Visualizing Earnings to Predict Post-Earnings Announcement Drift: A Deep Learning Approach

Jon A. Garfinkel[a], Paul Hribar[b], Lawrence Hsiao[c]

December, 2024

Abstract

We examine the potential of a deep learning model to use visualized earnings data to predict post-earnings announcement drift. After transforming quarterly earnings time series into bar chart images, we employ a convolutional neural network (CNN) to detect patterns and features within these visualizations that correlate with post-announcement drift. Out-of-sample tests reveal that the CNN-identified features significantly predict post-announcement returns, outperforming traditional drift predictors. This predictive capability remains consistent over time, is not accounted for by existing risk controls or known return anomalies and is robust across various model configurations. Our findings highlight the promise of applying AI to visualized financial data as a novel approach to predicting earnings changes and equity returns.

[a]Henry B. Tippie Research Professor of Finance, Tippie College of Business, University of Iowa, jon-garfinkel@uiowa.edu

[b]Henry B. Tippie Excellence Chair in Accounting and Professor, Tippie College of Business, University of Iowa, paul-hribar@uiowa.edu

[c]Assistant Professor of Finance, College of Management, National Taiwan University, and Researcher, Center for Research in Econometric Theory and Applications, National Taiwan University, lawrencehsiao@ntu.edu.tw

Hsiao gratefully acknowledges research support from the National Science and Technology Council (112-2410-H-002-152-) and the Ministry of Education of Taiwan (113L900203). We thank Joshua Yang for excellent research assistance.

"If I can't picture it, I can't understand it." –Albert Einstein

1 Introduction

Human brains process visual information more quickly than textual or numerical data, often interpreting visual stimuli instantaneously. Tools like graphs, charts, and infographics distill complex information into easily digestible formats, revealing patterns that may remain hidden in tables or raw numbers. The importance of visualization in financial reporting was highlighted by former SEC Chairman Christopher Cox, who remarked that "the visual presentation of information is such a key element of making disclosure understandable to investors" (SEC, 2007). Reflecting this advantage, financial disclosures increasingly incorporate visual elements to improve accessibility and engagement for diverse stakeholders, particularly investors. Recent research underscores this trend. For example, Christensen et al. [2024] document a significant increase in the use of visuals and infographics in 10-K filings, signaling a shift toward visual communication in financial reporting. Similarly, Nekrasov, Teoh, and Wu [2021] demonstrate that earnings announcements enhanced with visuals attract greater investor attention, as evidenced by higher engagement metrics such as retweet volumes on platforms like Twitter.

Simultaneously, advancements in AI and machine learning have introduced powerful tools for analyzing and interpreting financial statements. Recent research illustrates the usefulness of AI in this regard: Brown et al. (2020) employ a Bayesian topic-model algorithm to link specific topics in 10-K filings to financial misreporting risks; Bao et al. [2020] develop a machine learning model that predicts fraud using raw financial data; Chen et al [2022] apply machine learning to a detailed set of financial data to predict one-year ahead earnings changes; and Kim, Muhn, and Nikolaev [2024], demonstrate that a large language model (LLM) can analyze income statements and balance sheets without accompanying text and outperform analysts in predicting earnings changes. While these studies provide compelling evidence of the usefulness of AI at interpreting textual or numerical accounting data, the potential for analyzing *visualized* financial data remains largely unexplored.

Building on these developments, our study examines whether AI can perform financial analysis on visual representations of earnings data. Specifically, we investigate whether a trained AI model can use visualized earnings data to predict post-earnings announcement drift in a manner orthogonal to traditional drift predictors. To do so, we transform firms' historical quarterly earnings into bar charts and employ a convolutional neural network (CNN)—a deep learning algorithm inspired by the human visual system—to extract predictive features. CNNs excel at hierarchical feature extraction, with early layers identifying local patterns and deeper layers integrating them into complex, global insights. Our study aims to uncover whether CNNs can learn drift-predictive features from visualized earnings data and contribute new insights into the use of AI in financial analysis.

We begin by plotting earnings. For each firm announcing quarterly earnings from 1974Q1 to 2023Q2, we plot its most recent eight quarterly earnings in a bar chart that visualizes the magnitude as well as the sign of the earnings. We plot raw earnings figures rather than standardized unexpected earnings, as this approach better reflects the types of figures that firms typically showcase during their earnings calls. The plots are standardized and scaled so that the actual level of earnings is not discernible across different firms (see section 2 for details). Each earnings bar-chart image is paired with one of the three labels ("sell", "hold", or "buy") based on the relative performance of the firm's 63-day post-announcement buy-and-hold abnormal returns among the cross-section of firms in the same quarter. We then train the CNN model with 124,413 earnings images in a 20-year in-sample period (1974Q1 to 1993Q4) to autonomously learn features that best distinguish between the three assigned labels.

We apply the CNN stored parameters from the training phase to earnings images after 1993 to generate our key independent variable: "*CNN buy probability*". This variable can be thought of as the CNN-predicted likelihood for an image to be a "buy" when "sell" and "hold" options are also available. If features in earnings images are strongly associated with post-earnings announcement drift and the CNN is capable of detecting them during the training process, then firms with higher *CNN buy probability* should experience higher returns following their earnings announcement. To test this, in each quarter we sort firms into decile portfolios based on their CNN buy probability and then examine the average 63-day post-announcement buy-and-hold abnormal returns for each decile portfolio in the out-of-sample period from 1994Q3 to 2023Q2.

We find that when moving from the lowest decile to the highest decile of CNN buy probability, the average 63-day post-announcement buy-and-hold abnormal returns monotonically increase. Firms in the highest CNN buy probability decile outperform firms in the lowest CNN buy probability decile by 3.6% (14.4% annualized) using the market-adjusted buy-and-hold returns, with the return differential being positive in 99 out of 116 quarters. The lowest CNN buy decile associates with significantly negative BHARs, creating the more typical view of post-earnings announcement drift (positive surprises followed by positive drift and negative surprises followed by negative drift). This finding provides a clearer connection to the original drift puzzle compared to many follow-up studies, which have struggled to document the negative drift component (e.g., Garfinkel and Sokobin [2006]).

Our results are robust to alternative measures of buy-and-hold abnormal returns, such as size-adjusted buy-and-hold returns or buy-and-hold returns adjusted by factor models including the Fama-French four- and six-factor models (Fama and French

[1993], Carhart [1997], Fama and French [2015], Fama and French [2018]), the q^5-model (Hou et al. [2015], Hou et al. [2021]), and the risk-and-behavioral model (Daniel et al. [2020]). Hence, the CNN model's ability to detect drift-predicting features from earnings images does not appear attributable to existing risk factors.

We next compare the drift-predicting power of *CNN buy* features to fourteen other well-known anomalies and potential determinants of post-announcement returns including standardized unexpected earnings (Ball and Brown [1968], Bernard and Thomas [1989], Foster et al. [1984]), earnings acceleration (He and Narayanamoorthy [2020]), trend in gross profitability (Akbas et al. [2017]), market capitalization (Fama and French [1992], [1993]), book-to-market ratio (Fama and French [1992], [1993]), earnings announcement return (Foster et al. [1984], Chan et al. [1996]), pre-announcement return (Carhart [1997]), earnings persistence (Francis et al. [2004]), earnings volatility (Cao and Narayanamoorthy [2012]), gross profitability (Novy-Marx [2013]), operating profitability (Ball et al. [2016]), operating accruals (Sloan [1996], Hribar and Collins [2002]), total accruals (Richardson et al. [2005]), and asset growth (Cooper et al. [2008]). Using univariate portfolio analysis, we find that the post-announcement return differentials between the highest and lowest deciles, when sorted by each of the 14 firm characteristics, are smaller in magnitude than those sorted by the CNN buy probability.

We further examine whether the drift-predicting power of CNN buy *overlaps* with the aforementioned anomalies or captures unique mispricing characteristics. We simultaneously control for these other anomalies by running quarterly weighted Fama and MacBeth [1973] regressions of post-earnings announcement drift on the CNN buy probability and the 14 other anomaly indicators. The coefficient on the CNN buy probability is positive and highly significant, indicating that the CNN buy features provides incremental predictability for post-earnings announcement drift over existing well-known anomalies.

To gain more insight into how the CNN uses the earnings charts, we regress the *CNN buy* probability on the most recent eight quarterly earnings. We find that non-linear transformations of the earnings are an important determinant of the CNN buy recommendations. A linear association of the eight quarters of historical earnings explains only 30% of the variation in *CNN buy* probability, suggesting that most of the variation in *CNN buy* probability is driven by nonlinear transformations of the underlying historical earnings data.

We then explore the nature of the price relevant information in *CNN buy* features that is apparently missed by the market, leading to the post-earnings announcement drift. We hypothesize that the *CNN buy probability* has implications for future earnings growth that is overlooked by investors (e.g. Bernard and Thomas 1989). We find that *CNN buy probability* positively predicts one-quarter-ahead unexpected earnings as well as three-day abnormal returns around the next earnings announcement date, controlling for past earnings growth and other firm characteristics. We conduct a Mishkin test (Mishkin [1983], Abel and Mishkin [1983]), to show that the drift-predicting ability of *CNN buy* features likely manifests because investors underestimate the implications of past earnings features for future earnings growth.[1]

We perform a battery of additional robustness tests. First, we find that employing CNN predictions in a more conservative monthly-rebalancing long-short strategy yields a monthly return of around 1%. Second, we show that the out-of-sample performance of CNN predictions is insensitive to various model specifications, mitigating the concern that certain model hyperparameters are driving the results. We also train a one-

[1]Prior studies employing the Mishkin test framework include Sloan [1996], Dechow and Sloan [1997], Rangan and Sloan [1998], Collins and Hribar [2000], Narayanamoorthy [2006], Cao and Narayanamoorthy [2012], Chen and Shane [2014], Hui et al. [2016], Ma and Markov [2017], and He and Narayanamoorthy [2020].

dimensional CNN model with firms' time-series of raw earnings data and document that the out-of-sample drift-predicting performance is inferior to that of our two-dimensional CNN model, thus emphasizing the importance of image representation.

Our study make several contributions to the literature. First, we add to a growing literature studying the applications of machine learning techniques in financial statement analysis (e.g. Brown et al. [2020], Bao et al. [2020] Chen et al. [2022], Kim et al. [2024]) and returns prediction (Rapach et al. [2013], Kelly et al. [2019], Feng et al. [2020], Freyberger et al. [2020], Kozak et al. [2020], Gu et al. [2020], Gu et al. [2021], Leippold et al. [2022], Cao et al. [2024], Chen et al. [2024], Murray et al. [2024]). The key differentiating feature of our approach from the above studies is that our input focuses on how machine learning can use visual representations of earnings to predict returns and future earnings changes.[2]

Second, we contribute to the recent literature examining the information value in visualized data. For example, Nekrasov et al. [2022] find that visuals in firms' Twitter earnings announcements are associated with more retweets, representing increased attention to the earnings news. Moss [2022] finds that retail investors use their visual perception of earnings surprise displayed on Robinhood rather than the unexpected earnings scaled by stock price in their investment decisions. Hu and Ma [2023] quantify persuasion in visual, vocal, and verbal dimensions in start-up pitch videos, and find that passionate and warm pitches significantly increase funding probability. Cao et al. [2024] examine the value of visual information provided in corporate executive presentations and use AI to categorize the types of charts presented as forward looking or summarizing, and examine how market participants respond to such information. Christensen et al. [2024] document a significant increase in the disclosure of

[2]Transforming data from one-dimensional to two-dimensional can potentially create more nuanced information, thus adding flexibility in prediction tasks when the input data is scarce.

infographics in 10-K filings over time, and investigate the relation between the use of infographics and uncertainty in capital markets. Gu et al. [2024] find that a daily firm-level investor sentiment measure based on graphics interchange format images (GIFs) in postings about firms on Stocktwits.com is positively correlated with same-day stock returns while predicting stock return reversals in the following two weeks. Our paper, on the other hand, proposes a universal approach to visualize a firm's time-series of quarterly earnings into a bar-chart image that accounts for both the sign and magnitude of earnings as an input to AI, to examine whether it can use the information to predict post-earnings announcement drift.

