

4.5. Using financial ratios in the training sample

While the purpose of this paper is to investigate the possibility of deep-learning raw accounting information to make a prediction of future stock returns, it is also likely that certain financial ratios possess some degree of forecasting ability as well. In fact, various studies have investigated the predictive power of financial ratios. For instance, the market-to-book ratio has been shown to be a significant predictor of future stock returns, with firms exhibiting low market-to-book ratios tend to outperform those with high ratios (Fama and French, 1992). Additionally, leverage ratios, such as debt-to-assets, have been found to be negatively associated with future stock returns, as highly leveraged firms tend to be riskier and more prone to financial distress (George and Hwang, 2010). Furthermore, liquidity measures, including cash-to-assets, have been shown to positively predict future stock returns, as firms with higher cash reserves tend to be better equipped to weather financial shocks and capitalize on investment opportunities (Palazzo, 2012). Other accounting ratios, such as R&D-to-assets and capex-to-assets, have also been explored as potential predictors of future stock returns. Firms with high R&D expenditures tend to exhibit higher future stock returns, as R&D investments are often associated with innovation and growth opportunities (Chan et al., 2001). Similarly, capital expenditure intensity, as measured by capex-to-assets, has been linked to future stock performance, with firms investing heavily in capital projects tend to outperform those with lower capex intensity (Titman et al., 2004). These findings are consistent with the notion that firms investing in growth opportunities and intangible assets tend to be rewarded by the market.

[***] The independent two-sample t-test between the two improvement values has a t statistic of 2.05.

In this section, instead of deep-learning unstructured accounting numbers, we train the model using market-to-book, the natural of total assets, debt-to-assets, ROA, cash-to-assets, R&D-to-assets, and CapEx-to-assets ratios. The two-sample paired t-tests reported in Table 6 suggest that there are significant improvements in prediction accuracy when compared with human analyst forecasts; the degree of improvement, however, is much smaller than that of the previous models. For the case of using financial statement, interest rate, and stock return data, the accuracy improvement is only 0.1370 with a predicted R-squared of 0.109 (the last row of Table 6), but the improvement reported in Table 3 has a statistically significantly higher value of 0.1791 with a predicted R-squared of 0.270.

[Insert Table 6 Here]

4.6. Shallow- and deeper-learning models

The number of layers in the neural-network model and the number of times the entire dataset is fed into the model during training (i.e., the epochs) are critical factors in building a fit and efficient AI model. The impact of the numbers of network layers and epochs is particularly significant when shifting from shallow learning to deep learning. With more layers, the network becomes more complex and has the potential to memorize the training data rather than learning the underlying patterns. This could be due to the problem of vanishing gradients: as the number of layers increases, the gradients that are used to update the weights of each neuron in the neural-network model become very small. It results in slower training and can even cause the network to stop learning altogether, leading to poor performance in making predictions using the out-of-sample data. In shallow learning, however, the neural network consists of only one or two hidden layers and fewer number of epochs, making it relatively easy and fast to train. Certainly, in the early stages of training, the neural network may have a high error rate and produce inaccurate predictions. The training process continues with each epoch to further adjust weights and biases, reduce the error rate, and improve accuracy. On the other hand, as neural networks become "deeper," with multiple hidden layers, the number of epochs needed for successful training significantly increases.

One crucial factor in building a reliable and robust deep-learning model is the prevention of over-fitting when a network becomes too specialized in recognizing patterns in the training data, resulting in poor performance on out-of-sample prediction. A "shallow" neural network can often achieve good results with a small number of epochs as the network can quickly learn and adapt to the training data. In the first robustness check, we reduce the number of epochs to 2,000 and the number of layers to four (one input, two hidden, and one output layers), rather than 3,000 epochs and five layers in the previous models. In the Python code, this is specified as:

```
self.linear1 = nn.Linear(616, 4096)
self.linear2 = nn.Linear(4096, 8192)
self.linear3 = nn.Linear(8192, 2048)
self.linear4 = nn.Linear(2048, 1)
```

This 4-layer neural network is relatively simple and easy to train as it has a limited number of parameters, helping stabilize the gradients and facilitating faster and more stable training. While, the forecast accuracy of this 4-layer model with 2,000-epoch training certainly outperforms human analysts with an accuracy improvement of 0.1723 from learning only financial statement data and 0.1744 from all three types of data, the improvements are still slightly smaller than the ones of the 5-layer model with 3,000 training epochs (0.1743 and 0.1791 respectively in Table 3 Panel A). Both the MSE and predicted R-squared are similar to those in the 5-layer 3000-epoch model. For example, the MSE and R-squared are 0.1233 and 0.221

respectively in the model trained on financial statement data, whereas the MSE and R-squared are 0.1232 and 0.223 respectively in the 5-layer model (first row in Table 3 Panel A and Panel B).

[Insert Table 7 Here]

In the second robustness test, we use the same in-sample dataset to train a "deeper-learning" model that has six layers with 4,000 epochs rather than five layers in the previous exercises. In the Python code, this is specified as:

```
self.linear1 = nn.Linear(616, 4096)
self.linear2 = nn.Linear(4096, 8192)
self.linear3 = nn.Linear(8192, 8192)
self.linear4 = nn.Linear(16384, 4096)
self.linear5 = nn.Linear(4096, 2048)
self.linear6 = nn.Linear(2048, 1)
```

The forecast accuracy reported in Table 7 suggests that the model with more layers and more training epochs performs actually worse than that of the 5-layer model trained with fewer epochs. The accuracy of the model trained on accounting data is 0.2640 compared with the average accuracy of 0.2523 reported in Table 3. The error in prediction is not only larger than the 5-layer model but also the 4-layer model shown above. The MSE of the 6-layer model is 0.1338, whereas the MSE of the 4-layer model is only 0.1233. Similarly, the Predicted R-squared of the 6-layer model is 0.155 compared to that of the 4-layer model, which is 0.221.

In principle, neural networks with more layers can capture more intricate patterns within data; however, as the depth of the network increases, so does the risk of over-fitting, where the model learns noise and irrelevant patterns rather than the underlying structure of the data. In our case, the model with fewer layers with fewer training epochs outperforms a deeper counterpart when operating on limited or less complex datasets, such as the 616-variable in-sample training dataset used in this research. Thus, while deeper networks have their merits in theory, it is critical to assess model performance holistically and consider the specific context and dataset characteristics, rather than defaulting to the assumption that increased depth in AI deep learning always improves prediction accuracy. Certainly, it remains an empirical question whether simple linear statistical models or alternative nonlinear machine-learning models that are much less computationally intensive would outperform our complex deep-learning models when operating on the same in-sample training and out-of-sample prediction datasets, which we investigate in the next section.

4.7. Linear statistical model and ensemble machine learning

Numerous regression analysis techniques, including Ordinary Least Squares (OLS) and Least Absolute Shrinkage and Selection Operator (LASSO) are used to predict market outcomes by identifying linear relationships in the observed data, particularly in the form of panel data (Keane and Neal, 2020). In recent years, ensemble machine learning algorithms, such as Random Forest and Gradient Boosting that build on decision trees to make predictions, have gained significant popularity in predicting financial outcomes (Olson et al., 2021).

A simple comparison of these models' strengths and weaknesses is shown in Table 8 Panel A (Abraham et al., 2022). While OLS regression is the most fundamental statistical modeling technique that has been widely used in finance to model the relationship between stock returns and various predictor variables (Fama and French, 1993), it suffers from several limitations, mainly its inability to handle high-dimensional data and non-linear relationship (Hastie et al., 2009). In contrast, LASSO is a regularization technique that can handle high-dimensional data by selectively dropping the coefficients of irrelevant variables (Tibshirani, 1996). In fact, LASSO has been shown to be effective in selecting relevant predictor variables and improving the out-of-sample performance of stock return predictions (Chinco et al., 2019).

[Insert Table 8 Here]

Random Forest is an ensemble learning method that combines multiple decision trees to improve the accuracy and robustness of predictions (Breiman, 2001). It has been shown to be effective in handling non-linear relationships between predictor variables and financial outcomes (Khandani et al., 2010). Gradient Boosting is another ensemble learning method that sequentially combines multiple weak models to create a strong predictive model (Friedman, 2001), and has been shown to be effective in capturing complex interactions between predictor variables (Sadhwani et al., 2021). Several studies have compared the performance of these four techniques in predicting future stock returns using a large dataset of U.S. stocks and show that Gradient Boosting outperforms the other three techniques, followed closely by Random Forest. LASSO performed reasonably well, while OLS performed poorly (Gu et al., 2020).

In our experiments of in-sample OLS regressions and out-of-sample predictions, the data constantly cause larger-than-normal coefficient estimates and standard errors, as shown in the first and second rows of Table 8 Panel B. The accuracy measure of 0.5384 is much larger than the accuracy of human analyst forecasts (0.4266). When using linear regression models such as the OLS to analyze financial statements, the presence of multicollinearity can certainly lead to inaccurate and unreliable results. In accounting data, it is common for different variables to be interdependent but have a strong correlation with each other. For example, in the income statement, the revenue and cost of goods sold are highly correlated, as they are both affected by the level of sales. Similarly, on the balance sheet, assets and liabilities are interrelated, as the purchase of assets often involves taking on additional liabilities. Without creating meaningful financial ratios (as the purpose of this research is to let artificial intelligence learn by itself), the high correlation between accounting variables can lead to problems in generating regression estimates, often causing the coefficients to become unstable or have large standard errors.

One of the key advantages of neural networks over linear regression models, such as the OLS, is that they are less vulnerable to collinearity-type problems because they tend to be overparameterized, as in our research (using 629 variables; see the section of sample data for more detailed discussion). A large number of parameters are fitted using backward propagation, and the learned weights of neurons from such a repetitive training process create redundancies that make things that affect any small subset of features (such as multicollinearity) less important (Fang et al., 2022). Therefore, the ability of a network of interconnected nodes to learn and extract patterns from the predictors helps generate accurate predictions, regardless of whether the predictors are correlated or not. In some cases, data multicollinearity can even improve the performance of neural networks, as it provides additional information for the network to learn from.

In contrast to OLS, LASSO regression introduces an L1 regularization method for variable selection, wherein it can reduce the number of predictors and exclude irrelevant predictors by forcing some coefficients to zero, particularly when the predictors are highly correlated. By selectively shrinking the set of the predictive variables, LASSO can build a simpler model that retains only the most informative features and prevent the model from capturing noise in the training data. As a result, it can handle high-dimensional datasets, reduce the risk of over-fitting, and enhance predictive performance on out-of-sample data. While it excels in regularization and interpretability, it still assumes a linear relationship between predictors and outcomes, limiting its performance in capturing intricate patterns (i.e., non-linear relationships) present in the data. Additionally, LASSO regression may struggle when faced with highly correlated predictors, potentially leading to arbitrary selection of one predictor over another. In our experiment, LASSO performs well in out-of-sample prediction (third and fourth rows of Table 8 Panel B). Due to its ability to selectively shrink predictive variables, the accuracy measures are 0.2923 for both models using only accounting data and using all three types of data. It is slightly worse than the prediction accuracy of neural-network-based deep-learning model; however, the goodness of fit measures of both MSE and predictive R-squared are much worse than those of the deep-learning model. For instance, the MSE is 0.1481 and predictive R-squared is 0.065 in the model that uses accounting data, whereas these measures are 0.1232 and 0.223 respectively, in the benchmark AI model.

The main advantage of ensemble methods like Random Forest and Gradient Boosting in machine learning over LASSO regression is their ability to handle high-dimensional data with a large number of predictors and capture non-linear relationships between variables. In fact, the performance of these two machine-learning algorithms is surprisingly comparable to that of neural-network based models. In the models trained on only accounting data, the accuracy measure is 0.2771 in the Random Forest model and 0.2661 in the Gradient Boosting model. While they are still statistically significantly worse than the accuracy of neural-network deep-learning model (0.2523), it takes much less time (fewer than ten minutes) using both algorithms to compute the coefficients and make predictions in Python. For comparison, it takes more than ten hours to train the neural-network models in the same hardware equipment. Still, the neural-network model has a superior fit these two machine-learning models. For instance, the MSE of both machine-learning models is between 0.13 and 0.14 and the predicted R-squared is between 0.13 and 0.18, whereas the MSE of the benchmark AI model is about 0.12 and the R-squared is between 0.22 and 0.27.

4.8. Cross validation using k-fold

A crucial aspect of developing a reliable neural-network model is evaluating its performance. In this section, we use k-fold cross-validation to gauge the model's ability to generalize across different subsets of the data (Stone, 1974). This can help us to check the reliability of the performance metrics that we obtained in the earlier empirical exercises by gauging the model's ability to generalize across different subsets of the data. We divide the training dataset of financial statement data into five subsets (k = 5), training the model on the other four (k - 1) subsets, and testing it on the remaining one subset. This process is iterated five times, with each subset being used once as the test set. The average performance across these k iterations is then computed to obtain an overall assessment of the model's performance.