Last, we contribute to a burgeoning literature employing CNN to extract relevant information from images. For example, Obaid and Pukthuanthong [2022] extract information from a large sample of news media images and translate that information into a daily investor sentiment index. Jiang et al. [2023] extract return-predicting information from stock-level charts depicting daily open, close, high, and low prices, as well as trading volume and average prices over a past period, to forecast future returns.

The rest of the paper is structured as follows. Section 2 describes the data and variables. Section 3 describes how we generate earnings images, assign labels, and train the CNN model. Section 4 presents the out-of-sample drift-predicting performance of the CNN model. Section 5 examines the nature of CNN predictions and the source of their drift-predicting power. Section 6 performs robustness checks. Finally, Section 7 concludes the paper.

2. Data and Variables

We focus on U.S. common stocks traded on NYSE, AMEX, and NASDAQ, and obtain data from Compustat and CRSP. First, we collect Compustat firm-quarters whose earnings announcement date (Compustat item RDQ) is between January 1974 and June 2023, and delete observations with missing RDQ in the most recent eight quarters (quarters $q-7$ to q). Next, we apply filters in He and Narayanamoorthy [2020] to eliminate announcements that are potentially subject to data errors. In particular, we delete observations if in the most recent eight quarters, a firm has (i) more than one earnings announcement on any date (ii) earnings announcement date within 30 days of a previous earnings announcement date, or (iii) earnings announcement either prior to or more than 180 days after the corresponding fiscal period-end.

We require a firm to have non-missing earnings in the most recent eight quarters and a CRSP daily price higher than one dollar at the most recent earnings announcement date (quarter q). We use income before extraordinary items (Compustat item IBQ) as earnings. Financial and utility firms with SIC codes from 6000 to 6999 and from 4900 to 4949 are excluded. In addition, firms are required to have non-missing market capitalization (SIZE) and non-negative book-to-market ratio (BM), and have at least 90 non-missing daily return observations in the $[-150, -31]$ window relative to the current quarter earnings announcement date. We are left with 404,635 firm-quarter observations after applying all the above filters.

Next, we define the in-sample dataset and the out-of-sample dataset. The in-sample dataset consists of 124,413 firm-quarter observations between January 1974 to December 1993 (4,548 firms; 80 quarters), and is for CNN model training and validation. The complement out-of-sample dataset is for predicting and testing the out-of-sample CNN model performance, and thus serves as the dataset for all empirical analyses throughout the paper. It consists of 240,844 firm-quarter observations between July 1994 to December 2023 with non-missing firm characteristics (7,527 firms; 116

quarters).[3]

We summarize the definitions of all the variables used in this study, in the Appendix. To mitigate the impact of outliers, we transform most variables into decile ranks (numbered 0 to 9, from low to high) following prior research (e.g., Rangan and Sloan [1998], Livnat and Mendenhall [2006], Garfinkel and Sokobin [2006]).[4] The cutoff points for quarterly variables are based on the distribution of these variables in the previous quarter, and the cutoff points for annual variables from July in year t to June in year $t + 1$ are based on the distribution of these variables at the end of June in year t. Then, we convert all the decile ranks to scaled ranks by dividing by 9 and subtracting 0.5. The resulting scaled ranks vary from −0.5 to 0.5 with a mean of zero and a range of one. This variable transformation approach is to facilitate comparison of the economic magnitudes of firm characteristics. For example, the coefficient on a variable of interest (in scaled rank) in a return regression represents the return from a zero investment strategy of going long on the highest variable decile and short on the lowest variable decile.

3 The CNN Model and research design

In this section, we introduce the CNN training procedure, which can be thought of as an image classification task. First, we transform firms' times series of quarterly earnings into bar charts. Then, we assign labels to each earnings bar-chart image based

[3]The firm characteristics include SUE, EA, TREND, RET[−1,1], RET[−30,−2], PERSIST, VOL, GP, OP, OA, TA, and AG. Along with BM and SIZE, these firm characteristics are used as comparing/control variables throughout the paper. See Appendix for variable definitions. In addition, we address the reasons to set up a six-month lag between the end of the in-sample dataset and the out-of-sample dataset in Section 4.1. In addition, our empirical results are robust to using an in-sample (out-of-sample) period consisting of 60 quarters (136 quarters) or 100 quarters (96 quarters).

[4]Variables that are not transformed into decile ranks are the six measure of the 63-day post-announcement buy-and-hold abnormal return (BHAR), including market-adjusted return (MAR), size-adjusted return (SAR), and four factor-adjusted returns (FF4, FF6, HMXZ5, and DHS3).

on the relative performance of its post-earnings announcement drift among the cross-section of firms in the same quarter. Finally, we train the CNN model with the 124,413 earnings images in the in-sample period (1974Q1 to 1993Q4, 80 quarters) to "learn" drift-predicting information.

3.1 Generating earnings images

We begin by plotting the most recent eight quarterly earnings in bar charts. Our intent was to create a simple chart that would be roughly analogous to what might be presented in earnings conference call. For example, Figure 1 panel A provides a slide from Meta's 2024 Q3 earnings call where they display nine quarters of past earnings. Rather than have the CNN attempt to classify the variety of earnings images generated by firms, we provide a set of standardized charts for the CNN to train on and process. Following Jiang et al. [2023], we generate black-and-white rather than colored images for simplicity and uniformity. Each black-and-white image is of size 24 × 24 pixels, which is recognized by the machine as a 24 × 24 matrix of 0 (black pixel) and 255 (white pixel). We use black as the background color and white as the color for earnings, and the constant image size setup is for better comparison of earnings patterns across different firms in different quarters. Figure 1, Panel B provides an example of Meta's 2024 Q3 earnings in our standardized format.

Each quarter occupies 24 × 3 pixels in the image, and quarterly earnings are plotted as "white bars" in the middle column of each quarter. In particular, let $E_1, E_2, \dots, E_8$ denote the most recent eight quarterly earnings corresponding to quarter $q-7, q-6, \dots, q$, E_{MAX} and E_{MIN} denote the maximum and minimum of the eight quarterly earnings, and $r(\,)$ denote the function that rounds the input value to the nearest whole number. We set the bottom-left vertex of the image as the origin of a two-dimensional coordinate system, so a rectangular area in the image can be represented as $([x_1, x_2], [y_1, y_2])$. Next, we classify firms' most recent eight quarterly earnings into one of the three types, determine the values corresponding to the top and bottom of the image, and plot each quarterly earnings into bars accordingly. The three types are as follows:

- Type I ($E_{\mathrm{MIN}} \geq 0$; the most recent eight quarterly earnings are all non-negative): In this case, we set E_{MAX} and 0 as the top and bottom of the image, respectively. E_{i} is plotted as the area of

$$\left([3i-2, 3i-1], \left[0, r\left(24 * \frac{E_i}{E_{\mathrm{MAX}}}\right)\right]\right), \tag{1}$$

  for i = 1, ..., 8. Figure 2 displays an example earnings image of this type. The maximum earnings is E_7 and thus it occupies a whole column. All other quarterly earnings are plotted upward, and their heights are determined using E_7 as the reference point.

- Type II ($E_{\mathrm{MAX}} > 0$ and $E_{\mathrm{MIN}} < 0$; the maximum quarterly earnings is positive while the minimum earnings is negative): In this case, E_{MAX} and E_{MIN} coincide with the top and bottom of the image, respectively. The implicit "zero-earnings line" corresponds to $r\left(24 * \frac{-E_{MIN}}{E_{MAX}-E_{MIN}}\right)$, and E_{i} is plotted above or below the zero-earnings line as follows:

$$\begin{cases} [3i-2, 3i-1], \left[r\left(24 * \frac{-E_{MIN}}{E_{MAX}-E_{MIN}}\right), r\left(24 * \frac{-E_{MIN}}{E_{MAX}-E_{MIN}}\right) + r\left(24 * \frac{E_i}{E_{MAX}-E_{MIN}}\right)\right] & if\ E_i > 0, \\ [3i-2, 3i-1], \left[r\left(24 * \frac{-E_{MIN}}{E_{MAX}-E_{MIN}}\right) - r\left(24 * \frac{-E_i}{E_{MAX}-E_{MIN}}\right), r\left(24 * \frac{-E_{MIN}}{E_{MAX}-E_{MIN}}\right)\right] & if\ E_i \leq 0, \end{cases} \tag{2}$$

  for i = 1, ..., 8. Figure 3 displays an example earnings image of this type. Here we see the advantage of using bar charts as opposed to line graphs when plotting earnings. Bars can represent positive, zero, or negative earnings without further specifying numbers on the vertical axis. Positive earnings are plotted upward while negative earnings are plotted downward, and the bar lengths (in pixels) are computed as the rounded value of 24 multiplied by the absolute values of E_{i} scaled by $E_{\mathrm{MAX}} - E_{\mathrm{MIN}}$.

- Type III ($E_{\mathrm{MAX}} \leq 0$; the most recent eight quarterly earnings are all non-positive): In this case, 0 and E_{MIN} coincide with the top and bottom of the image, respectively. E_{i} is plotted as the area of

$$[3i - 2, 3i - 1], \left[24 - r\left(24 * \frac{E_i}{E_{\mathrm{MIN}}}\right), 24\right], \tag{3}$$

  for i = 1, ..., 8. Figure 4 displays an example earnings image of this type. The minimum earnings is E_5 and thus it occupies a whole column. All earnings are plotted downward, and their heights are plotted using E_5 as the reference point.

Note that in all three types, it is possible for E_{i} to be very close to zero after scaling and thus does not occupy a full pixel in the image after rounding, i.e., $y_1 = y_2$.[5] In addition, the distance between two neighboring earning of pixel between is greater than the distance between the leftmost (or rightmost) earnings and the border of the image, which is consistent with the default setup of a bar chart for most statistical software.

[5] One extreme case is that all eight quarterly earnings are very close to each other so that when plotting earnings on a bar chart, each earnings bar occupies a whole column. In this case, one cannot tell from the image whether all earnings are positive or negative. However, we checked all earnings images and did not find this extreme case.

3.2 Assigning labels to earnings images

Next, we assign one of the three labels ("sell", "hold", or "buy") to each firm's earnings image based on the firm's 63-day post-announcement buy-and-hold abnormal return (BHAR). There are different ways to define buy-and-hold abnormal returns, and we use market-adjusted buy-and-hold returns (MAR).[6] In particular, $\text{MAR}_{i,q+1}$ is defined as the difference between the buy-and-hold return of firm i and that of the CRSP value-weighted market portfolio over the windows [2, 64] in trading days relative to firm i's earnings announcement date t in quarter q:

$$\text{MAR}_{i,q+1} = \prod_{k=t+2}^{t+64} \left(1 + R_{i,k}\right) - \prod_{k=t+2}^{t+64} \left(1 + R_{M,k}\right) , \tag{4}$$

where R_i is the delisting-adjusted return of firm i, R_M is the return of the CRSP value-weighted market return, and t is quarter q's announcement date of firm i.[7] The 63-day holding window corresponds to the total number of trading days in three months. We follow previous studies (Vega [2006], Engelberg et al. [2012], Frank and Sanati [2018]) to compute MAR from day 2 to mitigate the impact of bid-ask bounce and other market microstructure effects, and our results are robust to MAR defined using the trading window of [1, 63].

Then, for each quarter in the in-sample period (1974Q1 to 1993Q4, 80 quarters), we sort firms announcing earnings into terciles based on their 63-day MAR. The bottom, mid, and top terciles are labeled "sell", "hold", and "buy", respectively. Since the number of training images for each label is about the same, we mitigate the class

[6] In untabulated tests, we find that the final CNN out-of-sample performance is robust to using alternative definitions of abnormal returns such as size-adjusted or factor-adjusted returns in the label-assigning process.

[7] We replace missing delisting-adjusted returns with market returns, which is equivalent to reinvesting any remaining proceeds in the market portfolio until the end of the holding period.

imbalance issue in CNN training that arises with a disproportionate ratio of labels.

3.3 CNN architecture and training

In this section, we introduce the general algorithm of CNN and describe the architecture and training process of our CNN model.

3.3.1 CNN architecture

A CNN model typically consists of multiple building blocks, with each building block consisting of a convolutional layer and a pooling layer.[8] In the convolutional layer, an input image is first scanned by a set of convolutional filters to generate feature maps. Convolutional filters, also known as kernels, are small matrices (usually of size 3×3, 5×5, or 7×7 pixels) that are applied over the input image's pixels to detect features. The matrix elements in convolutional filters are also known as "weights", which will later be optimized during the training process of the CNN model.

Then, a filter "scans" an image. It starts at the top-left corner of an image and moves one pixel at a time. In CNN terminology, this corresponds to a "stride" of 1, which is usually the default option in convolution. In each position, an element-wise multiplication is performed between the filter and the corresponding patch of the input image. The products are summed to a single value placed in the corresponding position of the feature map.[9] The process is repeated until the filter slides across the entire image, thus generating a complete feature map. Then, the feature map passes through

[8]A building block may also consist of multiple convolutional layers, working sequentially (as in our case) or in parallel (e.g., GoogLeNet (Szegedy et al. [2015])).

[9]In CNN terminology, a feature map is the output produced after applying a convolutional filter to an input image or the previous layer's feature map.

an activation function to introduce non-linearity.[10] Finally, in the pooling layer, pooling operations are applied to the feature map produced by the convolutional layer (after activation) to retain the most important information.

Figure 5 illustrates the operation of a CNN building block using a black-and-white 6 × 6 pixels input image and a 3 × 3 pixels convolutional filter as an example. We first apply "padding" to the input image by filling the absent neighbor elements with zeros for elements at the image's border, which helps preserve input dimensions and allow better edge feature detection.[11] Then, the convolutional layer applies the filter to the input image and produces a feature map of size 6 × 6 pixels. In particular, we see that the top-left (bottom-right) 3 × 3 pixels patch of the input image, after applying the 3 ×3 pixels convolutional filter, eventually becomes the top-left (bottom-right) element in the output feature map.

Next, we use "Leaky ReLU" as our activation function, which is a variation of the conventional ReLU function.[12] In particular, Leaky ReLU function transforms an input value x to itself if $x > 0$ and $0.01x$ otherwise. We see that in Figure 5, Leaky ReLU function is applied to every element of the feature map generated by the convolutional layer, and a feature map of the same size (6 × 6 pixels) is produced. Finally, we employ the most commonly used 2 × 2 pixels max-pooling filter with a stride of 2 as our pooling operation function. The max-pooling filter scans the feature map produced by the convolutional layer (after activation), selects the maximum element in the 2 × 2 pooling window, and eventually shrinks the height and width of the input feature map by half. Hence, max-pooling helps preserve the most prominent features and reduce the spatial

[10] Without activation functions, CNNs would just consist of linear operations (matrix multiplication).

[11] Convolution operations without padding inevitably reduce the spatial dimensions of the output feature maps.

[12] Compared to ReLU, Leaky ReLU allows a small, non-zero gradient for negative input values, which helps to address the "dying ReLU" problem (where some neurons can become permanently inactive during training) and thus enables more robust learning in neural networks. Interested readers may refer to Maas et al. [2013], which is the first modern deep learning reference to Leaky ReLU.

dimensions of the input feature maps.

Having introduced the components of a CNN building block, we proceed to describe the architecture of our CNN model and illustrate the details in Figure 6. In particular, we use three building blocks, with the first block consisting of 64 convolutional filters of 7 × 7 pixels, the second block consisting of 128 convolutional filters of 3 × 3 pixels, and the third block consisting of 256 convolutional filters of 3 ×3 pixels.[13] Since our input images are black-and-white with low resolution (24 ×24 pixels), we do not resort to an overly complicated CNN model with too many blocks. In addition, we employ filters of 7 × 7 pixels in the first block to ensure that pixels of neighboring quarterly earnings in the initial input earnings image can always be scanned simultaneously by the convolutional filters.[14]

After passing an image sequentially through the three building blocks, the CNN model "flattens" the elements in the feature maps generated by the last block to a vector.[15] Then, the fully connected layer linearly transforms the elements in the vector to three scores (Z_1, Z_2, and Z_3) of the three labels (label 1 = "sell", label 2 = "hold", label 3 = "buy").[16] Finally, the Softmax function transforms these scores to three probabilities ($\widehat{y_1}, \widehat{y_2}, \widehat{y_3}$), where $0 \leq \widehat{y_i} \leq 1$ and $\sum_{i=1}^{3} \widehat{y_i} = 1$.[17] Hence, one can

[13]We follow the literature and increase the number of filters after each convolutional layer by a factor of two. Following several well-known CNN architectures (e.g., VGGNet (Simonyan and Zisserman [2014]), ResNet (He et al. [2016]), DenseNet (Huang et al. [2017]), we employ 64 filters in the first convolution layer. The choice of 64 filters provides a balance between model complexity and computational efficiency, making it a popular choice.

[14]This filter size choice is based on the hypothesis that certain patterns in neighboring quarterly earnings are helpful in predicting post-earnings announcement drift, although in Table 9 we find that the main results are robust to using filters of 3 × 3 or 5 × 5 pixels in the first block.

[15]In the second and third building blocks where the input is a "stack" of feature maps instead of a single image, the output will be a stack of feature maps as well. In particular, each filter is applied to the stack of feature maps to perform convolution and eventually generate one feature map. Hence, the total number of feature maps in the output is equal to the number of filters. In CNN terminology, the number of stacks is usually referred to as the "depth" of the input.

[16]The linear transformation also requires parameters to be estimated and optimized in the training process.

[17]The Softmax function converts the three scores into a probability distribution of three outcomes, i.e.,$\widehat{y_i} = \frac{e^{Z_i}}{\sum_{k=1}^{3} e^{Z_k}}$, for $i = 1, 2, 3$.

interpret $\widehat{y_3}$ as the CNN-predicted likelihood of an earnings image to be classified as "buy" when the other two labels are available. Throughout the paper, we refer to $\widehat{y_3}$ as the *CNN buy* probability (CNNBP).

3.3.2 CNN training

CNN training is about finding the optimized weights, i.e., parameters in convolutional filters and the fully connected layer, to minimize model "loss" to a certain extent. We follow the CNN literature to use the cross-entropy loss function as the loss function for minimization. In particular, let $y = [y_1, y_2, y_3]'$ denote the label of an earnings image, which is either [1,0,0]′, [0,1,0]′, or [0,0,1]′ corresponding to the sell, hold, and buy label, respectively. The cross-entropy loss is computed as

$$Loss(y, \hat{y}) = -\sum_{i=1}^{3} y_i * \log \widehat{y_i} \tag{5}$$

where loss $\in [0,\infty)$ and smaller loss represents better CNN performance.

We closely follow the regularization procedures in Gu et al. [2020] and Jiang et al. [2023] to train our CNN model.[18] From the in-sample period (1974Q1 to 1993Q4, 80 quarters), we randomly select 70% earnings images for training and the other 30% for validation, which are labeled training dataset and validation dataset, respectively. First, 128 earnings images (batch size = 128) are randomly selected from the training dataset and are passed through the CNN model as described in Figure 6 to produce the average loss. The loss is propagated back through the model to update the weights via stochastic gradient descent and the Adam algorithm (Kingma and Ba [2014]) with a learning rate

[18]Interested readers may refer to Gu et al. [2020] for detailed explanations on those modeling choices. In particular, we use batch normalization (Ioffe and Szegedy [2015]) to mitigate the internal covariate shiftproblem, choose Xavier initializers (Glorot and Bengio [2010]) as initial weights for model training, and apply a 50% dropout rate (Srivastava et al. [2014]) to the fully connected layer to prevent over-fitting. In Table 9, we show that none of these choices affect our main results in Table 2.

of 10^{-5}. Next, the model randomly selects 128 images from the remaining images in the training dataset to update the weights. The iteration process stops when the model sees all earnings images in the training dataset. We then apply those updated weights to the earnings images in the validation dataset to compute validation loss. After completing the above process, we finish training an "epoch".

Next, we start the training process again using the updated weights of the first epoch as the initial weights, and eventually obtain updated weights and validation loss of the second epoch. This training iteration process is halted only when the validation loss fails to improve for two consecutive epochs, and the updated weights of the third-to-last epoch are stored as the optimized weights.[19] Then, we can then apply these optimized weights to a new earnings image to generate CNN-predicted likelihood for a label of interest.

4. Empirical Results

In this section, we examine the out-of-sample performance of the CNN trained model. We begin by applying the CNN-trained weights to the earnings images in 1994Q2 to 2023Q2 to generate the *CNN buy probability*.[20] Since CNN training can result in different outcomes even when using the same architecture and dataset due to the stochastic nature of optimization algorithms and the use of dropout rate, we train the same CNN model independently ten times (number of ensembles = 10) and then average the *CNN buy* probability, which helps achieve better accuracy and robustness.

If the CNN model is capable of extracting features that are indicative of post-earnings announcement performance (*CNN buy* features) from earnings images, there

[19]This technique is called 'early stopping", which is to prevent over-fitting to the training data.

[20]We start from 1994Q2 as late announcers in 1993Q4 require post-earnings announcement drift data in 1994Q1 to form labels.

should be a positive relation between the *CNN buy* probability and post-earnings announcement returns. Moreover, if the CNN is able to identify patterns that predict post announcement returns, we are also interested in the extent to which this is orthogonal to other known anomalies.

4.1 Portfolio analysis: univariate sorts

4.1.1 CNN buy probability and post-earnings announcement drift

In each quarter starting from 1994Q3, we assign firms announcing earnings into decile portfolios based on their *CNN buy* probability, where the cutoffs are based on the distribution of the previous quarter's *CNN buy* probability. This approach prevents hindsight bias from classifying firms into portfolios based on information not available at the time the strategy is implemented (Foster et al. [1984], Bernard and Thomas [1989]). Hence, the out-of-sample period is from 1994Q3 to 2023Q2 (a total of 116 quarters). Next, we compute the average 63-day MAR for each *CNN buy* probability decile. If the CNN model is competent in detecting the features of images that are indicative of future performance, the average 63-day MAR should monotonically increase when going from the lowest to the highest *CNN buy* probability decile.

Table 2 presents the results. We find that when moving from the lowest decile to the highest decile of *CNN buy* probability, the average *CNN buy* probability increases from 25.5% to 44.4%, and the average 63-day MAR increases monotonically from −0.6% to 3.0%.[21] The average difference in MAR is 3.6% (*t*-statistic = 7.213) in a quarter, which corresponds to an annualized return exceeding 14%. Figure 7 further depicts the return differential for each of the 116 quarters in the out-of-sample period.

[21] Note that random assignment would result in a 33.3% buy probability.

Specifically, the hedge return is positive in 99 out of the 116 quarters (85.3%, z-stat=13.26, $p < 0.0001$), indicating that the CNN out-of-sample performance is highly significant and reasonably stable over time.