[Insert Table 9 Here]

Before comparing the prediction accuracy and the goodness of fit reported in Table 9, it is important to note that the average values of both measures vary across the folds because the sub-samples are randomly drawn from the entire training sample. The improvement in accuracy over analyst forecasts ranges from 0.1453 to 0.1681, and they are smaller than the improvement of the original deep-learning model training on the entire accounting dataset (0.1743 in the first row in Table 3 Panel A). This discrepancy can be attributed to several factors, including differences in training sample size, model instability, and the inherent bias in the cross-validation procedure. In k-fold cross-validation, each model is trained on a subset of the total data (k-1 folds), which inherently means that the training dataset for each iteration is smaller than the dataset used to train the original model (Bengio and Grandvalet, 2004). This reduction in sample size can lead to differences in model parameter estimation, as the models are exposed to different data points during training. Consequently, the prediction values generated by these models can vary, as the models may not capture the underlying data distribution equally well due to the limited training data. Secondly, neural networks are known to be sensitive to initial conditions, such as weight initialization, and the stochastic nature of some optimization algorithms used in training (Glorot and Bengio, 2010). Even when trained on the same dataset, different initializations can lead to different solutions, a phenomenon known as model variability or instability. In the context of k-fold cross-validation, this variability can be compounded by the differences in the training datasets across folds. As a result, the models trained during cross-validation may converge to different local optima, leading to a range of prediction values that differ from those obtained from the original model trained on the full dataset. Furthermore, the cross-validation procedure itself can introduce bias, particularly if the dataset is not perfectly representative of the population or if there is a significant imbalance in the data distribution across folds (Breiman, 1996). Our finding reported here may suggest that certain folds contain a disproportionate number of outliers or the distribution of the stock returns varies significantly across folds; as a result, the models trained during cross-validation may not generalize equally well to the test sets.

V. DETERMINANTS OF DEVIATIONS

The results of mean difference tests presented thus far have shown the ability of AI to analyze large amounts of financial statement data and generate reliable stock returns predictions; however, empirically, it is equally of interest to examine how and why the predictions of deep-learning models differ from the forecasts of human analysts. While both human and machine-based methods aim to minimize the difference between a predicted future price and the actual realized price, based on available information on the day of the forecast, they differ in their approach, methodology, and the nature of the output they produce (Coleman et al., 2022). In this section, we construct a firm-level panel dataset and use a set of measures of firm characteristics to conduct a series of fixed-effects regressions to investigate the factors that could potentially drive the deviation between analyst forecasts and AI predictions. Table 10 Panel A provides a list of variables, including, for example, firm size (market cap and total assets), growth potential (market-to-book), financial leverage (debt-to-asset), profitability (ROA), liquidity (cash-to-assets), and investment (R&D-to-assets, capital expenditure-to-assets), which are thought to contribute to and influence stock price returns.

[Insert Table 10 Here]

The dependent variable is the absolute value of the difference between the predicted returns of human analysts and the AI model, and we call it "Deviation":

$$Deviation_{\text{Analyst,AI}} = \left| R_{\text{Analyst}} - R_{\text{AI}} \right| \tag{5}$$

Table 10 Panel B reports the summary statistics of all variables used in the subsequent regression analysis. Because we collapse a panel dataset from the firm-, forecast-, and time-level to a firm- and time-level dataset by consolidating multiple observations per firm-quarter, which are distinguished by different analyst forecasts, into a single observation per firm-quarter, the sample size is reduced from 284,305 in the univariate test sample to 85,086 in the panel dataset. It is interesting to note that the ratio of these two sample sizes (284,305 / 85,086 = 3.34) indicates that on average each firm receives about three to four forecasts from analysts in each calendar quarter. Overall, the financial characteristics of firms that receive analyst coverage for their 12-month target price are in line with those of publicly traded companies. For example, the average values of total assets ($11 billion) and market capitalization ($10 billion) are comparable to the average asset size of $10 billion (S&P Global Market Intelligence, 2022) and the average market cap of $8 billion (Bloomberg Capitalization Data, 2022) among all public firms. Similarly, the S&P Global Market Intelligence reported that institutional investors held approximately 67% of the total shares outstanding of U.S. publicly listed companies in 2010, and it has since increased to 73% in 2020, and we find an average institutional ownership of 71% with a median value of 81%. In addition, the Pearson's correlation coefficients among all independent variables to be used in the regression analysis are shown in the lower triangle of Table 11, and the Spearman's rank correlations are shown above the diagonal in the same table. There is some degree of correlations between firm size and other variables. For instance, the Pearson's correlation is 0.448 between the logarithm of total assets and stock return volatility, -0.446 between total assets and cash, 0.410 between assts and R&D. Besides, R&D is also highly correlated with ROA (-0.532) and cash (0.695).

[Insert Table 11 Here]

5.1. Benchmark results of panel regressions

In the first cross-sectional regression test using industry (2-digit SIC code) and time (calendar quarter) fixed-effects to capture unit-invariant heterogeneity due to both industry and time. The results of deep-learning financial statement data reported in Table 12 suggest some evidence that analysts and the AI model make similar predictions for firms that have larger total assets, higher market-to-book, lower leverage, higher ROA, lower cash holding, and lower R&D (Column 1). It should be noted that both human analysts and the AI model have access to the same accounting information on the prediction date (as included in the training dataset). Therefore, the significant deviation effects driven by these firm-level characteristics are likely caused by analysts' incompetence and bias when they were studying companies with these characteristics.

[Insert Table 12 Here]

When we add the level of stock price in Column (2), return volatility, and institutional ownership to the regression specification, the effects of firm size, market-to-book, financial leverage, and ROA on forecast deviations are reduced significantly. In fact, the coefficient of the market-to-books only has a 10% significance. It is likely that the level of a firm's stock price already captures a significant amount of information about its size, growth, leverage and profitability. This is consistent with the findings in the literature that firms with higher stock prices tend to be larger in size and have higher growth potential (Dyl and Elliott, 2006), tend to have lower financial leverage (Welch, 2004), and tend to be more profitable (Fama and French, 2006). The negative sign of the stock price variable's coefficient estimate suggests that analysts and the AI model are more likely to agree on forecasted returns when the stock price is higher on the day of the forecast. This is not surprising because higher stock prices are typically accompanied by more informative signals about the firm's prospects, making it easier for analysts to make informed forecasts (Grossman and Stiglitz, 1980). In addition, stock return volatility has a significantly positive effect on prediction deviation; in other words, AI and human analysts make similar predictions for a firm with a low degree of uncertainty.

At the same time, institutional ownership is a significantly negative factor in determining forecast diversion, meaning the difference between human and machine predictions is smaller for firms that have a larger share of institutional investors. Extensive research in finance suggests that the ownership of institutional investors is closely related to analysts' forecasts of the target price (Ljungqvist et al., 2006). Analysts play a more important function for smaller firms that have less institutional ownership (Asquith et al., 2005), and analyst forecasts are more valuable (i.e., stronger market reaction) for more opaque firms that have lower institutional ownership (Loh and Stulz, 2018). On the other hand, because the training data does not contain institutional ownership data, the AI model does not have any knowledge about this information, and therefore, its prediction is more unbiased than that of human analysts, who are not necessarily under the influence and scrutiny of institutions and regulators except the 1933 Glass-Steagall Act mandates the separation of commercial and investment banking activities and the 2002 Sarbanes-Oxley Act that requires the disclosure of knowable conflicts of interest (Jennings, 2013). Out finding does suggest that for firms with lower institutional ownership, the deviation in predicted price between the analysts and the AI model may increase.

In Columns (3) and (4), we add historical interest rates to the training dataset for the AI model. In Columns (5) and (6), we add historical stock returns, rather than interest rates, to the training dataset. In Columns (7) and (8), the AI-predicted stock returns are from the model trained on all three types of data: financial statements, interest rates, and stock returns. Across all columns, the coefficient estimates are similar in sign, magnitude, and significance level.

Overall, this set of benchmark regression results suggests that the analysts and the AI model are more likely to agree with each other in terms of their future performance about larger firms that are owned by institutions and exhibit signs of higher profitability, better growth potential, lower leverage, lower liquidity, smaller R&D, and lower uncertainty. The average R-squared across all specifications is well above 0.3; however, the fact that the "between-industry" R-squared is higher than the "within-industry" R-squared in all specifications may imply that the primary drivers of variation in forecast diversion are static, industry-specific characteristics. Therefore, we need to use firm fixed-effects to examine whether these factors are related to individual firms' changes over time.

5.2. Firm-fixed effect results

In order to control for unobservable firm characteristics that usually do not change much over time, we include both the firm and time (quarterly) fixed-effects in our regression specifications to examine the "within variation" of the same firm that was studied by both the AI model and human analysts over time. Across various combinations of using financial statement, interest rate, and stock return data in the training dataset, the coefficients of total assets, market-to-book, ROA, R&D, stock price, volatility, and institutional ownership are similar in sign, magnitude, and statistical significance to that of the benchmark model using industry and time fixed-effects. The sign of cash holding variable changed from positive to negative and it is statistically significant in all specifications. In addition, the coefficient of capital expenditure became significantly positive. These two changes suggest that when a firm increases its liquidity and reduces its investment in tangible assets, analysts and the AI model are likely to agree on the forecasted return. It is interesting to note that the coefficients of debt-to-assets become negative in Columns (2), (4), (6) and (8) when stock price variable is included in the regression specifications. It indicates that both AI and human analysts make more similar predictions for firms increases financial leverage.

[Insert Table 13 Here]

The "within" R-squared is much larger than the "between" R-squared (e.g., 0.153 vs. 0.021 in Column 1), suggesting that firm-level changes in these determinant factors (e.g., size, profitability, investment, price, volatility, and institutional ownership) can capture the variation of forecast diversion between the AI model and human analysts for each firm over time. It is important to note that if we assume that AI-generated predictions are based on impartial observation and objective analysis (Coleman et al., 2022), then the deviation of analysts' forecasts from AI predictions reflects the degree of optimism bias or the need to curry favor with companies' management.

5.3. Robustness checks using alterative models

It is known that machine-based prediction models can have biases and limitations when predicting cross-sectional outcomes, and these biases and limitations often come from the data on which the models are trained, or the noisy, incomplete nature of the data (Kleinberg et al., 2018). For instance, the benchmark results in Table 13 are based on the model trained on financial statement data of four prior quarters for each firm-quarter observation, and thus it does not capture the dynamics of stock returns over an extended period.

To enabling the identification of trends and seasonality, in the first robustness check, the outcome variable of “deviation” is calculated using a model trained on 16 prior quarters of financial statement data. The use of this alternative measure, supposedly containing more information in panel regressions, does not alter coefficient estimates of all factor variables in terms of sign, magnitude, and statistical significance (Table 14). However, the "within" R-squared does increase in six out of eight columns, indicating an improvement in the model’s explanatory power of forecast deviations for the variation within individual firms over time.

[Insert Table 14 Here]

In the second set of robustness regressions, the “deviation” variable is calculated based on the AI model that is trained on financial ratios, rather than financial statement data, of the last four quarters. While the coefficient estimates of most variables reported in Table 15 are similar to those of the benchmark results, the magnitude of the R&D becomes slightly smaller, and the magnitudes of both capital expenditure and volatility variables become much larger. At the same time, the “within” R-squared decreases dramatically in all columns compared to the benchmark model, suggesting that the model trained on accounting ratios may have less predictive power. However, it is also likely that a company’s accounting ratios do not change much over time compared to its raw financial statement data, and a lower within-firm variation may result in a lower “within” R-squared, not necessarily because accounting ratios are less relevant or suitable for predicting future returns.

[Insert Table 15 Here]

We have shown in the univariant test results that a shallow-learning model with fewer neural-network layers actually outperforms the deeper-learning model with more layers, potentially due to overfitting, and here we use their predictions to construct the "deviation" variable in a set of panel regression as the third robustness test. The coefficient estimates for the regressions using the shallow-learning model predicted returns are reported in Table 16 Panel A, and those using deeper-learning model predicted returns are in Panel B. In both cases, the sign, magnitude, and statistical significance are consistent with those reported in the benchmark model (Table 13). While shallow-learning model is able to make more precise predictions than the deeper-learning model, its "within" R-squared in terms of prediction deviation between human- and human-forecasts appear to be smaller than that of the deeper-learning model in all regression specifications.