We check that the results are robust to alternative risk adjustments by also examining size-adjusted (SAR) and factor-adjusted buy-and-hold returns. SAR is defined as the difference between the buy-and-hold return of an announcing firm and that of a size-matched portfolio over the 63-day window ([2, 64]) following its earnings announcement date. We use the monthly NYSE size decile breakpoints at the end of June in year t to determine the size-matched portfolio for a firm whose earnings announcement date is between July of year t to June of year $t+1$. Monthly size breakpoints and daily size portfolio returns are obtained from Kenneth French's website.

To compute the 63-day factor-adjusted buy-and-hold returns, we replace $R_{M,k}$ in equation (4) with daily return $\widehat{R_{F,k}}$ predicted by factor models. To compute $\widehat{R_{F,k}}$, we first estimate individual stock factor loadings by regressing returns on the factors on a 120-day rolling window from $t - 150$ to $t - 31$ for each stock:

$$r_{i,t} = \alpha_i + \beta_i' F_t + \epsilon_{i,t}, \tag{6}$$

where $r_{i,t}$ is the excess return on stock i and F_t is a vector of factors. The predicted return $\widehat{R_{F,k}}$ is then computed as $\widehat{\beta'}_\iota F_k$.[22] In particular, we consider the factors in the Fama-French four- and six-factor models (Fama and French [1993], Carhart [1997], Fama and French [2015], Fama and French [2018]), the q^5-model (Hou et al. [2015], Hou et al. [2021]), and the risk-and-behavioral model (Daniel et al. [2020]). The 63-day factor-adjusted buy-and-hold returns following an earnings announcement of these models are denoted FF4, FF6, HMXZ5, and DHS3, respectively.[23]

[22]See, for example, Savor [2012] and Kapadia and Zekhnini [2019].

[23]Fama and French [2015] extends the Fama-French three-factor model (Fama and French [1993])

Columns 4 to 10 in Table 2 present qualitatively similar results when we compute the average 63-day SAR or factor-adjusted buy-and-hold returns (FF4, FF6, HMXZ5, and DHS3) for each *CNN buy* probability decile. The return differential between the highest and lowest *CNN buy* probability deciles range from 3.1% to 3.5%, with *t*-statistics all statistically significant at the 1% level. Overall, Table 2 shows a significantly positive relation between *CNN buy* probability and post-announcement buy-and hold abnormal returns that are robust to various risk adjustments.[24]

4.1.2 Comparing to traditional drift indicators and other anomalies

We next examine whether the *CNN buy* probability is superior to the other known anomalies or determinants of drift. We first consider three earnings attributes: standardized unexpected earnings (Ball and Brown [1968], Bernard and Thomas [1989], Foster et al. [1984]), earnings acceleration (He and Narayanamoorthy [2020]), and trend in gross profitability (Akbas et al. [2017]). Standardized unexpected earnings (SUE) is the earnings surprise based on a seasonal random walk model, earnings acceleration (EA) captures the change in earnings growth from one quarter to the next, and trend in gross profitability (TREND) characterizes the recent path in a firm's profitability in addition to the profit level.

In addition to the three earnings attributes, we also compare to a host of known

---

to control for operating profitability (RMW) and investment (CMA). After the inclusion of a momentum factor (Carhart [1997]), we have Fama-French four-factor and six-factor models (Fama and French [2018]). Hou et al. [2015] propose the q-model to control for market, size (ME), investment (IVA), and profitability (return on equity, ROE), and Hou et al. [2021] further includes an expected growth factor (EG) into the q^5-model. Daniel et al. [2020] propose a 3-factor risk-and-behavioral model that accounts for market, long-term financing (FIN), and short-term earnings surprise (PEAD). Fama-French factors are obtained from Kenneth French's website, q^5-model factors are obtained from Lu Zhang's website, and DHS3 factors are obtained from Lin Sun's website.

[24]The results are qualitatively the same if we assign earnings images with labels based on firms' 21-day or 42-day post-announcement market-adjusted buy-and-hold returns, train the CNN model, and examine the 21-day or 42-day post-announcement buy-and-hold abnormal returns for *CNN buy* probability deciles.

anomalies: market capitalization (Fama and French [1992], [1993]), book-to-market ratio (Fama and French [1992], [1993]), earnings announcement return (Foster et al. [1984], Chan et al. [1996]), pre-announcement return (Carhart [1997]), earnings persistence (Francis et al. [2004]), earnings volatility (Cao and Narayanamoorthy [2012]), gross profitability (Novy-Marx [2013]), operating profitability (Ball et al. [2016]), total accruals (Richardson et al. [2005]), operating accruals (Sloan [1996], Hribar and Collins [2002]), and asset growth (Cooper et al. [2008]).

We follow the approach in the previous section to assign firms announcing earnings into decile portfolios based on one of the 14 firm characteristics, where the cutoffs are based on the distribution of the previous quarter's firm characteristic. Then, we compare the difference in the average 63-day MAR, SAR, and factor-adjusted buy-and-hold returns (FF4, FF6, HMXZ5, and DHS3) between the highest and lowest deciles of each characteristic. If a firm characteristic is more successful in predicting post-earnings announcement drift, the difference in the post-announcement buy-and-hold abnormal returns between the highest and lowest deciles sorted on the characteristic should be larger in magnitude.

Table 3 reports the results. The first row presents the return differential between the highest and lowest deciles sorted on *CNN buy* probability (ranging from 3.1% to 3.6%), which is the same as the last row in Table 2. In comparison, we find that the statistically significant return differential ranges from 2.0% to 2.6% for SUE, from 2.1% to 2.4% for EA, from 1.4% to 2.0% for TREND, from 3.3% to 3.8% for RET[−1, 1], from 1.6% to 2.5% for BM, from 1.0% to 2.4% for GP, and from −1.9% to −1.6% for AG. For the other firm characteristics, the return differential fails to remain statistically significant at the 10% level across all six abnormal return measures. Overall, Table 3 suggests that the drift-predicting power of the *CNN buy* features is superior to that of the usual determinants of PEAD, while being roughly on par with that of the earnings announcement return (RET[−1, 1]). Importantly, however, this test does not indicate the amount of overlap among the CNN identified hedge portfolios and the existing anomalies, which we investigate next.

4.2 Portfolio analysis: double sorts

In this section, we examine the extent to which the drift-predicting power of *CNN buy* features is incremental to that of the 14 other drift predictors and anomalies. We construct 5 × 5 portfolios sorted independently (Liu et al. [2018], He and Narayanamoorthy [2020]) on the *CNN buy* probability and one of the six firm characteristics (SUE, EA, TREND, RET[−1, 1], BM, GP, and AG). Again, to alleviate hindsight bias, we use the distribution of each firm characteristic in the previous quarter to form the quintile cutoffs. Then, we examine the average 63-day MAR of the 25 portfolios.[25]

We present the double sorts results in Table 4. Consistent with previous findings in the literature, in Panel A we find that firms with high SUE outperform firms with low SUE by a quarterly return of 0.9% to 2.5% depending on the *CNN buy* probability quintile. On the other hand, the return differential between the high and low *CNN buy* probability quintiles is significantly positive across medium to high SUE quintile while being insignificantly positive in the bottom two quintiles. This finding indicates that the *CNN buy* features exhibit incremental drift-predicting power for medium to high SUE firms, while this predictive ability appears to be subsumed by the SUE effect for low SUE firms.

In Panel B we present the analogous two-way sorting based on firms' earnings acceleration. We find that the average difference in 63-day MAR between the top and

[25]The results are robust to alternative return measures (SAR, FF4, FF6, HMXZ5, and DHS3).

bottom CNN buy probability quintiles ranges from 1.6% to 5.4% (t-statistics between 2.711 and 5.954) across all EA quintiles. In Panel C, we find similar evidence: the positive relation between the *CNN buy* probability and 63-day MAR is not limited to any TREND quintiles. In particular, the hedge return based on the *CNN buy* probability appears to be larger in magnitude in the low and high quintiles of EA and TREND. We also find that the *CNN buy* features subsume some return predictability of EA and TREND. Turning to Panels D, E, F, and G we find that the average difference in 63-day MAR between the highest and lowest *CNN buy* probability quintiles remains positive and statistically significant at the 1% level across all RET[−1,1], BM, GP, and AG quintiles.

We report the double sorts results based on the other eight characteristics in Table IA1 of the Internet Appendix, and find that the drift-predicting power of *CNN buy* features persists in all the quintiles of the other seven characteristics. The results in Table 4 and Table IA1 combined indicate that CNN's ability to predict post-earnings announcement drift is mostly distinct from that of the usual determinants.

4.3 Cross-sectional regression

We next perform a cross-sectional regression analysis to simultaneously control for the firm characteristics that may affect the positive relation between the *CNN buy* probability and post-earnings announcement drift. Following prior literature (Akbas [2016]), we estimate quarterly weighted Fama and MacBeth [1973] regressions in which the dependent variable is the firm's 63-day MAR. We begin by first running the following cross-sectional regression every quarter:

$$\text{MAR}_{i,q+1} = \alpha_q + \beta_q \text{CNN buy probability}_{i,q} + \sum \beta_{c,q} \text{Controls}_{i,q} + \varepsilon_{i,q+1}, \quad (7)$$

where i refers to the stock, q refers to the quarter, and the *CNN buy* probability and control variables are converted into scaled ranks ranging from −0.5 to 0.5 with a mean of zero. Then, we average the cross-sectional coefficients across all quarters, where the weights correspond to the number of observations in each quarterly cross-sectional regression. In addition to using MAR as the dependent variable, we also employ SAR and four factor-adjusted buy-and-hold returns (FF4, FF6, HMXZ5, and DHS3).

Table 5 presents the regression results. The coefficient on the *CNN buy* probability in Column 1 is 0.014 (*t*-statistic = 3.529), suggesting that a long-short strategy of going long on the highest *CNN buy* probability decile and short on the lowest decile generates an incremental 63-day MAR of around 1.4%, controlling for other anomalies. In columns 2 to 5 where we replace MAR with SAR and factor-adjusted buy-and-hold returns, the coefficients on the *CNN buy probability* range from 0.010 to 0.014 and are all statistically significant at the 1% level, suggesting that our results are not caused by omission of risk factors. Thus, although the CNN is picking up some of the features that relate to existing anomalies, there remains and significant portion of the future abnormal returns that are incremental to the set of risk factors and known anomalies.

We also find that in all model specifications, the post-earnings announcement drift is significantly increasing in earnings announcement return (RET[−1, 1]) and operating profitability (OP) while decreasing in market capitalization (SIZE).[26] Overall, the results in Table 5 provide strong support for the out-of-sample drift-predicting power of the *CNN buy* features, which is distinct from existing stock return stock anomalies and risk factors.

[26]While SUE and RET[−1,1] both proxy for earnings surprises, their coefficients remain positively significant, consistent with Kishore et al. [2008]'s findings that trading strategies formed based on SUE and RET[−1,1] are largely independent of each other.

5 Determining what information the CNN is using

Having demonstrated that *CNN* recommendations possess significant drift-predicting power, in this section we try to better understand what information is being used by the CNN. We first explore the nature of these *CNN buy* features via linear approximation. Then, we examine whether *CNN buy* features exhibit incremental predictive ability for future earnings growth, and whether the drift-predicting power of *CNN buy* features can be attributable to market investors missing such predictive ability.

5.1 Linear approximation of CNN predictions

We begin by linearly fitting CNN predictions with firm characteristics and historical earnings. In Table 6 we estimate quarterly weighted Fama and MacBeth [1973] regressions in the out-of-sample period (1994Q3-2023Q2, 116 quarters) using *CNN buy* probability as the dependent variable. The independent variables are the 14 anomalies considered before in specification 1 and earnings in the most recent eight quarters in specification 2.[27]

In specification 1, we find that CNN buy probability is positively correlated with standardized unexpected earnings (SUE), earnings acceleration (EA), earnings announcement return (RET[−1, 1]), pre-announcement return (RET[−30, −2]), earnings volatility (VOL), book-to-market ratio (BM), gross profitability (GP), and asset growth (AG), while negatively related to earnings persistence (PERSIST), market capitalization (SIZE), operating profitability (OP), and total accruals (TA). The result suggests the CNN model is capable of discerning some meaningful return-predicting information, such as the earnings surprise effect, the SUE effect, the earnings

[27]We follow Ball et al. [2009] to use return on assets (ROA) as the earnings measure, where ROA is defined as the quarterly earnings scaled by total assets in the previous quarter.

acceleration effect, the gross profitability effect, the value effect, and the size effect solely from historical earnings represented in the form of images. However, the *CNN buy* probability appears to load negative (positively) on operating profitability (asset growth) despite the fact that it has been documented to positively (negatively) predict subsequent returns.

Economically, the coefficient on SUE (= 0.510) is the largest and exceeds the second-largest coefficient (= −0.274) on SIZE by around a half. Moving from the lowest decile to the highest decile of SUE (SIZE) is associated with a 51.0% (27.4%) incremental increase (decrease) in the decile of *CNN buy* probability. Overall, the 14 firm characteristics collectively explain 32.8% of the variation in *CNN buy* probability, indicating most of the variation in *CNN buy* features is left unexplained.

Turning to specification 2, we find that the *CNN buy* probability is most associated with the current quarter's ROA and the ROA four quarters prior in terms of economic magnitude. Moving from the lowest decile to the highest decile of ROA in the current quarter (four quarters prior) leads to a 64.4% (42.8%) incremental increase (decrease) in the decile of *CNN buy* probability. This result to some extent explains why the *CNN buy* features most resemble SUE among all the considered firm characteristics in specification 1. The eight historical earnings collectively explain 30% of the variation in *CNN buy* probability, implying that 70% of the variation in *CNN buy* probability is attributable to the nonlinear transformation of the underlying historical earnings.

In order to provide more insights on the reasoning behind CNN predictions, Figure 8 displays earnings images whose *CNN buy* probabilities rank in the top and bottom 15 among the 240,844 earnings images in the out-of-sample period (1994Q3 to 2023 Q2, 116 quarters). We find that earnings images with the highest *CNN buy* probabilities all have the current earnings as the maximum earnings and mostly have the earnings four quarters prior as the minimum earnings. In addition, there seems to be an increasing trend of quarterly earnings. In contrast, we see that earnings images with the lowest *CNN buy* probabilities mostly have the current earnings as the minimum earnings, and their earnings in the previous two to four quarters are relatively high.

5.2 CNN predictions and future earnings growth

In this section, we examine whether the CNN buy features possess incremental predictive ability for future earnings growth and whether the post-announcement abnormal return based on CNN buy probability is associated with this predictive ability. Since the earnings images in Section 3 are plotted in a way similar to what investors would see during earnings call conferences or generate on their own, we conjecture that the return-predicting power of CNN is because investors do not fully incorporate the implications of past earnings for future earnings growth.

First, we run a regression of one-quarter-ahead earnings growth on *CNN buy* probability (CNNBP) in the out-of-sample period (1994Q3-2023Q2, 116 quarters). We use the previously defined SUE as the earnings growth measure (Bernard and Thomas [1989], Ball and Bartov [1996]). In particular, we have

$$\mathrm{SUE}_{q+1} = \alpha + \gamma_1 \text{CNN buy probability}_q + \sum \gamma_c \text{Controls}_q + \delta_{q+1}, \tag{8}$$

where SUE_{q+1} represents one-quarter-ahead earnings growth. Control variables are the 14 firm characteristics considered before.[28] In columns 1 and 2 of Table 7, the coefficients on CNNBP are positive and statistically significant at the 1% level, suggesting that *CNN buy* probability is a significant predictor of earnings growth in the

[28] In particular, SUE_q serves as the control for the well-documented earnings autocorrelation pattern in prior studies.

subsequent quarter. Economically, moving from the lowest decile to the highest decile of CNNBP in the current quarter leads to a 13.3% incremental increase in the decile of one-quarter-ahead earnings growth, controlling for past earnings growth and other firm characteristics. On the other hand, the coefficients on SUE are significantly positive, consistent with the previous findings in the literature.

We next examine whether the *CNN buy* probability (CNNBP) helps predict the three-day abnormal return around the one-quarter-ahead earnings announcement ($\mathrm{RET}[-1,1]_{q+1}$).[29] If this is the case, then investors do not appear to incorporate fully the implications of earnings acceleration for earnings announcement. We find that in columns 3 and 4 of Table 7, the coefficients on CNNBP are positive and highly significant, indicating a positive relation between *CNN buy* probability and the three-day abnormal return around the next earnings announcement date. In terms of economic magnitude, moving from the lowest decile to the highest decile of CNNBP in the current quarter leads to a 0.4% incremental increase in $\mathrm{RET}[-1,1]_{q+1}$.

Since *CNN buy* features have positive implications for both one-quarter-ahead earnings growth and for the three-day abnormal return surrounding the next earnings announcement date, we use the Mishkin test (Mishkin [1983], Abel and Mishkin [1983]) which is widely used in the earnings-based anomaly literature, to test whether the market fully understands the implications of the *CNN buy* features for SUE_{q+1}.[30] This involves simultaneously estimating two equations: an earnings forecasting equation and a rational pricing equation. In our context, the earnings forecasting equation is equation (8) that characterizes the evolution of earnings growth.

For the rational pricing equation, we assume a linear abnormal return (AR) model

[29]Shorter-window returns are typically less susceptible to risk considerations (Bernard and Thomas [1990], Sloan [1996], Narayanamoorthy [2006], Cao and Narayanamoorthy [2012])

[30]See, for example, Sloan [1996], Dechow and Sloan [1997], Rangan and Sloan [1998], Collins and Hribar [2000], Narayanamoorthy [2006], Cao and Narayanamoorthy [2012], Chen and Shane [2014], Hui et al. [2016], Ma and Markov [2017], and He and Narayanamoorthy [2020].

(e.g., Sloan [1996]) that satisfies the efficient-markets condition:

$$\mathrm{AR}_{q+1} = \beta(\mathrm{SUE}_{q+1} - \mathrm{SUE}^{e}_{q+1}) + \varepsilon_{q+1}, \tag{9}$$

where β is a multiple, $\mathrm{SUE}^{e}_{q+1} = E_q(\mathrm{SUE}_{q+1})$ is the rational forecast of SUE_{q+1} in quarter q, and ε_{q+1} is a noise in quarter $q+1$ satisfying $E_q(\varepsilon_{q+1}) = 0$. In equation (9), abnormal returns are zero in expectation, i.e., $E_q(\mathrm{AR}_{q+1}) = 0$, and market efficiency implies that only$(\mathrm{SUE}_{q+1} - \mathrm{SUE}^{e}_{q+1})$, the unanticipated changes in SUE, can be correlated with AR_{q+1}. In other words, if the market correctly understands the implications of the *CNN buy* features for future earnings growth as depicted in equation (8), AR_{q+1} should only be related to the earnings growth surprise $(\mathrm{SUE}_{q+1} - \mathrm{SUE}^{e}_{q+1} = \delta_{q+1})$, but not related to the *CNN buy* probability in quarter q.

Combining the earnings growth forecasting model in equation (8) with the rational pricing model in equation (9) provides the following system:

$$\text{Forecasting equation: } \mathrm{SUE}_{q+1} = \alpha + \gamma_1 \mathrm{CNNBP}_q + \sum \gamma_c \mathrm{Controls}_q + \delta_{q+1} \tag{10}$$

$$\text{Pricing equation: } \mathrm{AR}_{q+1} = \beta(\mathrm{SUE}_{q+1} - \alpha^* - \gamma_1^* \mathrm{CNNBP}_q - \sum \gamma_c^* \mathrm{Controls}_q) + \varepsilon_{q+1}. \tag{11}$$

The two systems are simultaneously estimated using iterative-weighted non-linear least squares (Mishkin [1983]), and the coefficients with * represent the coefficients inferred from market investors' expectation of SUE_{q+1}.[31] In particular, we are interested in testing whether $\gamma_1 = \gamma_1^*$ holds or not, i.e., whether the observed relation between SUE_{q+1} and *CNN buy* features is the same as the relation between SUE_{q+1} and *CNN*

[31]Kraft et al. [2007] shows that the exclusion of control variables from the forecasting and pricing equations leads to an omitted variables problem. That is, if the variables omitted are not rationally priced and are also correlated with the variable of interest in the forecasting equation, then the source of market inefficiency cannot be correctly identified. Hence, we include various control variables that may be related to *CNN buy* probability.

buy features implicit in AR_{q+1}. In other words, $\gamma_1 = \gamma_1^*$ indicates that investors are fully aware of the implications of *CNN buy* features for SUE_{q+1}, and this restriction yields a likelihood ratio test statistic that has a chi-square distribution with one degree of freedom.[32] If $\gamma_1 = \gamma_1^*$ is rejected while $0 < \gamma_1^* < \gamma_1$, then investors only partially incorporate the implications of *CNN buy* features for future earnings growth. On the other hand, if $\gamma_1 = \gamma_1^*$ is rejected and $\gamma_1^* = 0$, then investors appear to completely ignore the implications of *CNN buy* features for future earnings growth.

We report the estimated coefficients, *t*-statistics based on firm and quarter double-clustered standard errors, and likelihood ratio test statistics of the Mishkin test in Table 8. In particular, AR_{q+1} is either the abnormal return from a three-day window around quarter $q + 1$'s earnings or the quarter-long window starting two days after the quarter q earnings and ending on the next announcement date. All variables except for the abnormal return AR are converted into scaled ranks ranging from −0.5 to 0.5 with a mean of 0.

Since $\gamma_1 = \gamma_1^*$ is rejected at the 1% level (likelihood ratio statistic 8.218 for the three-day window and 21.444 for the quarter-long window) and since $\gamma_1 > \gamma_1^*$, it appears that market investors are underestimating the implications of *CNN buy* features for future earnings growth. In particular, the quarter-long window γ_1^* (= 0.010) is statistically indistinguishable from zero[33] while the three-day window γ_1^* (= 0.062) is highly significant, implying that market investors are completely ignoring the positive implications of *CNN buy* features for future earnings growth at the time of the current

[32] The test statistic of the Mishkin test is $2 \times n \times \ln{(SSR^c/SSR^u)}$ distributed asymptotically $\mathcal{X}^2(q)$, where q is the number of constraints imposed by market efficiency, n is the number of observations in each equation, SSR^c is the sum of squared residuals from the constrained weighted system, and SSR^u is the sum of squared residuals from the unconstrained weighted system.

[33] In untabulated tests we use SAR_{q+1} and four factor-adjusted buy-and-hold returns (FF4_{q+1}, FF6_{q+1}, HMXZ5_{q+1}, and DHS3_{q+1}) to measure quarter-long AR_{q+1}, respectively. We find that in all specifications, $\gamma_1 = \gamma_1^*$ is rejected at the 1% level, and γ_1 is statistically indistinguishable from zero. Hence, the fact that market investors are completely unaware of the implications of *CNN buy* features can not be attributed to lack of risk controls.

earnings announcement, but partially understands these implications by the time of the next earnings announcement. In other words, the market gradually learns more about the implications of *CNN buy* features for future earnings growth from other sources of information by the time of the next earnings announcement.