[Insert Table 16 Here]

Finally, it is essential to recognize that neural network is not the sole methodology for learning historical data and predicting future outcomes. We have shown that traditional machine-learning techniques, such as gradient boosting and random forest, also possess considerable potential in predicting future stock returns. In the forth robustness check, we calculate the "deviation" variable using the predictions from two machine-learning models and report the coefficient estimates in Table 17. Surprisingly, in both the case of Random Forest (Panel A) and that of Gradient Boosting (Panel B), the coefficient estimates of most predictive variables do not deviate significantly from those of the benchmark model, except that the firm size effect is slightly smaller and the effects of capital expenditure and volatility become higher. It is important to note, however, that the values of "within" R-squared from the machine-learning models are much smaller than those of the benchmark models, particularly in the specification with the model trained solely on financial statement data (Column 1 and 2). For instance, the values are 0.153 and 0.208 in the benchmark AI model, whereas the values are only

0.1 and 0.152 in the Random-Forest model, and 0.115 and 0.169 in the Gradient-Boosting model.

[Insert Table 17 Here]

VI. CONCLUSION

In recent years, advancements in artificial intelligence have revolutionized the field of stock market analysis. AI systems have the ability to analyze vast amounts of data at a speed and accuracy that far surpasses that of human analysts. By using machine-learning algorithms, AI can identify patterns and trends in the market that may go unnoticed by human analysts. In this research, we are able to train a simple neural-network model to deep-learn financial statement data without using pre-specified financial models or pre-defined accounting ratios, and indeed, the AI model is able to outperform human analysts in predicting 12-month-ahead returns by a large margin. Cross-sectionally, equity analysts and the AI model are more likely to agree with each other in large and mature firms that have higher market-to-book, lower leverage, more profits, lower liquidity, lower R&D-to-assets, lower return volatility, and higher institutional ownership on the day of the forecast. However, the statistical dynamics within a single firm over time are slightly different. By accounting for firm-specific effects, analysts and the AI model are more likely to disagree when the size of a firm increases and the level of its liquidity decreases while the effects of other factors such as return volatility and institutional ownership remain unchanged.

The empirical findings in this paper demonstrate that AI does have the potential to outperform stock analysts in predicting the performance of stocks. While human intuition and experience are invaluable in the world of finance, the sheer processing power and data analysis capabilities of AI can provide a significant advantage in the prediction of stock performance. As AI technology continues to evolve, it is likely that we will see an increasing number of investors turning to AI systems for their market predictions.

Perhaps the main takeaway from our data training experiment is the demonstration of how the efficacy of neural network models in deep-learning financial market data is intricately linked to the level of complexity in the data structure. A simple panel dataset including financial statements, interest rates and stock returns does not have much complex relationships between variables, and therefore, it can be more effectively processed by neural networks with fewer layers (or even simpler linear models of data mining). Conversely, models with deeper architectures might be less suitable for such datasets, potentially leading to over-fitting and hence lower prediction performance. This suggests a potential trade-off between model complexity and data simplicity.

REFERENCE

Abraham, K.G., Jarmin, R.S., Moyer, B.C., Shapiro, M.D., 2022. Big Data for Twenty-First-Century Economic Statistics. University of Chicago Press.

Asquith, P., Mikhail, M.B., Au, A.S., 2005. Information content of equity analyst reports. Journal of Financial Economics 75, 245–282.

Barbaglia, L., Manzan, S., Tosetti, E., 2023. Forecasting Loan Default in Europe with Machine Learning. Journal of Financial Econometrics 21, 569–596.

Bengio, Y., Grandvalet, Y., 2004. No Unbiased Estimator of the Variance of K-Fold Cross-Validation. Journal of Machine Learning Research 5, 1089–1105.

Beyer, A., Cohen, D.A., Lys, T.Z., Walther, B.R., 2010. The financial reporting environment: Review of the recent literature. Journal of Accounting and Economics 50, 296–343.

Bilinski, P., Lyssimachou, D., Walker, M., 2013. Target Price Accuracy: International Evidence. The Accounting Review 88, 825–851.

Birru, J., Gokkaya, S., Liu, X., Stulz, R.M., 2022. Are Analyst Short-Term Trade Ideas Valuable? The Journal of Finance 77, 1829–1875.

Bloomberg Capitalization Data, 2022. Bloomberg World Exchange Market Capitalization [WWW Document]. Bloomberg World Exchange Market Capitalization. URL https://www.bloomberg.com/quote/WCAUWRLD:US (accessed 9.26.25).

Bradshaw, M.T., Brown, L.D., Huang, K., 2013. Do sell-side analysts exhibit differential target price forecasting ability? Review of Accounting Studies 18, 930–955.

Brav, A., Lehavy, R., 2003. An Empirical Analysis of Analysts' Target Prices: Short-term Informativeness and Long-term Dynamics. The Journal of Finance 58, 1933–1967.

Breiman, L., 2001. Random Forests. Machine Learning 45, 5–32.

Breiman, L., 1996. Bagging predictors. Machine Learning 24, 123–140.

Brown, L.D., Call, A.C., Clement, M.B., Sharp, N.Y., 2016. The activities of buy-side analysts and the determinants of their stock recommendations. Journal of Accounting and Economics 62, 139–156.

Campbell, J.Y., Lo, A.W., MacKinlay, A.C., 2012. The Econometrics of Financial Markets. Princeton University Press.

Campbell, J.Y., Thompson, S.B., 2008. Predicting Excess Stock Returns Out of Sample: Can Anything Beat the Historical Average? The Review of Financial Studies 21, 1509–1531.

Cao, S., Jiang, W., Wang, J., Yang, B., 2024. From Man vs. Machine to Man + Machine: The art and AI of stock analyses. Journal of Financial Economics 160, 103910.

Chan, L.K.C., Lakonishok, J., Sougiannis, T., 2001. The Stock Market Valuation of Research and Development Expenditures. The Journal of Finance 56, 2431–2456.

Chinco, A., Clark-Joseph, A.D., Ye, M., 2019. Sparse Signals in the Cross-Section of Returns. The Journal of Finance 74, 449–492.

Cochrane, J.H., 2008. The Dog That Did Not Bark: A Defense of Return Predictability. The Review of Financial Studies 21, 1533–1575.

Coleman, B., Merkley, K., Pacelli, J., 2022. Human Versus Machine: A Comparison of Robo-Analyst and Traditional Research Analyst Investment Recommendations. The Accounting Review 97, 221–244.

De Veaux, R.D., Ungar, L.H., 1994. Multicollinearity: A tale of two nonparametric regressions, in: Cheeseman, P., Oldford, R.W. (Eds.), Selecting Models from Data. Springer, New York, NY, pp. 393–402.

Dechow, P.M., You, H., 2020. Understanding the Determinants of Analyst Target Price Implied Returns. The Accounting Review 95, 125–149.
Dong, G.N., 2024. Carbon Footprint and Market Value of Cryptocurrencies and the Real Economy, in: FinTech and Green Investment: Transforming Challenges into Opportunities. World Scientific Publishing, Singapore.
Dormann, C.F., Elith, J., Bacher, S., Buchmann, C., Carl, G., Carré, G., Marquéz, J.R.G., Gruber, B., Lafourcade, B., Leitão, P.J., Münkemüller, T., McClean, C., Osborne, P.E., Reineking, B., Schröder, B., Skidmore, A.K., Zurell, D., Lautenbach, S., 2013. Collinearity: a review of methods to deal with it and a simulation study evaluating their performance. Ecography 36, 27–46.
Dyl, E.A., Elliott, W.B., 2006. The Share Price Puzzle. The Journal of Business 79, 2045–2066.
Erel, I., Stern, L.H., Tan, C., Weisbach, M.S., 2021. Selecting Directors Using Machine Learning. The Review of Financial Studies 34, 3226–3264.
Fama, E.F., French, K.R., 2006. Profitability, investment and average returns. Journal of Financial Economics 82, 491–518.
Fama, E.F., French, K.R., 1993. Common risk factors in the returns on stocks and bonds. Journal of Financial Economics 33, 3–56.
Fama, E.F., French, K.R., 1992. The Cross-Section of Expected Stock Returns. The Journal of Finance 47, 427–465.
Fama, E.F., French, K.R., 1988. Dividend yields and expected stock returns. Journal of Financial Economics 22, 3–25.
Fang, C., Gu, Y., Zhang, W., Zhang, T., 2022. Convex Formulation of Overparameterized Deep Neural Networks. IEEE Transactions on Information Theory 68, 5340–5352.
Friedman, J.H., 2001. Greedy Function Approximation: A Gradient Boosting Machine. The Annals of Statistics 29, 1189–1232.
George, T.J., Hwang, C.-Y., 2010. A resolution of the distress risk and leverage puzzles in the cross section of stock returns. Journal of Financial Economics 96, 56–79.
Glorot, X., Bengio, Y., 2010. Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. Presented at the Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings, pp. 249–256.
Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.
Gorodnichenko, Y., Pham, T., Talavera, O., 2023. The Voice of Monetary Policy. American Economic Review 113, 548–584.
Green, J., Hand, J.R.M., Sikochi, A., 2024. The asymmetric mispricing information in analysts' target prices. Review of Accounting Studies 29, 889–915.
Grossman, S.J., Stiglitz, J.E., 1980. On the Impossibility of Informationally Efficient Markets. The American Economic Review 70, 393–408.
Gu, S., Kelly, B., Xiu, D., 2020. Empirical Asset Pricing via Machine Learning. The Review of Financial Studies 33, 2223–2273.
Guyon, I., Elisseeff, A., 2003. An introduction to variable and feature selection. J. Mach. Learn. Res. 3, 1157–1182.
Hansen, L.K., Salamon, P., 1990. Neural network ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence 12, 993–1001.
Hastie, T., Tibshirani, R., Friedman, J., 2009. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition. Springer.

Hawkins, D.M., 2004. The Problem of Overfitting. Journal of Chemical Information and Computer Sciences 44, 1–12.
Igami, M., 2020. Artificial intelligence as structural estimation: Deep Blue, Bonanza, and AlphaGo. The Econometrics Journal 23, S1–S24.
Iskhakov, F., Rust, J., Schjerning, B., 2020. Machine learning and structural econometrics: contrasts and synergies. The Econometrics Journal 23, S81–S124.
Jennings, M.M., 2013. Ethics and Financial Markets: The Role of the Analyst. Research Foundation of CFA Institute.
Joos, P., Piotroski, J.D., Srinivasan, S., 2016. Can analysts assess fundamental risk and valuation uncertainty? An empirical analysis of scenario-based value estimates. Journal of Financial Economics 121, 645–663.
Keane, M., Neal, T., 2020. Comparing deep neural network and econometric approaches to predicting the impact of climate change on agricultural yield. The Econometrics Journal 23, S59–S80.
Khachiyan, A., Thomas, A., Zhou, H., Hanson, G., Cloninger, A., Rosing, T., Khandelwal, A.K., 2022. Using Neural Networks to Predict Microspatial Economic Growth. American Economic Review: Insights 4, 491–506.
Khandani, A.E., Kim, A.J., Lo, A.W., 2010. Consumer credit-risk models via machine-learning algorithms. Journal of Banking & Finance 34, 2767–2787.
Kleinberg, J., Lakkaraju, H., Leskovec, J., Ludwig, J., Mullainathan, S., 2018. Human Decisions and Machine Predictions. The Quarterly Journal of Economics 133, 237–293.
Kotipalli, P.V., Singh, R., Wood, P., Laguna, I., Bagchi, S., 2019. AMPT-GA: automatic mixed precision floating point tuning for GPU applications, in: Proceedings of the ACM International Conference on Supercomputing, ICS '19. Association for Computing Machinery, New York, NY, USA, pp. 160–170.
Li, K.K., You, H., 2015. What is the value of sell-side analysts? Evidence from coverage initiations and terminations. Journal of Accounting and Economics 60, 141–160.
Livnat, J., Zhang, Y., 2012. Information interpretation or information discovery: which role of analysts do investors value more? Review of Accounting Studies 17, 612–641.
Ljungqvist, A., Marston, F., Wilhelm Jr., W.J., 2006. Competing for Securities Underwriting Mandates: Banking Relationships and Analyst Recommendations. The Journal of Finance 61, 301–340.
Loh, R.K., Stulz, R.M., 2018. Is Sell-Side Research More Valuable in Bad Times? The Journal of Finance 73, 959–1013.
Maliar, L., Maliar, S., Winant, P., 2021. Deep learning for solving dynamic economic models. Journal of Monetary Economics 122, 76–101.
Olson, L.M., Qi, M., Zhang, X., Zhao, X., 2021. Machine learning loss given default for corporate debt. Journal of Empirical Finance 64, 144–159.
Palazzo, B., 2012. Cash holdings, risk, and expected returns. Journal of Financial Economics 104, 162–185.
Qi, M., Zhang, G.P., 2001. An investigation of model selection criteria for neural network time series forecasting. European Journal of Operational Research 132, 666–680.
Sadhwani, A., Giesecke, K., Sirignano, J., 2021. Deep Learning for Mortgage Risk. Journal of Financial Econometrics 19, 313–368.
S&P Global Market Intelligence, 2022. S&P Global Market Intelligence [WWW Document]. S&P Global Market Intelligence. URL https://www.spglobal.com/market-intelligence/en (accessed 9.26.25).