Overall, the results in this section provide evidence that the positive relation between the *CNN buy* probability and post-earnings announcement drift is consistent with market investors not fully understanding the implications of the *CNN buy* features for one-quarter-ahead earnings growth.

6 Robustness

6.1 Monthly rebalancing trading strategy

The out-of-sample tests in Table 2 involves buying and selling stocks two days after an earnings announcement, which requires significant attention and thus may be difficult to implement in reality. Hence, in this section we examine whether sorting stocks based on the *CNN buy* probability to form a more conservative month-based rebalancing trading strategy (Hou et al. [2020], Jensen et al. [2023]) can still generate profits.

In particular, at the end of each month t in the out-of-sample period, we sort firms into deciles based on the *CNN buy* probability computed using the most recent eight quarterly earnings. For a firm to enter the portfolio formation at the end of month t, we require that announcement date of the most recent earnings to be within three months prior to portfolio formation to exclude stale earnings information. We then examine the average returns in the subsequent month $t+1$ for each *CNN buy* probability decile.

Table 9 presents the equal-weighted and value-weighted average portfolio returns for each *CNN buy* probability decile. Panel A shows that a hedge portfolio going long in the top CNN-based buy probability decile and short in the bottom decile yields an average equal-weighted monthly return of 1.0%. The factor-adjusted hedge returns range from 0.7% to 0.8% and are all statistically significant at the 1% level. On the other hand, the average value-weighted hedge returns are significantly positive but are smaller in magnitude (ranging from 0.3% to 0.5%). This is because CNN model is treating each input image equally during the training phase, regardless of market capitalization. Hence, CNN predictions are unsurprisingly more accurate when we employ CNN stored parameters to form out-of-sample portfolios with equal-weights rather than value-weights.

6.2 Alternative CNN modeling choices

We next explore whether the main results in Table 2 are sensitive to model specifications. In particular, we re-train the CNN model with alternative modeling choices, as listed in the first column of Table 10, and then examine the return differential in the 63-day post-announcement BHAR between the highest and lowest *CNN buy* probability deciles in the out-of-sample period.

In Panel A, we experiment with different combinations of filter size and number of convolution layers. The combination of our CNN model can be expressed as (7 × 7, 3 × 3, 3 × 3), while we consider alternative modeling choices of (5 × 5, 3 × 3, 3 × 3), (5 × 5, 3 × 3, 3 × 3), (7 × 3, 3 × 3), (5 × 5, 3 × 3), and (3 × 3, 3 × 3). We find that the 63-day MAR, SAR, and factor-adjusted returns remain positive and highly significant, indicating that CNN model performance is mostly insensitive to those choices. In addition, omitting the batch normalization step or Xavier initialization, adjusting the activation function from leaky ReLU to ReLU, or lowering the dropout rate from 0.5 to 0 does not generate a noticeable loss in performance either. Hence, our main results are robust to alternative modeling choices.

In Panel B, we employ a one-dimensional CNN model in training where the inputs are 1 × 8 pixels row vectors consisting of the time-series of firms' most recent eight quarterly earnings (in the form of ROA) as inputs, and the convolutional filters sliding across the inputs are row vectors as well.[34] In other words, the one-dimensional CNN model is a special case of the two-dimensional CNN model, with both the inputs and the convolutional filters shrinking from matrices to row vectors. In particular, we consider the following modeling choices: (1 × 7, 1 × 3, 1 × 3), (1 × 5, 1 × 3, 1 × 3), (1 × 3, 1 × 3, 1 × 3), (1 × 7, 1 × 3), (1 × 5, 1 × 3), and (1 × 3, 1 × 3), and find that the 63-day MAR, SAR, and factor-adjusted returns are statistically indistinguishable from zero. The only exception is when we consider a modeling choice of (1 × 7, 1 × 3, 1 × 3), but the magnitude of the return differences is less than one-third of that in Panel A. The results suggest that image representation of historical earnings produces more information useful in prediction post-earnings announcement drift.

7 Conclusion

Our research shows that applying AI to visualized earnings images of an earnings time series can predict post-earnings announcement drift in a manner that is orthogonal to existing anomalies and not accounted for by risk. In out-of-sample tests, we find that firms in the highest *CNN buy* probability decile significantly outperform firms in the lowest *CNN buy* probability decile by 3.6% in the 63-day post-announcement window.

[34] The results are robust to using unscaled earnings numbers.

In addition, the drift-predicting power of *CNN buy* features is robust to a battery of controls for risk, distinct from that of the previously documented anomalies and earnings attributes, and stable over time.

Although the *CNN buy probability* overlaps somewhat with existing anomalies, a lot of the variation is largely left unexplained. In particular, the *CNN buy probability* appear to be positively associated with one-quarter-ahead earnings growth as well as the three-day abnormal return surrounding the next earnings announcement. As a result, we employ a direct market efficiency test and find that high abnormal returns following high *CNN buy* probability and the positive implications of *CNN buy* features for future earnings growth are strongly associated. In other words, the drift-predicting power of *CNN buy* features is consistent with investors not incorporating fully the implications of *CNN buy* features for future earnings growth.

In addition, the drift-predicting ability of *CNN buy* features persists in a more conservative monthly-rebalancing strategy setting, and remains insensitive to various model specifications when image representation is used. Overall, our paper highlights the usefulness of applying deep learning techniques to visualized data as a way to identify future earnings growth and abnormal returns.

References

Abel, A. B. and F. S. Mishkin (1983). On the econometric testing of rationality-market efficiency. *The Review of Economics and Statistics*, 318–323.

Akbas, F. (2016). The calm before the storm. *The Journal of Finance 71* (1), 225–266.

Akbas, F., C. Jiang, and P. D. Koch (2017). The trend in firm profitability and the cross-section of stock returns. *The Accounting Review 92* (5), 1–32.

Bali, T. G., H. Beckmeyer, M. Moerke, and F. Weigert (2023). Option return predictability with machine learning and big data. *The Review of Financial Studies 36* (9), 3548–3602.

Ball, R. and E. Bartov (1996). How naive is the stock market's use of earnings information? *Journal of Accounting and Economics 21* (3), 319–337.

Ball, R. and P. Brown (1968). An empirical evaluation of accounting income numbers. *Journal of Accounting Research 6* (2), 159–178.

Ball, R., J. Gerakos, J. T. Linnainmaa, and V. Nikolaev (2016). Accruals, cash flows, and operating profitability in the cross section of stock returns. *Journal of Financial Economics 121* (1), 28–45.

Ball, R., G. Sadka, and R. Sadka (2009). Aggregate earnings and asset prices. *Journal of Accounting Research 47* (5), 1097–1133.

Bao, Y., B. Ke, B. Li, and J. Yu. (2020). Detecting Accounting Fraud in Publicly Traded U.S. Firms Using a Machine Learning Approach. *Journal of Accounting Research* 58(1): 199-235.

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

Bernard, V. L. and J. K. Thomas (1990). Evidence that stock prices do not fully reflect the implications of current earnings for future earnings. *Journal of accounting and economics 13* (4), 305–340.

Bianchi, D., M. Büchner, and A. Tamoni (2021). Bond risk premiums with machine learning. *The Review of Financial Studies 34* (2), 1046–1089.

Brown, N. R. Crowley, B. Elliott (2020) What are you saying? Using topic to Detect Financial Misreporting. *Journal of Accounting Research* 58(1): 237-291.

Büchner, M. and B. Kelly (2022). A factor model for option returns. *Journal of Financial Economics 143* (3), 1140–1161.

Cao, S., Y. Cheng, M. Wang, Y. Xia, and B. Yang (2024). Visual information and ai divide: Evidence from corporate executive presentations. *Working paper* .

Cao, S., W. Jiang, J. Wang, and B. Yang (2024). From man vs. machine to man+ machine: *The art and ai of stock analyses. Journal of Financial Economics 160*, 103910.

Cao, S. S. and G. S. Narayanamoorthy (2012). Earnings volatility, post–earnings announcement drift, and trading frictions. *Journal of Accounting Research 50* (1), 41–74.

Carhart, M. M. (1997). On persistence in mutual fund performance. *The Journal of Finance 52* (1), 57–82.

Chan, L. K., N. Jegadeesh, and J. Lakonishok (1996). Momentum strategies. *The Journal of Finance 51* (5), 1681–1713.

Chen, J. Z. and P. B. Shane (2014). Changes in cash: persistence and pricing implications. *Journal of Accounting Research 52* (3), 599–634.

Chen, L., M. Pelger, and J. Zhu (2024). Deep learning in asset pricing. *Management Science 70* (2), 714–750.

Chen, X. Y. Cho, Y. Dou, and B. Lev (2022). Predicting Earnings Changes using Machine Learning and Detailed Financial Data. Journal of Accounting Research 60(2): 467-515.

Christensen, T. E., K. E. Fronk, J. A. Lee, and K. K. Nelson (2024). Data visualization in 10-k filings. *Journal of Accounting and Economics 77* (2-3), 101631.

Collins, D. W. and P. Hribar (2000). Earnings-based and accrual-based market anomalies: one effect or two? *Journal of Accounting and Economics 29* (1), 101–123.

Cooper, M. J., H. Gulen, and M. J. Schill (2008). Asset growth and the cross-section of stock returns. *the Journal of Finance 63* (4), 1609–1651.

Daniel, K., D. Hirshleifer, and L. Sun (2020). Short-and long-horizon behavioral factors. *The Review of Financial Studies 33* (4), 1673–1736.

Dechow, P. M. and R. G. Sloan (1997). Returns to contrarian investment strategies: Tests of naive expectations hypotheses. *Journal of Financial Economics 43* (1), 3–27.

DeMiguel, V., J. Gil-Bazo, F. J. Nogales, and A. A. Santos (2023). Machine learning and fund characteristics help to select mutual funds with positive alpha. *Journal of Financial Economics 150* (3), 103737.

Engelberg, J., C. Sasseville, and J. Williams (2012). Market madness? the case of mad money. *Management Science 58* (2), 351–364.

Fama, E. F. and K. R. French (1992). The cross-section of expected stock returns. *the Journal of Finance 47* (2), 427–465.

Fama, E. F. and K. R. French (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics 33* (1), 3–56.

Fama, E. F. and K. R. French (2015). A five-factor asset pricing model. *Journal of Financial Economics 116* (1), 1–22.

Fama, E. F. and K. R. French (2018). Choosing factors. *Journal of Financial Economics 128* (2), 234–252.

Fama, E. F. and J. D. MacBeth (1973). Risk, return, and equilibrium: Empirical tests. *Journal of Political Economy 81* (3), 607–636.

Feng, G., S. Giglio, and D. Xiu (2020). Taming the factor zoo: A test of new factors. *The Journal of Finance 75* (3), 1327–1370.

Foster, G., C. Olsen, and T. Shevlin (1984). Earnings releases, anomalies, and the behavior of security returns. *The Accounting Review ,* 574–603.

Francis, J., R. LaFond, P. M. Olsson, and K. Schipper (2004). Costs of equity and earnings attributes. *The Accounting Review 79* (4), 967–1010.

Frank, M. Z. and A. Sanati (2018). How does the stock market absorb shocks? *Journal of Financial Economics 129* (1), 136–153.

Freyberger, J., A. Neuhierl, and M. Weber (2020). Dissecting characteristics nonparametrically. *The Review of Financial Studies 33* (5), 2326–2377.

Garfinkel, J. A. and J. Sokobin (2006). Volume, opinion divergence, and returns: A study of post–earnings announcement drift. *Journal of Accounting Research 44* (1), 85–112.

Glorot, X. and Y. Bengio (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256. JMLR Workshop and Conference Proceedings.