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research 15, 1929–1958.

Stevens, E., Antiga, L., Viehmann, T., 2020. Deep Learning with PyTorch: Build, train, and tune neural networks using Python tools. Simon and Schuster.

Stone, M., 1974. Cross-Validatory Choice and Assessment of Statistical Predictions. Journal of the Royal Statistical Society: Series B (Methodological) 36, 111–133.

Tibshirani, R., 1996. Regression Shrinkage and Selection Via the Lasso. Journal of the Royal Statistical Society: Series B (Methodological) 58, 267–288.

Titman, S., Wei, K.C.J., Xie, F., 2004. Capital Investments and Stock Returns. Journal of Financial and Quantitative Analysis 39, 677–700.

Welch, I., 2004. Capital Structure and Stock Returns. Journal of Political Economy 112, 106–132.

Yang, J., Jin, H., Tang, R., Han, X., Feng, Q., Jiang, H., Zhong, S., Yin, B., Hu, X., 2024. Harnessing the Power of LLMs in Practice: A Survey on ChatGPT and Beyond. ACM Transactions on Knowledge Discovery from Data.

Figure 1. Neural network model

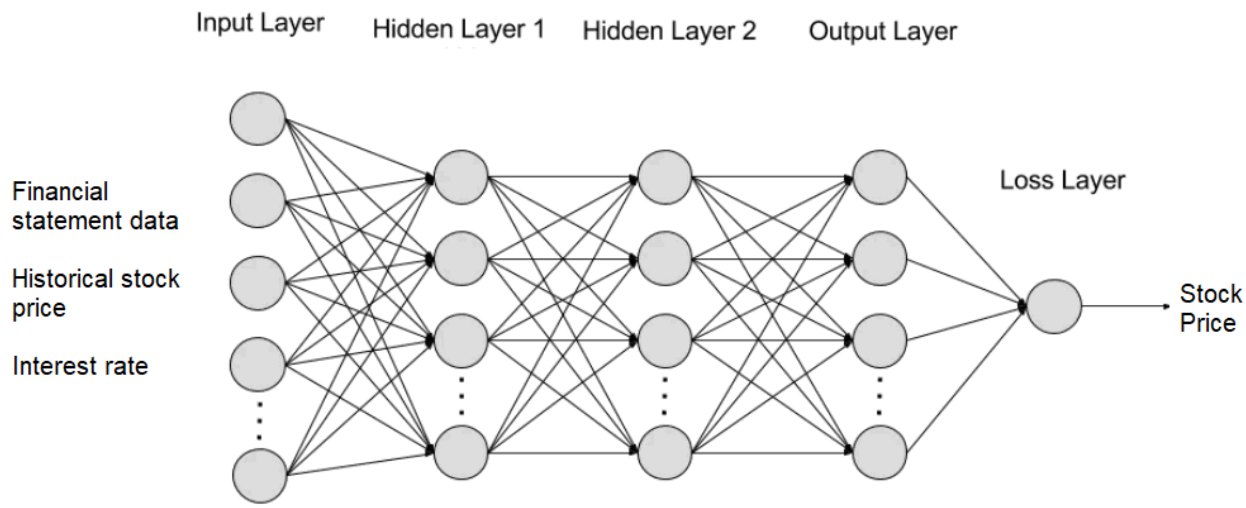

Table 1. List of software, equipment, training time, and power consumption

Software		TDP (Watts)	MSRP ($)
Operating system	Ubuntu 24.04 Linux distribution based on Debian	0	Free
Coding tool	Python 3.12	0	Free
Neural network library	Pytorch 2.4	0	Free
Statistical model library	Scikit-learn 1.6	0	Free

Hardware			
Motherboard	Gigabyte Z590 Aorus Ultra LGA1200	50W	$400
CPU	Intel Core i5-11400T 3.70 GHz 6-core 12-thread	35W	$200
DIMM memory	Samsung 64GB 2666MHz 4 × 16GB	5W	$100
Nvidia GPU	Zotac Gaming GeForce RTX 3080 Trinity OC	320W	$700
SSD Storage	Western Digital 1TB SN580 NVMe	10W	$50
Total cost of hardware		420W	$1,450

Energy consumption			
Total running time	Each training includes 3,000 epochs	15 hours	
Unit cost of electricity	Average price per kWh in Boston area (after tax)	$0.40	
Total cost of electricity		$2.52	

Table 2. List of financial statement variables used in deep-learning

Variable Name	Description
ACCHGQ	Accounting Changes - Cumulative Effect
ACOMINCQ	Accumulated Other Comprehensive Income
ACOQ	Current Assets - Other - Total
ACTQ	Current Assets - Total
ALTOQ	Other Long-term Assets
ANCQ	Non-Current Assets - Total
ANOQ	Assets Netting & Other Adjustments
AOCIDERGLQ	Accum Other Comp Inc - Derivatives Unrealized Gain/Loss
AOCIOTHERQ	Accum Other Comp Inc - Other Adjustments
AOCIPENQ	Accum Other Comp Inc - Min Pension Liab Adj
AOCISECGLQ	Accum Other Comp Inc - Unreal G/L Ret Int in Sec Assets
AOL2Q	Assets Level2
AOQ	Assets - Other - Total
APQ	Account Payable/Creditors - Trade
AQPL1Q	Assets Level1
ATQ	Assets - Total
AUL3Q	Assets Level3
CAPSQ	Capital Surplus/Share Premium Reserve
CEQQ	Common/Ordinary Equity - Total
CHEQ	Cash and Short-Term Investments
CHQ	Cash
CIBEGNIQ	Comp Inc - Beginning Net Income
CICURRQ	Comp Inc - Currency Trans Adj
CIDERGLQ	Comp Inc - Derivative Gains/Losses
CIMIIQ	Comprehensive Income - Noncontrolling Interest
CIOTHERQ	Comp Inc - Other Adj
CIPENQ	Comp Inc - Minimum Pension Adj
CIQ	Comprehensive Income - Total
CISECGLQ	Comp Inc - Securities Gains/Losses
CITOTALQ	Comprehensive Income - Parent
COGSQ	Cost of Goods Sold
CSH12Q	Common Shares Used to Calculate Earnings Per Share - 12M Moving
CSHFD12	Common Shares Used to Calc Earnings Per Share - Fully Diluted - 12M Moving
CSHFDQ	Com Shares for Diluted EPS
CSHIQ	Common Shares Issued
CSHOPQ	Total Shares Repurchased - Quarter
CSHOQ	Common Shares Outstanding
CSHPRQ	Common Shares Used to Calculate Earnings Per Share - Basic
CSTKCVQ	Carrying Value
CSTKEQ	Common Stock Equivalents - Dollar Savings
CSTKQ	Common/Ordinary Stock
DCOMQ	Deferred Compensation
DD1Q	Long-Term Debt Due in One Year
DILADQ	Dilution Adjustment
DILAVQ	Dilution Available - Excluding Extraordinary Items
DLCQ	Debt in Current Liabilities
DLTTQ	Long-Term Debt - Total
DOQ	Discontinued Operations
DPACTQ	Depreciation, Depletion and Amortization

(Table 2 continued)

Variable Name	Description
DPQ	Depreciation and Amortization - Total
DRCQ	Deferred Revenue - Current
DRLTQ	Deferred Revenue - Long-term
DVPQ	Dividends - Preferred/Preference
EPSF12	Earnings Per Share
EPSFI12	Earnings Per Share
EPSFIQ	Earnings Per Share
EPSFXQ	Earnings Per Share
EPSPI12	Earnings Per Share
EPSPIQ	Earnings Per Share
EPSPXQ	Earnings Per Share
EPSX12	Earnings Per Share
ESOPCTQ	Common ESOP Obligation - Total
ESOPNRQ	Preferred ESOP Obligation - Non-Redeemable
ESOPRQ	Preferred ESOP Obligation - Redeemable
ESOPTQ	Preferred ESOP Obligation - Total
GDWLQ	Goodwill
IBADJ12	Income Before Extra Items - Adj for Common Stock Equivalents - 12MM
IBADJQ	Income Before Extraordinary Items - Adjusted for Common Stock Equivalents
IBCOMQ	Income Before Extraordinary Items - Available for Common
IBMIIQ	Income before Extraordinary Items and Noncontrolling Interests
IBQ	Income Before Extraordinary Items
ICAPTQ	Invested Capital - Total - Quarterly
INTANOQ	Other Intangibles
INTANQ	Intangible Assets - Total
INVFGQ	Inventory - Finished Goods
INVOQ	Inventory - Other
INVRMQ	Inventory - Raw Materials
INVTQ	Inventories - Total
INVWIPQ	Inventory - Work in Process
IVLTQ	Total Long-term Investments
IVSTQ	Short-Term Investments- Total
LCOQ	Current Liabilities - Other - Total
LCTQ	Current Liabilities - Total
LLTQ	Long-Term Liabilities
LNOQ	Liabilities Netting & Other Adjustments
LOL2Q	Liabilities Level2
LOQ	Liabilities - Other
LOXDRQ	Liabilities - Other - Excluding Deferred Revenue
LQPL1Q	Liabilities Level1
LSEQ	Liabilities and Stockholders Equity - Total
LTMIBQ	Liabilities - Total and Noncontrolling Interest
LTQ	Liabilities - Total
LUL3Q	Liabilities Level3
MIBNQ	Noncontrolling Interests - Nonredeemable - Balance Sheet
MIBQ	Noncontrolling Interest - Redeemable - Balance Sheet
MIBTQ	Noncontrolling Interests - Total - Balance Sheet
MIIQ	Noncontrolling Interest - Income Account
MSAQ	Accum Other Comp Inc - Marketable Security Adjustments

(Table 2 continued)

Variable Name	Description
NIQ	Net Income
NOPIQ	Non-Operating Income
NPQ	Notes Payable
OEPF12	Earnings Per Share - Diluted - from Operations - 12MM
OEPS12	Earnings Per Share from Operations - 12 Months Moving
OEPSXQ	Earnings Per Share - Diluted - from Operations
OIADPQ	Operating Income After Depreciation - Quarterly
OIBDPQ	Operating Income Before Depreciation - Quarterly
OPEPSQ	Earnings Per Share from Operations
PIQ	Pretax Income
PNRSHOQ	Nonred Pfd Shares Outs
PPEGTQ	Property, Plant and Equipment - Total
PPENTQ	Property Plant and Equipment - Total
PRCRAQ	Repurchase Price - Average per share Quarter
PRSHOQ	Redeem Pfd Shares Outs
PSTKNQ	Preferred/Preference Stock - Nonredeemable
PSTKQ	Preferred/Preference Stock
PSTKRQ	Preferred/Preference Stock - Redeemable
RDIPAQ	In Process R&D Expense After-tax
RDIPDQ	In Process R&D Expense Diluted EPS Effect
RDIPEPSQ	In Process R&D Expense Basic EPS Effect
RDIPQ	In Process R&D
RECDQ	Receivables - Estimated Doubtful
RECTAQ	Accum Other Comp Inc - Cumulative Translation Adjustments
RECTOQ	Receivables - Current Other incl Tax Refunds
RECTQ	Receivables - Total
RECTRQ	Receivables - Trade
REQ	Retained Earnings
REUNAQ	Unadjusted Retained Earnings
REVTQ	Revenue - Total
SALEQ	Sales/Turnover
SEQOQ	Other Stockholders- Equity Adjustments
SEQQ	Stockholders Equity > Parent > Index Fundamental > Quarterly
SPIQ	Special Items
STKCOQ	Stock Compensation Expense
TEQQ	Stockholders Equity - Total
TFVAQ	Total Fair Value Assets
TFVLQ	Total Fair Value Liabilities
TSTKNQ	Treasury Stock - Number of Common Shares
TSTKQ	Treasury Stock - Total
TXDBAQ	Deferred Tax Asset - Long Term
TXDBCAQ	Current Deferred Tax Asset
TXDBCLQ	Current Deferred Tax Liability
TXDBQ	Deferred Taxes - Balance Sheet
TXDITCQ	Deferred Taxes and Investment Tax Credit
TXPQ	Income Taxes Payable
TXTQ	Income Taxes - Total
TXWQ	Excise Taxes
WCAPQ	Working Capital

(Table 2 continued)

Variable Name	Description
XACCQ	Accrued Expenses
XIDOQ	Extraordinary Items and Discontinued Operations
XINTQ	Interest and Related Expense- Total
XIQ	Extraordinary Items
XOPRQ	Operating Expense- Total
XRDQ	Research and Development Expense
XSGAQ	Selling, General and Administrative Expenses
TFVLQ	Total Fair Value Liabilities
TSTKNQ	Treasury Stock - Number of Common Shares
TSTKQ	Treasury Stock - Total
TXDBAQ	Deferred Tax Asset - Long Term
TXDBCAQ	Current Deferred Tax Asset
TXDBCLQ	Current Deferred Tax Liability
TXDBQ	Deferred Taxes - Balance Sheet
TXDITCQ	Deferred Taxes and Investment Tax Credit
TXPQ	Income Taxes Payable
TXTQ	Income Taxes - Total
TXWQ	Excise Taxes
WCAPQ	Working Capital
XACCQ	Accrued Expenses
XIDOQ	Extraordinary Items and Discontinued Operations
XINTQ	Interest and Related Expense- Total
XIQ	Extraordinary Items
XOPRQ	Operating Expense- Total
XRDQ	Research and Development Expense
XSGAQ	Selling, General and Administrative Expenses

Table 3. Univariate comparison of return prediction accuracy (benchmark models)

Panel A. Accuracy of out-of-sample prediction

This table reports the accuracy of analyst forecasts and the associated paired t-test in column (1). The accuracy is defined as the average value of the absolute differences between the realized stock return and the return based on each analyst's target price:

$$Accuracy_{\mathrm{Analyst}} = \left| \frac{P_{\mathrm{Analyst}} - P_0}{P_0} - R_{\mathrm{Realized}} \right|$$

Both the realized returns and analyst-forecasted returns are winsorized at 2.5 percent and 97.5 percent of their sample distributions. Column (2) reports the accuracy of AI predictions and the associated paired t-test. The accuracy is defined as the average value of the absolute differences between the realized return and the return based on AI-predicted price:

$$Accuracy_{\mathrm{AI}} = \left| R_{\mathrm{AI}} - R_{\mathrm{Realized}} \right|$$

Both the realized returns and AI-predicted returns are winsorized at 2.5 percent and 97.5 percent of their sample distributions. The difference between the above two accuracy measures and the associated paired t-test result are reported in the third column in the table. Standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

AI Prediction Uses Data in			N	Accuracy of Analyst Forecast	Accuracy of AI Prediction	Difference
Financial Statement	Interest Rate	Stock Return		(1)	(2)	(1) – (2)
Yes			284,305	0.4266*** (0.0008)	0.2523*** (0.0005)	0.1743*** (0.0008)
	Yes		284,305	0.4266*** (0.0008)	0.2940*** (0.0005)	0.1326*** (0.0007)
		Yes	284,305	0.4266*** (0.0008)	0.3620*** (0.0006)	0.0646*** (0.0008)
Yes	Yes		284,305	0.4266*** (0.0008)	0.2538*** (0.0005)	0.1728*** (0.0008)
Yes		Yes	284,305	0.4266*** (0.0008)	0.2508*** (0.0004)	0.1758*** (0.0008)
	Yes	Yes	284,305	0.4266*** (0.0008)	0.3422*** (0.0005)	0.0844*** (0.0008)
Yes	Yes	Yes	284,305	0.4266*** (0.0008)	0.2475*** (0.0004)	0.1791*** (0.0008)

Panel B. Goodness of fit

This reports two measures of goodness of fit in the prediction of stock price. The first measure is the Predicted Mean Squared Error (MSE) which is the average squared difference between the realized stock return and the return based on predicted price:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(R_{\text{Realized}} - R_{\text{Predicted}})^2$$

The difference between the MSE of analyst forecast and MSE of AI out-of-sample prediction are reported in the third column in the table.

AI Prediction Uses Data in						
Financial Statement	Interest Rate	Stock Price	N	$MSE_{\text{Analyst Forecast}}$	$MSE_{\text{AI out-of-sample Prediction}}$	Difference
Yes			284,305	0.3560	0.1232	0.2328
	Yes		284,305	0.3560	0.1490	0.2070
		Yes	284,305	0.3560	0.2240	0.1320
Yes	Yes		284,305	0.3560	0.1223	0.2337
Yes		Yes	284,305	0.3560	0.1200	0.2360
	Yes	Yes	284,305	0.3560	0.2000	0.1560
Yes	Yes	Yes	284,305	0.3560	0.1157	0.2403

The second measure is the Predicted R-squared which is the proportion of variance in the realized stock returns that can be explained by the variance in the out-of-sample predicted returns:

$$Predicted\ R^2 = 1 - \frac{\sum_{i=1}^{N}(R_{\text{Realized}} - R_{\text{Predicted}})^2}{\sum_{i=1}^{N}(R_{\text{Realized}} - \bar{R}_{\text{Predicted}})^2}$$

The difference between the predicted R^2 of analyst forecast and the predicted R^2 of AI out-of-sample prediction are reported in the third column in the table.

AI Prediction Uses Data in						
Financial Statement	Interest Rate	Stock Return	N	$R^2_{\text{Analyst Forecast}}$	$R^2_{\text{AI out-of-sample Prediction}}$	Difference
Yes			284,305	–1.247	0.223	–1.470
	Yes		284,305	–1.247	0.060	–1.307
		Yes	284,305	–1.247	–0.414	–0.833
Yes	Yes		284,305	–1.247	0.228	–1.475
Yes		Yes	284,305	–1.247	0.453	–1.490
	Yes	Yes	284,305	–1.247	–0.263	–0.984
Yes	Yes	Yes	284,305	–1.247	0.270	–1.517

Table 4. Breakdown of prediction accuracy comparison

This table reports the accuracy of analyst forecasts and the associated paired t-test in column (1). Column (2) reports the accuracy of AI predictions using financial statements of prior four quarters and the associated paired t-test. The difference between these two accuracy measures and the associated paired t-test statistic are reported in the column titled (1) – (2). Column (3) reports the accuracy of AI predictions using financial statements, interest rates and stock returns of prior four quarters and the associated paired t-tests. Standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

Year	N	Accuracy of Analyst Forecast	Accuracy of AI Deep-Learning Financial Statement	Difference	Accuracy of AI Deep-Learning Financial Statement + Price + Interest Rate	Difference
		(1)	(2)	(1) – (2)	(3)	(1) – (3)
2010	19,144	0.3634*** (0.0025)	0.2322*** (0.0015)	0.1312*** (0.0023)	0.2321*** (0.0015)	0.1313*** (0.0024)
2011	20,043	0.4110*** (0.0028)	0.2075*** (0.0015)	0.2035*** (0.0028)	0.2071*** (0.0015)	0.2039*** (0.0028)
2012	18,764	0.3727*** (0.0027)	0.2350*** (0.0016)	0.1377*** (0.0023)	0.2342*** (0.0016)	0.1385*** (0.0023)
2013	20,268	0.3260*** (0.0023)	0.2196*** (0.0015)	0.1064*** (0.0021)	0.2215*** (0.0015)	0.1045*** (0.0021)
2014	20,328	0.4224*** (0.0028)	0.2090*** (0.0014)	0.2134*** (0.0027)	0.2066*** (0.0013)	0.2158*** (0.0027)
2015	20,129	0.4349*** (0.0029)	0.2454*** (0.0016)	0.1895*** (0.0029)	0.2458*** (0.0016)	0.1891*** (0.0029)
2016	21,022	0.3427*** (0.0024)	0.2427*** (0.0016)	0.1000*** (0.0021)	0.2483*** (0.0016)	0.0944*** (0.0021)
2017	20,315	0.3758*** (0.0025)	0.2221*** (0.0015)	0.1537*** (0.0024)	0.2224*** (0.0025)	0.1534*** (0.0024)
2018	20,103	0.4531*** (0.0031)	0.2280*** (0.0016)	0.2251*** (0.0031)	0.2293*** (0.0031)	0.2238*** (0.0030)
2019	23,975	0.5220*** (0.0030)	0.2984*** (0.0016)	0.2236*** (0.0029)	0.3041*** (0.0016)	0.2179*** (0.0029)
2020	24,419	0.4527*** (0.0025)	0.3500*** (0.0017)	0.1027*** (0.0022)	0.3477*** (0.0017)	0.1050*** (0.0022)
2021	20,282	0.5238*** (0.0035)	0.2221*** (0.0016)	0.3017*** (0.0037)	0.2286*** (0.0016)	0.2952*** (0.0036)
2022	22,122	0.5083*** (0.0035)	0.2411*** (0.0017)	0.2672*** (0.0036)	0.2431*** (0.0017)	0.2652*** (0.0035)

Table 5. Larger training sample with 16 prior quarters

The training sample includes financial statement, interest rate and stock return data in previous four years. The sample size is reduced to 276,439 because of this 16-prior-quarter restriction in the sample. The various combinations of these three types of data are specified in the first three columns. The accuracy of analyst forecasts and the associated paired t-test in the forth column. The mean value of the accuracy is slightly different to what is reported previously because this sample is smaller than the previous one that uses only 4-prior-quarter data. The fifth column reports the accuracy of AI predictions and the associated paired t-test. The difference between these two accuracy measures and the associated paired t-test statistic are reported in the sixth column. The seventh and eighth columns are the predicted mean squared error and R-squared respectively. Standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

AI Prediction Uses Data in			Accuracy of Analyst Forecast	Accuracy of AI Prediction	Difference	MSE	R^2
Financial Statement	Interest Rate	Stock Return					
Yes			0.4149*** (0.0008)	0.2466*** (0.0004)	0.1683*** (0.0008)	0.1157	0.246
	Yes		0.4149*** (0.0008)	0.2904*** (0.0005)	0.1245*** (0.0007)	0.1457	0.050
		Yes	0.4149*** (0.0008)	0.3497*** (0.0006)	0.0652*** (0.0008)	0.2077	0.354
Yes	Yes		0.4149*** (0.0008)	0.2571*** (0.0005)	0.1578*** (0.0008)	0.1323	0.138
Yes		Yes	0.4149*** (0.0008)	0.2423*** (0.0004)	0.1726*** (0.0008)	0.1117	0.272
	Yes	Yes	0.4149*** (0.0008)	0.3405*** (0.0005)	0.0744*** (0.0008)	0.1964	0.280
Yes	Yes	Yes	0.4149*** (0.0008)	0.2380*** (0.0004)	0.1769*** (0.0007)	0.1078	0.298

Table 6. Financial ratios as training sample

The training sample includes financial ratio data in previous four quarters. These ratios include market-to-book, the natural logarithm of total assets, debt-to-assets, market-to-book, return-on-assets, cash-to-assets, R&D-to-assets, and CapEx-to-assets. The sample size is 264,626 due to the availability of certain accounting values. The various combinations of these three types of data are specified in the first three columns. Three scenarios (interest rate, stock return, and the combination of these two) are dropped because they are irrelevant to the substitution of financial ratios for accounting statements. The accuracy of analyst forecasts and the associated paired t-test in the forth column. The mean value of the accuracy is slightly different to what is reported previously because this sample is smaller than the previous one that uses only 4-prior-quarter data. The fifth column reports the accuracy of AI predictions and the associated paired t-test. The difference between these two accuracy measures and the associated paired t-test statistic are reported in the sixth column. The seventh and eighth columns are the predicted mean squared error and R-squared respectively. Standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

AI Prediction Uses Data in			Accuracy of Analyst Forecast	Accuracy of AI Prediction	Difference	MSE	R^2
Financial Ratio	Interest Rate	Stock Return					
Yes			0.4190*** (0.0008)	0.2590*** (0.0005)	0.1600*** (0.0008)	0.1231	0.211
Yes	Yes		0.4190*** (0.0008)	0.2773*** (0.0005)	0.1417*** (0.0008)	0.1357	0.130
Yes		Yes	0.4190*** (0.0008)	0.2732*** (0.0005)	0.1458*** (0.0008)	0.1333	0.145
Yes	Yes	Yes	0.4190*** (0.0008)	0.2820*** (0.0005)	0.1370*** (0.0008)	0.1390	0.109

Table 7. Shallow- and Deeper-learning models

The type of model is specified in the first column. The shallow-learning model uses a 4-layer neural-network and 2,000 epochs for training (rather than 5-layer and 3,000 epochs), and the deeper-learning model uses 6-layer and 4,000 epochs. The accuracy of analyst forecasts and the associated paired t-test in the second column. The third column specifies the type of training data, and it can be either financial statement data only, or the combination of financial statement, interest rate, and stock return data of prior four quarters. The forth column reports the accuracy of AI predictions and the associated paired t-test. The difference between these two accuracy measures and the associated paired t-test statistic are reported in the fifth column. The sixth and seventh columns are the predicted mean squared error and R-squared respectively. Standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