Gu, M., S. H. Teoh, and S. Wu (2024). Gif sentiment and stock returns. *Working paper .*

Gu, S., B. Kelly, and D. Xiu (2020). Empirical asset pricing via machine learning. *The Review of Financial Studies 33* (5), 2223–2273.

Gu, S., B. Kelly, and D. Xiu (2021). Autoencoder asset pricing models. *Journal of Econometrics 222* (1), 429–450.

He, K., X. Zhang, S. Ren, and J. Sun (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

He, S. and G. G. Narayanamoorthy (2020). Earnings acceleration and stock returns. *Journal of Accounting and Economics 69* (1), 101238.

Hou, K., H. Mo, C. Xue, and L. Zhang (2021). An augmented q-factor model with expected growth. Review of Finance 25 (1), 1–41.

Hou, K., C. Xue, and L. Zhang (2015). Digesting anomalies: An investment approach. *The Review of Financial Studies 28* (3), 650–705.

Hou, K., C. Xue, and L. Zhang (2020). Replicating anomalies. *The Review of Financial Studies 33* (5), 2019–2133.

Hribar, P. and D. W. Collins (2002). Errors in estimating accruals: Implications for empirical research. *Journal of Accounting Research 40* (1), 105–134.

Hu, A. and S. Ma (2023). Persuading investors: A video-based study. *forthcoming*, *The Journal of Finance.*

Huang, G., Z. Liu, L. Van Der Maaten, and K. Q. Weinberger (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708.

Hui, K. W., K. K. Nelson, and P. E. Yeung (2016). On the persistence and pricing of industry-wide and firm-specific earnings, cash flows, and accruals. *Journal of Accounting and Economics 61* (1), 185–202.

Ioffe, S. and C. Szegedy (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pp. 448–456. pmlr.

Jegadeesh, N. and J. Livnat (2006). Post-earnings-announcement drift: The role of revenue surprises. *Financial Analysts Journal 62* (2), 22–34.

Jensen, T. I., B. Kelly, and L. H. Pedersen (2023). Is there a replication crisis in finance?

The Journal of Finance 78 (5), 2465–2518.

Jiang, J., B. Kelly, and D. Xiu (2023). (re-) imag (in) ing price trends. *The Journal of Finance 78* (6), 3193–3249.

Kaniel, R., Z. Lin, M. Pelger, and S. Van Nieuwerburgh (2023). Machine-learning the skill of mutual fund managers. *Journal of Financial Economics 150* (1), 94–138.

Kapadia, N. and M. Zekhnini (2019). Do idiosyncratic jumps matter? *Journal of Financial Economics 131* (3), 666–692.

Kausar, A. (2018). Post-earnings-announcement drift and the return predictability of earnings levels: one effect or two? *Management Science 64* (10), 4877–4892.

Kelly, B., D. Palhares, and S. Pruitt (2023). Modeling corporate bond returns. *The Journal of Finance 78* (4), 1967–2008.

Kelly, B. T., S. Pruitt, and Y. Su (2019). Characteristics are covariances: A unified model of risk and return. *Journal of Financial Economics 134* (3), 501–524.

Kim, A. M. Muhn, V. Nikolaev (2024) Financial Statement Analysis with Large Language Models. *Booth School of Business Working Paper*.

Kingma, D. P. and J. Ba (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

Kishore, R., M. W. Brandt, P. Santa-Clara, and M. Venkatachalam (2008). Earnings announcements are full of surprises. *Working paper*.

Kozak, S., S. Nagel, and S. Santosh (2020). Shrinking the cross-section. *Journal of Financial Economics 135* (2), 271–292.

Kraft, A., A. J. Leone, and C. E. Wasley (2007). Regression-based tests of the market pricing of accounting numbers: The mishkin test and ordinary least squares. *Journal of Accounting Research 45* (5), 1081–1114.

Leippold, M., Q. Wang, and W. Zhou (2022). Machine learning in the chinese stock market. *Journal of Financial Economics 145* (2), 64–82.

Liu, J., R. F. Stambaugh, and Y. Yuan (2018). Absolving beta of volatility's effects. *Journal of Financial Economics 128* (1), 1–15.

Livnat, J. and R. R. Mendenhall (2006). Comparing the post–earnings announcement drift for surprises calculated from analyst and time series forecasts. *Journal of Accounting Research 44* (1), 177–205.

Ma, G. and S. Markov (2017). The market's assessment of the probability of meeting or beating the consensus. *Contemporary Accounting Research 34* (1), 314–342.

Maas, A. L., A. Y. Hannun, A. Y. Ng, et al. (2013). Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, Volume 30, pp. 3. Atlanta, GA.

Mishkin, F. S. (1983). Are market forecasts rational? In *A Rational Expectations Approach to Macroeconometrics*, pp. 59–75. National Bureau of Economic Research Monograph. Chicago, IL: The University of Chicago Press.

Moss, A. (2022). How do brokerages' digital engagement practices affect retail investor information processing and trading? *Ph.D. thesis, The University of Iowa*.

Murray, S., Y. Xia, and H. Xiao (2024). Charting by machines. *Journal of Financial Economics 153*, 103791.

Narayanamoorthy, G. (2006). Conservatism and cross-sectional variation in the post–earnings announcement drift. *Journal of Accounting Research 44* (4), 763–789.

Nekrasov, A., S. H. Teoh, and S. Wu (2022). Visuals and attention to earnings news on twitter. *Review of Accounting Studies 27* (4), 1233–1275.

Newey, W. K. and K. D. West (1987). A simple, positive semi-definite, heteroskedasticity and autocorrelation. *Econometrica 55* (3), 703–708.

Novy-Marx, R. (2013). The other side of value: The gross profitability premium. *Journal of Financial Economics 108* (1), 1–28.

Obaid, K. and K. Pukthuanthong (2022). A picture is worth a thousand words: Measuring investor sentiment by combining machine learning and photos from news. *Journal of Financial Economics 144* (1), 273–297.

Rangan, S. and R. G. Sloan (1998). Implications of the integral approach to quarterly reporting for the post-earnings-announcement drift. *The Accounting Review* , 353–371.

Rapach, D. E., J. K. Strauss, and G. Zhou (2013). International stock return predictability: What is the role of the united states? *The Journal of Finance 68* (4), 1633–1662.

Richardson, S. A., R. G. Sloan, M. T. Soliman, and I. Tuna (2005). Accrual reliability, earnings persistence and stock prices. *Journal of Accounting and Economics 39* (3), 437–485.

Savor, P. G. (2012). Stock returns after major price shocks: The impact of information.

Journal of Financial Economics 106 (3), 635–659.

Simonyan, K. and A. Zisserman (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* .

Sloan, R. G. (1996). Do stock prices fully reflect information in accruals and cash flows about future earnings? *The Accounting Review* , 289–315.

Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov (2014). Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research 15* (1), 1929–1958.

Szegedy, C., W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9.

Vega, C. (2006). Stock price reaction to public and private information. *Journal of Financial Economics 82* (1), 103–133.

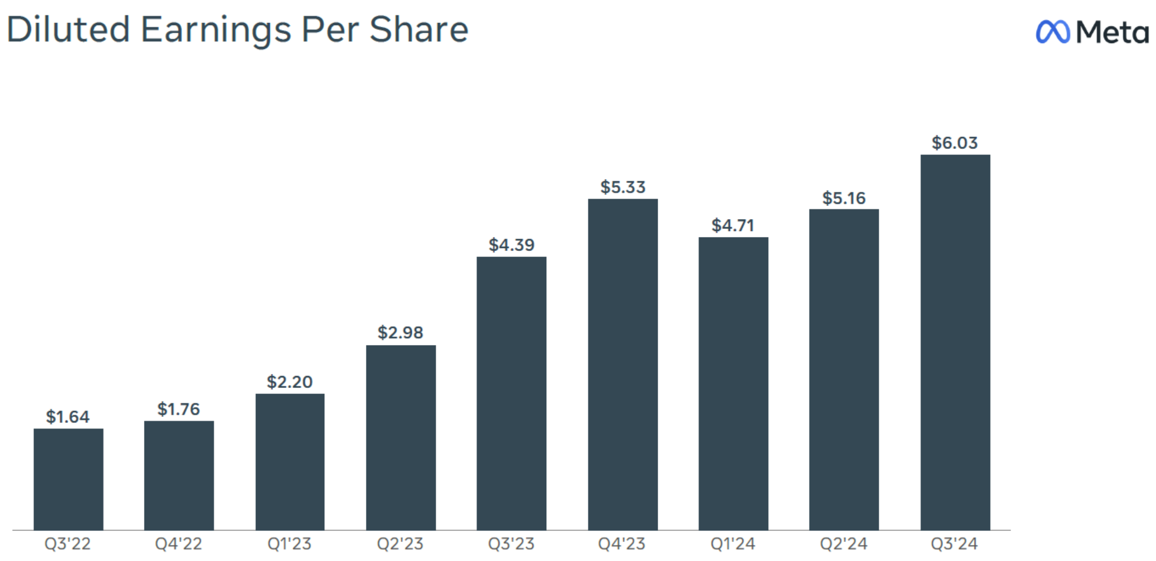
Wu, W., J. Chen, Z. Yang, and M. L. Tindall (2021). A cross-sectional machine learning approach for hedge fund return prediction and selection. *Management Science 67* (7), 4577–4601.

Appendix. Variable Definitions. This table summarizes variable definitions. Compustat quarterly/annual items are colored in blue.

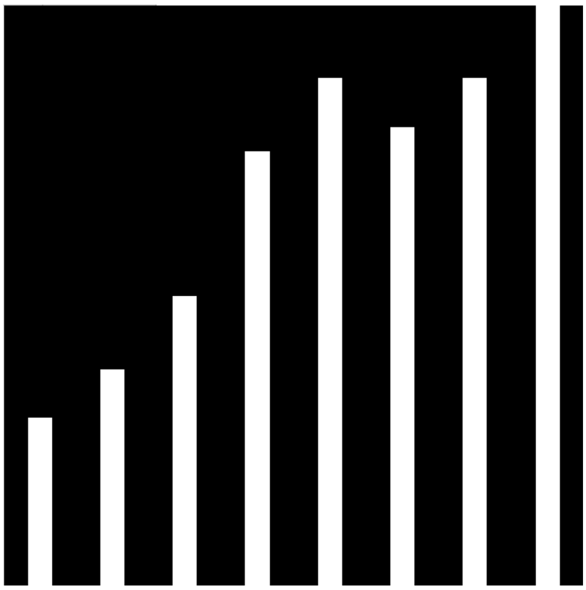
Variables	Descriptions
MAR	Market-adjusted return (MAR) is defined as the difference between the buy-and-hold return of an announcing firm and that of the CRSP value-weighted market portfolio over the 63-day windows [2, 64] following its earnings announcement date.
SAR	Size-adjusted return (SAR) is defined as the difference between the buy-and-hold return of an announcing firm and that of a size-matched portfolio over the 63-day window ([2, 64]) following its earnings announcement date. We use the monthly NYSE size decile breakpoints at the end of June in year t to determine the size-matched portfolio for a firm whose earnings announcement date is between July of year t to June of year $t+1$.
FF4	Fama-French three-factor and momentum-adjusted buy-and-hold return during the 63-day window ([2, 64]) following earnings announcement date, with factor loadings estimated using the 120-day window ([-150, -31], 90 days minimum) prior to the earnings announcement date. The factors are market, size, value, and momentum.
FF6	Fama-French five-factor and momentum-adjusted buy-and-hold return during the 63-day window ([2, 64]) following earnings announcement date, with factor loadings estimated using the 120-day window ([-150, -31], 90 days minimum) prior to the earnings announcement date. The factors are market, size, value, operating profitability, investment, and momentum.
HMXZ5	q^5-factor-adjusted buy-and-hold return during the 63-day window ([2, 64]) following earnings announcement date, with factor loadings estimated using the 120-day window ([-150, -31], 90 days minimum) prior to the earnings announcement date. The factors are market, size, investment, return on equity, and expected growth.
DHS3	Behavioral-factor-adjusted buy-and-hold return during the 63-day window ([2, 64]) following earnings announcement date, with factor loadings estimated using the 120-day window ([-150, -31], 90 days minimum) prior to the earnings announcement date. The factors are market, financing, and post earnings announcement drift.
SUE	Standardized unexpected earnings, defined as the change in split-adjusted quarterly earnings per share ($\frac{\text{EPSPXQ}}{\text{AJEXQ}}$) from its value four quarters ago divided by the standard deviation of this change over the prior eight quarters (six quarters minimum). SUE also serves as the earnings growth proxy.