Model	Accuracy of Analyst Forecast	Training Data	Accuracy of AI Prediction	Difference	MSE	R^2
Shallow-learning (3-layer, 2,000 epochs)	0.4266*** (0.0008)	Financial Statement	0.2543*** (0.0005)	0.1723*** (0.0008)	0.1233	0.221
		Financial, Interest Rate, Stock Return	0.2521*** (0.0004)	0.1744*** (0.0008)	0.1203	0.240
Deeper-learning (6-layer, 4,000 epochs)	0.4266*** (0.0008)	Financial Statement	0.2640*** (0.0005)	0.1626*** (0.0008)	0.1338	0.155
		Financial, Interest Rate, Stock Return	0.2724*** (0.0005)	0.1541*** (0.0008)	0.1409	0.111

Table 8. Linear regression and machine learning

Panel A. Comparison of predictive models

Feature	OLS	LASSO	Random Forest	Gradient Boosting	Neural Network
Interpretability	Very High	High	Moderate	Low	Very Low
Complexity	Very Low	Low	High	Very High	Very High
Feature Selection	None	Moderate	High	High	High
Data Requirement	Low	Moderate	High	High	Very High
Non-linearity Handling	Low	Low	High	High	Very High
Over-fitting Resistance	Moderate	Moderate	Moderate	Moderate	Low
Computation Cost	Low	Moderate	High	High	Very High

Panel B. OLS and machine-learning models

The type of model is specified in the first column. The accuracy of analyst forecasts and the associated paired t-test in the second column. The third column specifies the type of training data, and it can be either financial statement data only, or the combination of financial statement, interest rate, and stock return data of prior four quarters. The forth column reports the accuracy of OLS and machine-learning based predictions and the associated paired t-test. The difference between these two accuracy measures and the associated paired t-test statistic are reported in the fifth column. The sixth and seventh columns are the predicted mean squared error and R-squared respectively. Standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

Model	Accuracy of Analyst Forecast	Training Data	Accuracy of OLS and Machine-learning	Difference	MSE	R^2
OLS	0.4266*** (0.0008)	Financial Statement	0.5384*** (0.0008)	−0.1118*** (0.0011)	0.4949	−2.124
		Financial, Interest Rate, Stock Return	0.5412*** (0.0009)	−0.1146*** (0.0011)	0.4994	−2.152
Lasso	0.4266*** (0.0008)	Financial Statement	0.2923*** (0.0005)	0.1343*** (0.0007)	0.1481	0.065
		Financial, Interest Rate, Stock Return	0.2923*** (0.0005)	0.1343*** (0.0007)	0.1481	0.065
Random Forest	0.4266*** (0.0008)	Financial Statement	0.2771*** (0.0005)	0.1495*** (0.0007)	0.1360	0.142
		Financial, Interest Rate, Stock Return	0.2800*** (0.0005)	0.1466*** (0.0007)	0.1384	0.126
Gradient Boosting	0.4266*** (0.0008)	Financial Statement	0.2661*** (0.0004)	0.1604*** (0.0008)	0.1297	0.181
		Financial, Interest Rate, Stock Return	0.2681*** (0.0005)	0.1585*** (0.0008)	0.1303	0.178

Table 9. K-fold cross validation

The training sample is randomly split into five sub-samples or folds (i.e., k-fold with k=5). Among them, four sub-samples are used to train the neural-network model, and the fifth sub-sample is used for prediction. The training data includes only financial statements. The fold number is specified in the first column. The accuracy of analyst forecasts and the associated paired t-test in the second column. The third column reports the accuracy of AI predictions and the associated paired t-test. The difference between these two accuracy measures and the associated paired t-test statistic are reported in the forth column. The fifth and sixth columns are the predicted mean squared error and R-squared respectively. Standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

Fold	N	Accuracy of Analyst Forecast	Accuracy of AI Prediction	Difference	MSE	R^2
1	56,020	0.4220*** (0.0017)	0.2574*** (0.0010)	0.1646*** (0.0017)	0.1225	0.199
2	56,039	0.4332*** (0.0018)	0.2651*** (0.0010)	0.1681*** (0.0017)	0.1304	0.204
3	56,039	0.4246*** (0.0018)	0.2793*** (0.0011)	0.1453*** (0.0018)	0.1468	0.034
4	55,954	0.4233*** (0.0017)	0.2556*** (0.0010)	0.1677*** (0.0017)	0.1205	0.223
5	55,943	0.4279*** (0.0018)	0.2657*** (0.0010)	0.1622*** (0.0017)	0.1307	0.206

Table 10. Variable definition and summary statistics
Panel A. Definition of variables used in the regression analysis

Variable	Definition and data source	Source
Market Capitalization (million $)	Average stock price × Average shares outstanding in a quarter	CRSP
Market to book	Market Capitalization / Stockholders Equity	Compustat
Total Assets (million $)	Total assets at the end of each quarter	Compustat 10Q, 10K
Debt to asset	(Short-term debt + Long-term debt) / Total assets	
Return on assets	Net income (loss) / Total assets	
Cash to assets	Cash and short-term investments / Total assets	
R&D expenses to assets	Research and development expense / Total assets	
Capital Expenditures to assets	(Capital expenditures – Sale of Property) /Total assets	
Stock price on on the day of the forecast ($)	Stock price in the quarter when an analyst sets the target price	CRSP IBES Detail
Stock return daily volatility (90-day prior)	Stock return volatility estimated during the 90-day period prior to the date when an analyst sets the 12-month target price	
Realized stock return	12-month stock return from the time of an analyst' forecast	
Analyst forecasted return	12-month forecasted return using the ratio of realized stock price and analyst forecasted price $R_{Forecast} = \frac{P_{Forecast} - P_0}{P_0}$	
Institutional investor ownership	Total shares owned by institutional investors/Total shares outstanding	Thomson Reuters S34
AI predicted return	AI predicted stock return (12-month period from the time of an analyst' forecast)	Deep Learning
Deviation between analyst and AI predictions	$Deviation_{\text{Analyst,AI}} = \left\| R_{\text{Analyst}} - R_{\text{AI}} \right\|$	

Panel B. Summary statistics of variables used in the regression analysis

Variable	N	Mean	Std. Dev.	Med	Min	Max
Total Assets (million $)	85,056	11,459	24,190	2,336	50.3	121401
Market Capitalization (million $)	85,056	10,157	20,454	2,336	59.1	102681.0
Market to book	85,056	3.988	4.546	2.428	0.506	22.96
Debt to asset	85,056	0.258	0.192	0.248	0	0.680
Profit margin	85,056	-0.326	1.637	0.0513	-9.194	0.455
Return on assets	85,056	0.0006	0.0409	0.0088	-0.149	0.0637
Cash to assets	85,056	0.191	0.214	0.105	0.0025	0.830
R&D expenses to assets	85,056	0.0112	0.0213	0	0	0.0921
Capital Expenditures to assets	85,056	0.0253	0.0311	0.0138	0	0.139
Stock price on the day of the forecast ($)	85,056	49.75	52.63	32.34	2.79	242.6
Daily return volatility (90-day prior)	85,056	0.0268	0.0144	0.0229	0.0091	0.0702
Institutional ownership	85,056	0.709	0.284	0.808	0	0.990
Realized stock return	85,056	0.077	0.403	0.0514	-0.697	1.217
Analyst forecasted return	85,056	0.303	0.414	0.177	-0.110	2.005
AI predicted return (using financial statements, stock price and interest rate)	85,056	0.0192	0.219	-0.00135	-0.399	0.519
Deviation between analyst and AI prediction (using financial statements, stock price and interest rate)	85,056	0.375	0.430	0.245	0.00002	2.405
Deviation between analyst and AI prediction (using financial statements and stock price)	85,056	0.391	0.447	0.252	0.00002	2.546
Deviation between analyst and AI prediction (using financial statements and interest rate)	85,056	0.387	0.442	0.250	0.00000	2.564
Deviation between analyst and AI prediction (using financial statements only)	85,056	0.386	0.443	0.249	0.00000	2.547

Table 11. Correlation matrix

The Pearson's correlation coefficients are shown in the lower triangle, and the Spearman's rank correlations are shown above the diagonal.

	Log(Total assets)	Market to book	Debt to asset	Return on assets	Cash to assets	R&D to assets	CapEx to assets	Log(Stock price)	Return volatility	Institutional ownership
Log(Total assets)		-0.081	0.329	0.220	-0.389	-0.349	0.014	0.437	-0.483	0.183
Market to book	-0.082		0.025	0.210	0.293	0.317	0.057	0.360	-0.092	0.127
Debt to asset	0.284	0.145		-0.070	-0.425	-0.287	0.073	0.048	-0.062	0.069
Return on assets	0.356	-0.030	0.022		-0.098	-0.202	0.115	0.324	-0.375	0.141
Cash to assets	-0.446	0.254	-0.367	-0.409		0.568	-0.177	-0.086	0.277	-0.056
R&D to assets	-0.410	0.263	-0.261	-0.532	0.695		-0.102	-0.055	0.252	-0.060
CapEx to assets	-0.012	-0.029	0.058	0.060	-0.174	-0.129		0.028	-0.003	0.042
Log(Stock price)	0.443	0.251	0.030	0.292	-0.108	-0.125	-0.023		-0.404	0.270
Return volatility	-0.448	0.039	-0.018	-0.461	0.322	0.324	0.028	-0.401		-0.145
Institutional ownership	0.267	0.073	0.045	0.227	-0.091	-0.108	-0.014	0.307	-0.236	

Table 12. Determinants of prediction deviation with industry and time fixed effects

The dependent variable is the deviation between analyst and AI predictions. The independent variables include the natural logarithm of total assets, market capitalization to book value of equity, debt to total assets, return on assets (ROA), cash to total assets, R&D expenses to total assets, capital expenditures to total assets, the natural logarithm of stock price on the day of the forecast, and the daily stock return volatility 90-day prior to the day of the forecast. All specifications use OLS regressions with time (quarterly) and industry (2-digit SIC code) fixed-effects. Robust standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

Financial Stmt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Interest Rate			Yes	Yes			Yes	Yes
Stock Price					Yes	Yes	Yes	Yes
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Log(Total assets)	-0.0482***	-0.0119**	-0.0475***	-0.0120**	-0.0500***	-0.0130**	-0.0482***	-0.0122**
	(0.00886)	(0.00543)	(0.00869)	(0.00526)	(0.00888)	(0.00526)	(0.00887)	(0.00535)
Market to book	-0.00941***	-0.00349*	-0.00904***	-0.00330*	-0.00979***	-0.00379*	-0.00939***	-0.00357*
	(0.00232)	(0.00198)	(0.00222)	(0.00188)	(0.00239)	(0.00201)	(0.00231)	(0.00194)
Debt to asset	0.133***	0.0369**	0.130***	0.0365**	0.135***	0.0374**	0.133***	0.0380**
	(0.0182)	(0.0169)	(0.0183)	(0.0163)	(0.0200)	(0.0180)	(0.0194)	(0.0173)
Return on assets	-2.681***	-1.807***	-2.612***	-1.753***	-2.722***	-1.827***	-2.591***	-1.719***
	(0.163)	(0.147)	(0.155)	(0.136)	(0.162)	(0.142)	(0.149)	(0.130)
Cash to assets	0.121***	0.118***	0.114***	0.112***	0.119***	0.116***	0.109***	0.106***
	(0.0244)	(0.0247)	(0.0240)	(0.0231)	(0.0248)	(0.0246)	(0.0244)	(0.0224)
R&D to assets	1.627***	1.602***	1.629***	1.604***	1.599***	1.571***	1.586***	1.560***
	(0.409)	(0.516)	(0.390)	(0.496)	(0.401)	(0.507)	(0.401)	(0.504)
CapEx to assets	0.0584	0.0612	0.0576	0.0577	0.0768	0.0764	0.0800	0.0802
	(0.110)	(0.127)	(0.111)	(0.129)	(0.113)	(0.130)	(0.112)	(0.130)
Log(Stock price)		-0.0603***		-0.0579***		-0.0607***		-0.0587***
		(0.00587)		(0.00578)		(0.00623)		(0.00605)
Return volatility		4.784***		4.763***		4.985***		4.825***
		(0.372)		(0.384)		(0.395)		(0.374)
Institutional ownership		-0.206***		-0.203***		-0.207***		-0.206***
		(0.0288)		(0.0289)		(0.0290)		(0.0289)
Constant	0.620***	0.538***	0.608***	0.524***	0.634***	0.544***	0.621***	0.535***
	(0.0663)	(0.0543)	(0.0650)	(0.0523)	(0.0667)	(0.0535)	(0.0673)	(0.0549)
Industry FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Time FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
N	85,086	85,086	85,086	85,086	85,086	85,086	85,086	85,086
R-squared (within industry)	0.275	0.330	0.284	0.338	0.277	0.334	0.245	0.304
R-squared (between)	0.655	0.809	0.674	0.795	0.650	0.805	0.648	0.795
R-squared (overall)	0.329	0.385	0.335	0.390	0.328	0.385	0.298	0.356

Table 13. Determinants of prediction deviation with firm and time fixed effects (benchmark)

The dependent variable is the deviation between analyst and AI predictions. The independent variables include the natural logarithm of total assets, market capitalization to book value of equity, debt to total assets, return on assets (ROA), cash to total assets, R&D expenses to total assets, capital expenditures to total assets, the natural logarithm of stock price on the day of the forecast, and the daily stock return volatility 90-day prior to the day of the forecast. All specifications use OLS regressions with time (quarterly) and firm (gvkey) fixed-effects. Robust standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

Financial Stmt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Interest Rate			Yes	Yes			Yes	Yes
Stock Price					Yes	Yes	Yes	Yes
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Log(Total assets)	0.0282***	0.126***	0.0286***	0.124***	0.0305***	0.132***	0.0274***	0.124***
	(0.00784)	(0.00905)	(0.00779)	(0.00905)	(0.00788)	(0.00912)	(0.00778)	(0.00902)
Market to book	-0.00332***	0.00799***	-0.00316***	0.00776***	-0.00395***	0.00769***	-0.00370***	0.00737***
	(0.000639)	(0.000742)	(0.000638)	(0.000744)	(0.000656)	(0.000756)	(0.000643)	(0.000745)
Debt to asset	0.130***	-0.0918***	0.127***	-0.0879***	0.140***	-0.0886***	0.135***	-0.0833***
	(0.0229)	(0.0234)	(0.0227)	(0.0234)	(0.0233)	(0.0239)	(0.0228)	(0.0234)
Return on assets	-0.615***	-0.0699	-0.584***	-0.0551	-0.645***	-0.0825	-0.565***	-0.0272
	(0.0646)	(0.0654)	(0.0646)	(0.0652)	(0.0659)	(0.0669)	(0.0643)	(0.0653)
Cash to assets	-0.0977***	-0.00248	-0.105***	-0.0131	-0.0970***	0.00105	-0.102***	-0.00803
	(0.0226)	(0.0215)	(0.0223)	(0.0214)	(0.0230)	(0.0220)	(0.0224)	(0.0215)
R&D to assets	1.043***	1.154***	1.118***	1.224***	1.077***	1.190***	1.129***	1.235***
	(0.305)	(0.303)	(0.307)	(0.306)	(0.309)	(0.309)	(0.305)	(0.305)
CapEx to assets	0.180**	0.452***	0.167**	0.430***	0.207***	0.486***	0.197***	0.464***
	(0.0708)	(0.0709)	(0.0701)	(0.0706)	(0.0715)	(0.0720)	(0.0701)	(0.0707)
Log(Stock price)		-0.159***		-0.153***		-0.163***		-0.155***
		(0.00630)		(0.00631)		(0.00647)		(0.00632)
Return volatility		0.999***		1.029***		1.078***		1.110***
		(0.208)		(0.208)		(0.211)		(0.207)
Institutional ownership		-0.0605***		-0.0634***		-0.0630***		-0.0639***
		(0.0128)		(0.0128)		(0.0130)		(0.0128)
Constant	0.120**	-0.142**	0.110*	-0.144**	0.102*	-0.169***	0.123**	-0.137**
	(0.0576)	(0.0579)	(0.0573)	(0.0578)	(0.0580)	(0.0585)	(0.0573)	(0.0578)
Firm FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Time FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
N	85,086	85,086	85,086	85,086	85,086	85,086	85,086	85,086
R-squared (within firm)	0.153	0.208	0.185	0.235	0.170	0.226	0.113	0.169
R-squared (between)	0.021	0.000	0.020	0.000	0.002	0.003	0.001	0.003
R-squared (overall)	0.044	0.018	0.057	0.026	0.038	0.016	0.019	0.008

Table 14. Determinants of prediction deviation using AI learning 16-prior-quarter data

The dependent variable is the deviation between analyst and AI predictions. The independent variables include the natural logarithm of total assets, market capitalization to book value of equity, debt to total assets, return on assets (ROA), cash to total assets, R&D expenses to total assets, capital expenditures to total assets, the natural logarithm of stock price on the day of the forecast, and the daily stock return volatility 90-day prior to the day of the forecast. All specifications use OLS regressions with time (quarterly) and firm (gvkey) fixed-effects. Robust standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

Financial Stmt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Interest Rate			Yes	Yes			Yes	Yes
Stock Price					Yes	Yes	Yes	Yes
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Log(Total assets)	0.0317***	0.122***	0.0336***	0.122***	0.0342***	0.124***	0.0331***	0.122***
	(0.00769)	(0.00885)	(0.00763)	(0.00878)	(0.00769)	(0.00887)	(0.00762)	(0.00882)
Market to book	-0.00256***	0.00770***	-0.00254***	0.00753***	-0.00282***	0.00744***	-0.00280***	0.00732***
	(0.000621)	(0.000726)	(0.000630)	(0.000737)	(0.000621)	(0.000726)	(0.000619)	(0.000726)
Debt to asset	0.111***	-0.0894***	0.112***	-0.0866***	0.120***	-0.0815***	0.113***	-0.0855***
	(0.0224)	(0.0230)	(0.0224)	(0.0229)	(0.0225)	(0.0231)	(0.0223)	(0.0230)
Return on assets	-0.491***	0.0204	-0.526***	-0.0182	-0.488***	0.0256	-0.498***	0.00754
	(0.0643)	(0.0650)	(0.0639)	(0.0643)	(0.0644)	(0.0651)	(0.0642)	(0.0648)
Cash to assets	-0.0802***	0.00257	-0.0854***	-0.00407	-0.0824***	0.000193	-0.0844***	-0.00255
	(0.0216)	(0.0206)	(0.0214)	(0.0204)	(0.0215)	(0.0205)	(0.0214)	(0.0205)
R&D to assets	0.958***	1.075***	0.929***	1.043***	1.010***	1.127***	1.049***	1.164***
	(0.292)	(0.292)	(0.289)	(0.288)	(0.293)	(0.293)	(0.294)	(0.295)
CapEx to assets	0.166**	0.412***	0.148**	0.390***	0.199***	0.444***	0.183***	0.426***
	(0.0686)	(0.0685)	(0.0686)	(0.0682)	(0.0687)	(0.0689)	(0.0688)	(0.0691)
Log(Stock price)		-0.143***		-0.140***		-0.143***		-0.141***
		(0.00612)		(0.00611)		(0.00613)		(0.00608)
Return volatility		1.027***		1.182***		1.120***		1.035***
		(0.207)		(0.205)		(0.206)		(0.204)
Institutional ownership		-0.0676***		-0.0650***		-0.0648***		-0.0693***
		(0.0129)		(0.0129)		(0.0130)		(0.0128)
Constant	0.0671	-0.177***	0.0593	-0.187***	0.0551	-0.193***	0.0630	-0.178***
	(0.0570)	(0.0568)	(0.0566)	(0.0565)	(0.0571)	(0.0569)	(0.0567)	(0.0568)
Firm FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Time FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
N	82,104	82,104	82,104	82,104	82,104	82,104	82,104	82,104
R-squared (within firm)	0.202	0.248	0.194	0.239	0.133	0.182	0.121	0.170
R-squared (between)	0.007	0.010	0.001	0.005	0.009	0.010	0.007	0.009
R-squared (overall)	0.044	0.023	0.042	0.021	0.014	0.007	0.013	0.007

Table 15. Determinants of prediction deviation using AI learning financial ratios

The dependent variable is the deviation between analyst and AI predictions. The independent variables include the natural logarithm of total assets, market capitalization to book value of equity, debt to total assets, return on assets (ROA), cash to total assets, R&D expenses to total assets, capital expenditures to total assets, the natural logarithm of stock price on the day of the forecast, and the daily stock return volatility 90-day prior to the day of the forecast. All specifications use OLS regressions with time (quarterly) and firm (gvkey) fixed-effects. Robust standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

Financial Stmt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Interest Rate			Yes	Yes			Yes	Yes
Stock Price					Yes	Yes	Yes	Yes
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Log(Total assets)	0.0379*** (0.00732)	0.121*** (0.00873)	0.0318*** (0.00730)	0.113*** (0.00886)	0.0371*** (0.00725)	0.125*** (0.00874)	0.0347*** (0.00733)	0.118*** (0.00889)
Market to book	-0.00320*** (0.000748)	0.00844*** (0.000894)	-0.00318*** (0.000715)	0.00810*** (0.000868)	-0.00370*** (0.000728)	0.00861*** (0.000867)	-0.00322*** (0.000707)	0.00833*** (0.000852)
Debt to asset	0.101*** (0.0227)	-0.0933*** (0.0237)	0.116*** (0.0222)	-0.0711*** (0.0234)	0.116*** (0.0225)	-0.0923*** (0.0235)	0.121*** (0.0221)	-0.0745*** (0.0233)
Return on assets	-0.809*** (0.0709)	-0.311*** (0.0708)	-0.755*** (0.0692)	-0.282*** (0.0693)	-0.780*** (0.0691)	-0.246*** (0.0689)	-0.730*** (0.0685)	-0.227*** (0.0685)
Cash to assets	-0.0654*** (0.0240)	0.00861 (0.0231)	-0.0616*** (0.0232)	0.0105 (0.0226)	-0.0681*** (0.0232)	0.0105 (0.0222)	-0.0714*** (0.0226)	0.00265 (0.0218)
R&D to assets	0.836*** (0.304)	0.812*** (0.303)	0.958*** (0.308)	0.936*** (0.306)	0.736*** (0.286)	0.710** (0.282)	0.806*** (0.287)	0.781*** (0.283)
CapEx to assets	0.581*** (0.0721)	0.801*** (0.0732)	0.517*** (0.0709)	0.732*** (0.0720)	0.484*** (0.0714)	0.718*** (0.0714)	0.401*** (0.0696)	0.621*** (0.0703)
Log(Stock price)		-0.132*** (0.00641)		-0.129*** (0.00648)		-0.139*** (0.00629)		-0.130*** (0.00629)
Return volatility		1.272*** (0.207)		0.936*** (0.207)		1.570*** (0.204)		1.496*** (0.202)
Institutional ownership		-0.0560*** (0.0136)		-0.0638*** (0.0135)		-0.0623*** (0.0138)		-0.0637*** (0.0138)
Constant	0.0242 (0.0547)	-0.185*** (0.0559)	0.0413 (0.0548)	-0.151*** (0.0565)	0.0422 (0.0541)	-0.185*** (0.0554)	0.0384 (0.0549)	-0.175*** (0.0563)
Firm FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Time FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
N	78,191	78,191	78,191	78,191	78,191	78,191	78,191	78,191
R-squared (within firm)	0.072	0.109	0.087	0.122	0.080	0.125	0.091	0.132
R-squared (between)	0.006	0.006	0.000	0.003	0.009	0.005	0.008	0.005
R-squared (overall)	0.002	0.001	0.012	0.004	0.003	0.002	0.006	0.004

Table 16. Determinants of prediction deviation using AI shallow- and deeper-learning

Panel B. Shallow-learning models

The dependent variable is the deviation between analyst and AI predictions. The independent variables include the natural logarithm of total assets, market capitalization to book value of equity, debt to total assets, return on assets (ROA), cash to total assets, R&D expenses to total assets, capital expenditures to total assets, the natural logarithm of stock price on the day of the forecast, and the daily stock return volatility 90-day prior to the day of the forecast. All specifications use OLS regressions with time (quarterly) and firm (gvkey) fixed-effects. Robust standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

Financial Stmt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Interest Rate			Yes	Yes			Yes	Yes
Stock Price					Yes	Yes	Yes	Yes
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Log(Total assets)	0.0256***	0.125***	0.0252***	0.120***	0.0289***	0.127***	0.0291***	0.126***
	(0.00789)	(0.00911)	(0.00783)	(0.00908)	(0.00780)	(0.00904)	(0.00779)	(0.00904)
Market to book	-0.00340***	0.00804***	-0.00327***	0.00766***	-0.00392***	0.00733***	-0.00354***	0.00757***
	(0.000642)	(0.000743)	(0.000639)	(0.000744)	(0.000649)	(0.000746)	(0.000645)	(0.000743)
Debt to asset	0.139***	-0.0843***	0.131***	-0.0838***	0.130***	-0.0923***	0.134***	-0.0860***
	(0.0230)	(0.0237)	(0.0229)	(0.0236)	(0.0230)	(0.0236)	(0.0230)	(0.0236)
Return on assets	-0.631***	-0.0836	-0.607***	-0.0794	-0.628***	-0.0797	-0.606***	-0.0649
	(0.0663)	(0.0671)	(0.0658)	(0.0669)	(0.0659)	(0.0670)	(0.0653)	(0.0663)
Cash to assets	-0.103***	-0.00696	-0.0981***	-0.00563	-0.0922***	0.00270	-0.107***	-0.0131
	(0.0229)	(0.0219)	(0.0226)	(0.0218)	(0.0226)	(0.0216)	(0.0226)	(0.0217)
R&D to assets	1.087***	1.199***	1.058***	1.165***	1.007***	1.115***	1.171***	1.278***
	(0.308)	(0.307)	(0.309)	(0.309)	(0.306)	(0.307)	(0.306)	(0.307)
CapEx to assets	0.170**	0.444***	0.166**	0.429***	0.228***	0.498***	0.183***	0.451***
	(0.0717)	(0.0723)	(0.0716)	(0.0722)	(0.0709)	(0.0717)	(0.0706)	(0.0714)
Log(Stock price)		-0.161***		-0.153***		-0.158***		-0.156***
		(0.00639)		(0.00632)		(0.00632)		(0.00629)
Return volatility		0.892***		0.970***		1.193***		1.118***
		(0.210)		(0.209)		(0.210)		(0.210)
Institutional ownership		-0.0600***		-0.0658***		-0.0599***		-0.0674***
		(0.0130)		(0.0129)		(0.0128)		(0.0130)
Constant	0.150**	-0.111*	0.152***	-0.0994*	0.122**	-0.145**	0.121**	-0.139**
	(0.0581)	(0.0583)	(0.0576)	(0.0581)	(0.0574)	(0.0578)	(0.0575)	(0.0580)
Firm FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Time FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
N	85,086	85,086	85,086	85,086	85,086	85,086	85,086	85,086
R-squared (within firm)	0.139	0.194	0.130	0.182	0.130	0.184	0.126	0.180
R-squared (between)	0.007	0.002	0.006	0.002	0.002	0.003	0.002	0.004
R-squared (overall)	0.033	0.012	0.031	0.011	0.025	0.010	0.021	0.008

Panel B. Deeper-learning models

The dependent variable is the deviation between analyst and AI predictions. The independent variables include the natural logarithm of total assets, market capitalization to book value of equity, debt to total assets, return on assets (ROA), cash to total assets, R&D expenses to total assets, capital expenditures to total assets, the natural logarithm of stock price on the day of the forecast, and the daily stock return volatility 90-day prior to the day of the forecast. All specifications use OLS regressions with time (quarterly) and firm (gvkey) fixed-effects. Robust standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

Financial Stmt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Interest Rate			Yes	Yes			Yes	Yes
Stock Price					Yes	Yes	Yes	Yes
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Log(Total assets)	0.0308*** (0.00795)	0.129*** (0.00915)	0.0267*** (0.00795)	0.123*** (0.00919)	0.0317*** (0.00803)	0.133*** (0.00939)	0.0249*** (0.00791)	0.126*** (0.00911)
Market to book	-0.00330*** (0.000645)	0.00796*** (0.000750)	-0.00323*** (0.000638)	0.00781*** (0.000752)	-0.00379*** (0.000663)	0.00788*** (0.000770)	-0.00410*** (0.000657)	0.00751*** (0.000756)
Debt to asset	0.134*** (0.0230)	-0.0888*** (0.0236)	0.133*** (0.0231)	-0.0846*** (0.0237)	0.130*** (0.0234)	-0.102*** (0.0240)	0.145*** (0.0234)	-0.0850*** (0.0241)
Return on assets	-0.642*** (0.0655)	-0.0951 (0.0661)	-0.615*** (0.0657)	-0.0826 (0.0661)	-0.628*** (0.0661)	-0.0564 (0.0670)	-0.625*** (0.0660)	-0.0585 (0.0666)
Cash to assets	-0.103*** (0.0229)	-0.00832 (0.0219)	-0.105*** (0.0227)	-0.0118 (0.0217)	-0.113*** (0.0233)	-0.0141 (0.0223)	-0.111*** (0.0230)	-0.0129 (0.0220)
R&D to assets	1.096*** (0.306)	1.205*** (0.303)	1.153*** (0.306)	1.260*** (0.305)	1.294*** (0.315)	1.406*** (0.314)	1.160*** (0.308)	1.271*** (0.306)
CapEx to assets	0.159** (0.0714)	0.429*** (0.0714)	0.176** (0.0707)	0.442*** (0.0708)	0.194*** (0.0723)	0.474*** (0.0729)	0.159** (0.0714)	0.439*** (0.0718)
Log(Stock price)		-0.158*** (0.00642)		-0.155*** (0.00639)		-0.163*** (0.00657)		-0.163*** (0.00645)
Return volatility		1.137*** (0.209)		0.972*** (0.209)		1.315*** (0.212)		1.242*** (0.211)
Institutional ownership		-0.0623*** (0.0128)		-0.0624*** (0.0131)		-0.0640*** (0.0131)		-0.0647*** (0.0130)
Constant	0.0909 (0.0585)	-0.174*** (0.0588)	0.129** (0.0584)	-0.126** (0.0587)	0.481*** (0.0591)	0.201*** (0.0599)	0.137** (0.0582)	-0.139** (0.0583)
Firm FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Time FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
N	85,086	85,086	85,086	85,086	85,086	85,086	85,086	85,086
R-squared (within firm)	0.253	0.301	0.259	0.305	0.377	0.420	0.263	0.312
R-squared (between)	0.001	0.002	0.135	0.025	0.004	0.001	0.004	0.001
R-squared (overall)	0.076	0.040	0.126	0.065	0.150	0.092	0.092	0.048

Table 17. Determinants of prediction deviation using ensemble machine learning

Panel A. Random Forest models

The dependent variable is the deviation between analyst and AI predictions. The independent variables include the natural logarithm of total assets, market capitalization to book value of equity, debt to total assets, return on assets (ROA), cash to total assets, R&D expenses to total assets, capital expenditures to total assets, the natural logarithm of stock price on the day of the forecast, and the daily stock return volatility 90-day prior to the day of the forecast. All specifications use OLS regressions with time (quarterly) and firm (gvkey) fixed-effects. Robust standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

Financial Stmt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Interest Rate			Yes	Yes			Yes	Yes
Stock Price					Yes	Yes	Yes	Yes
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Log(Total assets)	0.0148** (0.00745)	0.106*** (0.00878)	0.0140* (0.00750)	0.104*** (0.00883)	0.0154** (0.00738)	0.111*** (0.00867)	0.0162** (0.00740)	0.111*** (0.00868)
Market to book	-0.00397*** (0.000610)	0.00649*** (0.000703)	-0.00391*** (0.000612)	0.00642*** (0.000711)	-0.00451*** (0.000626)	0.00647*** (0.000712)	-0.00448*** (0.000627)	0.00634*** (0.000716)
Debt to asset	0.145*** (0.0220)	-0.0633*** (0.0226)	0.146*** (0.0220)	-0.0606*** (0.0227)	0.147*** (0.0222)	-0.0730*** (0.0228)	0.142*** (0.0223)	-0.0751*** (0.0229)
Return on assets	-0.558*** (0.0643)	-0.0413 (0.0644)	-0.584*** (0.0638)	-0.0736 (0.0640)	-0.581*** (0.0648)	-0.0351 (0.0647)	-0.586*** (0.0647)	-0.0475 (0.0646)
Cash to assets	-0.0885*** (0.0214)	0.000344 (0.0206)	-0.0874*** (0.0215)	0.000383 (0.0208)	-0.0961*** (0.0218)	-0.00295 (0.0211)	-0.0999*** (0.0218)	-0.00787 (0.0211)
R&D to assets	1.121*** (0.294)	1.220*** (0.290)	1.137*** (0.298)	1.234*** (0.294)	1.221*** (0.305)	1.324*** (0.302)	1.235*** (0.303)	1.336*** (0.300)
CapEx to assets	0.211*** (0.0671)	0.464*** (0.0678)	0.230*** (0.0673)	0.479*** (0.0680)	0.214*** (0.0685)	0.479*** (0.0689)	0.210*** (0.0684)	0.471*** (0.0692)
Log(Stock price)		-0.146*** (0.00609)		-0.144*** (0.00612)		-0.153*** (0.00614)		-0.151*** (0.00613)
Return volatility		1.292*** (0.205)		1.281*** (0.207)		1.475*** (0.206)		1.460*** (0.206)
Institutional ownership		-0.0621*** (0.0125)		-0.0629*** (0.0125)		-0.0627*** (0.0125)		-0.0653*** (0.0125)
Constant	0.229*** (0.0544)	-0.0256 (0.0551)	0.240*** (0.0547)	-0.0102 (0.0555)	0.209*** (0.0542)	-0.0623 (0.0547)	0.205*** (0.0543)	-0.0619 (0.0549)
Firm FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Time FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
N	85,086	85,086	85,086	85,086	85,086	85,086	85,086	85,086
R-squared (within firm)	0.100	0.152	0.100	0.151	0.101	0.158	0.101	0.157
R-squared (between)	0.034	0.000	0.044	0.000	0.031	0.000	0.027	0.000
R-squared (overall)	0.047	0.018	0.052	0.020	0.046	0.018	0.044	0.018

Panel B. Gradient Boosting model

The dependent variable is the deviation between analyst and AI predictions. The independent variables include the natural logarithm of total assets, market capitalization to book value of equity, debt to total assets, return on assets (ROA), cash to total assets, R&D expenses to total assets, capital expenditures to total assets, the natural logarithm of stock price on the day of the forecast, and the daily stock return volatility 90-day prior to the day of the forecast. All specifications use OLS regressions with time (quarterly) and firm (gvkey) fixed-effects. Robust standard errors are shown in the parentheses with ***, ** and * indicating its statistical significant level of 1%, 5% and 10% respectively.

Financial Stmt	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Interest Rate			Yes	Yes			Yes	Yes
Stock Price					Yes	Yes	Yes	Yes
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Log(Total assets)	0.0168** (0.00761)	0.113*** (0.00893)	0.0163** (0.00759)	0.112*** (0.00893)	0.0198*** (0.00757)	0.121*** (0.00892)	0.0190** (0.00760)	0.119*** (0.00894)
Market to book	-0.00406*** (0.000637)	0.00697*** (0.000726)	-0.00394*** (0.000629)	0.00708*** (0.000722)	-0.00454*** (0.000646)	0.00710*** (0.000737)	-0.00447*** (0.000643)	0.00706*** (0.000736)
Debt to asset	0.160*** (0.0229)	-0.0599** (0.0234)	0.159*** (0.0228)	-0.0605*** (0.0233)	0.162*** (0.0231)	-0.0711*** (0.0236)	0.161*** (0.0231)	-0.0707*** (0.0237)
Return on assets	-0.698*** (0.0672)	-0.153** (0.0669)	-0.694*** (0.0671)	-0.150** (0.0666)	-0.630*** (0.0669)	-0.0496 (0.0668)	-0.635*** (0.0668)	-0.0603 (0.0667)
Cash to assets	-0.0834*** (0.0224)	0.00985 (0.0214)	-0.0854*** (0.0223)	0.00777 (0.0214)	-0.0949*** (0.0225)	0.00332 (0.0215)	-0.0961*** (0.0225)	0.00126 (0.0215)
R&D to assets	0.915*** (0.312)	1.019*** (0.307)	0.907*** (0.308)	1.011*** (0.302)	0.972*** (0.313)	1.080*** (0.308)	1.015*** (0.317)	1.123*** (0.312)
CapEx to assets	0.246*** (0.0713)	0.511*** (0.0721)	0.238*** (0.0711)	0.503*** (0.0722)	0.264*** (0.0711)	0.543*** (0.0721)	0.266*** (0.0711)	0.543*** (0.0720)
Log(Stock price)		-0.154*** (0.00626)		-0.154*** (0.00627)		-0.162*** (0.00634)		-0.161*** (0.00634)
Return volatility		1.383*** (0.212)		1.362*** (0.212)		1.643*** (0.212)		1.631*** (0.212)
Institutional ownership		-0.0604*** (0.0127)		-0.0603*** (0.0128)		-0.0577*** (0.0127)		-0.0587*** (0.0127)
Constant	0.221*** (0.0557)	-0.0491 (0.0565)	0.223*** (0.0555)	-0.0453 (0.0564)	0.194*** (0.0553)	-0.0989* (0.0562)	0.199*** (0.0556)	-0.0905 (0.0565)
Firm FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Time FE	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
N	85,086	85,086	85,086	85,086	85,086	85,086	85,086	85,086
R-squared (within firm)	0.115	0.169	0.117	0.171	0.115	0.175	0.116	0.175
R-squared (between)	0.038	0.000	0.039	0.000	0.014	0.000	0.018	0.000
R-squared (overall)	0.051	0.018	0.052	0.019	0.037	0.015	0.041	0.017