Variables	Descriptions
EA	Earnings acceleration. For firm i in quarter q, we use $$\frac{\text{EPS}_{i,q}-\text{EPS}_{i,q-4}}{\text{Stock Price}_{i,q-1}}-\frac{\text{EPS}_{i,q-1}-\text{EPS}_{i,q-5}}{\text{Stock Price}_{i,q-2}},$$ where $\text{EPS}_{i,q}$ is earnings per share for firm i in quarter q. Shares are adjusted for stock splits.
TREND	Trend in quarterly gross profitability. For firm i in quarter q, we use $\beta_{i,q}$ estimated from the following time-series regression: $$GPQ_{i,q}=\alpha_{i,q}+\beta_{i,q}t+\lambda_{1,i,q}D1+\lambda_{2,i,q}D2+\lambda_{3,i,q}D3+\epsilon_{i,q}\ ,$$ where $t=1,2,\ldots,8$ and represents a deterministic time trend covering quarter $q-7$ through q, and D1 to D3 represent quarterly dummy variables. GPQ is calculated as sales revenue (SALEQ) minus costs of goods sold (COGSQ), divided by total assets (ATQ). If SALEQ is unavailable, we use quarterly revenue (REVTQ). If COGSQ is unavailable, we use quarterly total operating expenses (XOPRQ) minus quarterly selling, general and administrative expenses (XSGAQ, zero if missing).
RET[−1, 1]	Earnings announcement return, defined as the value-weighted market-adjusted stock return during the [−1, 1] window around earnings announcement date.
RET[−30, -2]	Pre-announcement return, defined as the value-weighted market-adjusted stock return during the [−30, −2] window prior to earnings announcement date.
PERSIST	Earnings persistence. For firm i in quarter q, we use $\beta_{i,q}$ estimated from the following time-series regression: $$\text{EARNINGS}_{i,q}=\alpha_{i,q}+\beta_{i,q}\text{EARNINGS}_{i,q-1}+\epsilon_{i,q}\ ,$$ with the most recent eight quarters (quarter $q-7$ to q) of earnings (IBQ).
VOL	Earnings volatility. We use the standard deviation of earnings (IBQ) in the most recent eight quarters (quarter $q-7$ to q).
SIZE	Firm size for July of year t to June of year $t+1$ is defined as June market capitalization (from CRSP) of year t.

Variables	Descriptions
BM	Book-to-market ratio for July of year t to June of year $t+1$ is defined as book equity for the fiscal year ending in calendar year $t-1$ divided by the market capitalization at the end of December of $t-1$. Book equity is computed as stockholders' book equity (SEQ), plus deferred taxes (TXDB, zero if missing) and investment tax credit (ITCB, zero if missing), minus the book value of preferred stock (depending on availability, we use redemption (PSTKRF), carrying (PSTKL), or par value (PSTK)).
GP	Gross profitability for July of year t to June of year $t+1$ is defined as sales revenue (SALE) minus cost of goods sold (COGS), divided by total assets (AT) for the fiscal year ending in calendar year $t-1$. If SALE is unavailable, we use revenue (REVT). If COGS is unavailable, we use total operating expenses (XOPR) minus selling, general and administrative expenses (XSGA, zero if missing).
OP	Operating profitability for July of year t to June of year $t+1$ is defined as sales revenue (SALE) minus cost of goods sold (COGS), minus selling, general, and administrative expenses (XSGA), and plus research and development expenditures (XRD, zero if missing), scaled by total assets (AT) for the fiscal year ending in calendar year $t-2$. If SALE is unavailable, we use revenue (REVT). If COGS is unavailable, we use total operating expenses (XOPR) minus selling, general and administrative expenses (XSGA, zero if missing).
TA	Total accruals for July of year t to June of year $t+1$ is defined as net income (NI) minus operating, investing, and financing net cash flows (OANCF, IVNCF, and FINCF) plus sales of stocks (SSTK, zero if missing) minus stock repurchases and dividends (items PRSTKC and DV, zero if missing) for the fiscal year ending in calendar year $t-1$, scaled by total assets (AT) for the fiscal year ending in $t-2$.
OA	Operating accruals for July of year t to June of year $t+1$ is defined as net income (NI) minus net cash flow from operations (OANCF) for the fiscal year ending in calendar year $t-1$, scaled by total assets (AT) for the fiscal year ending in $t-2$.
AG	Asset growth for July of year t to June of year $t+1$ is defined as total assets (AT) for the fiscal year ending in calendar year $t-1$ minus total assets for the fiscal year ending in $t-2$, scaled by total assets for the fiscal year ending in $t-2$.
ROA	Return on assets is defined as quarterly earnings (IBQ) divided by total assets (ATQ) in the previous quarter.

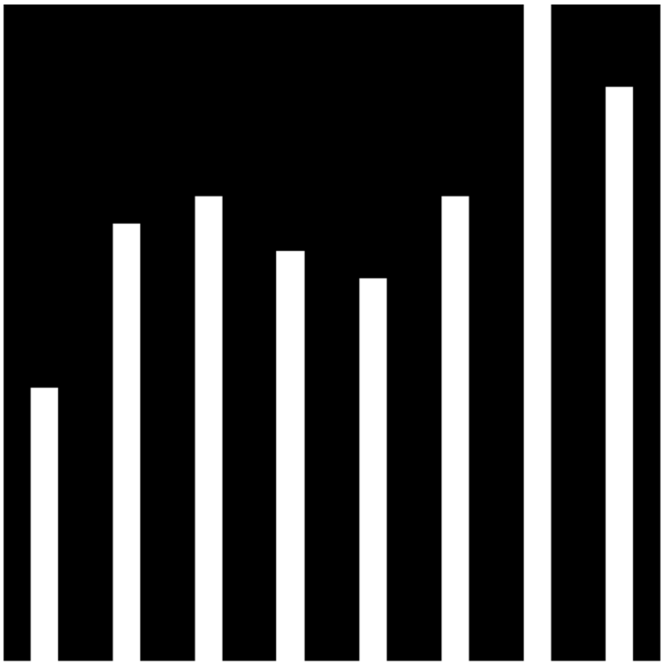


Panel A. Actual earnings chart from Meta's Q3 2024 Earnings Presentation



Panel B. Transformed 24 x 24 pixel image of Meta's earnings.

Figure 1. Examples of visualized earnings information. This figure displays the actual visualized earnings provided in Meta's Q3 2024 Earnings presentation and the transformed image that is given to the CNN to predict the buy probability.

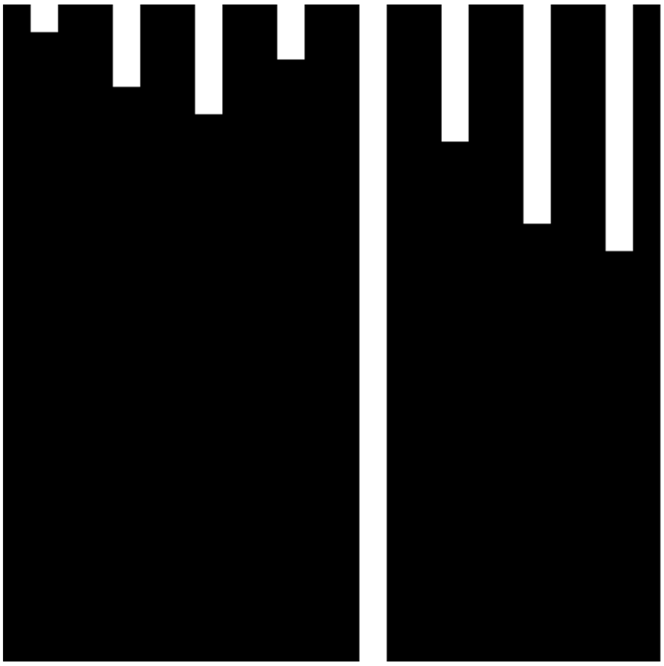


	E_1	E_2	E_3	E_4	E_5	E_6	E_7	E_8
Earnings (in millions)	3.163	4.882	5.068	4.472	4.243	5.263	7.348	6.542
$[x_1, x_2]$	[1, 2]	[4, 5]	[7, 8]	[10, 11]	[13, 14]	[16, 17]	[19, 20]	[22, 23]
$[y_1, y_2]$	[0, 10]	[0, 16]	[0, 17]	[0, 15]	[0, 14]	[0, 17]	[0, 24]	[0, 21]

Figure 2. Type I Earnings Image. This figure displays a black-and-white 24 × 24 pixels earnings image for a firm whose quarterly earnings in the most recent eight quarters (quarters q−7 to q) are all non-negative. E_1, E_2, ..., and E_8 represent the quarterly earnings in quarter q−7, q−6, ..., and q, respectively. The bottom-left vertex of an image is set as the origin of a two-dimensional coordinate system, and a rectangular area in an image is represented as $([x_1, x_2], [y_1, y_2])$.

	E_1	E_2	E_3	E_4	E_5	E_6	E_7	E_8
Earnings (in millions)	-0.263	-0.609	-0.110	0.114	0.322	1.122	0.989	0.945
$[x_1, x_2]$	[1, 2]	[4, 5]	[7, 8]	[10, 11]	[13, 14]	[16, 17]	[19, 20]	[22, 23]
$[y_1, y_2]$	[4, 8]	[0, 8]	[6, 8]	[8, 10]	[8, 12]	[8, 24]	[8, 22]	[8, 21]

Figure 3. Type II Earnings Image. This figure displays a black-and-white 24 × 24 pixels earnings image for a firm whose maximum quarterly earnings in the most recent eight quarters (quarters $q-7$ to q) is positive, and the minimum quarterly earnings in the most recent eight quarters is negative. E_1 , E_2, ..., and E_8 represent the quarterly earnings in quarter $q-7$, $q-6$, ..., and q, respectively. The bottom-left vertex of an image is set as the origin of a two-dimensional coordinate system, and a rectangular area in an image is represented as $([x_1, x_2], [y_1, y_2])$.



	E_1	E_2	E_3	E_4	E_5	E_6	E_7	E_8
Earnings (in millions)	-0.364	-1.214	-1.763	-0.920	-10.361	-1.985	-3.551	-4.016
$[x_1, x_2]$	[1, 2]	[4, 5]	[7, 8]	[10, 11]	[13, 14]	[16, 17]	[19, 20]	[22, 23]
$[y_1, y_2]$	[23, 24]	[21, 24]	[20, 24]	[22, 24]	[0, 24]	[19, 24]	[16, 24]	[15, 24]

Figure 4. Type III Earnings Image. This figure displays a black-and-white 24 × 24 pixels earnings image for a firm whose quarterly earnings in the most recent eight quarters (quarters $q-7$ to q) are all non-positive. E_1, E_2, ..., and E_8 represent the quarterly earnings in quarter $q-7$, $q-6$, ..., and q, respectively. The bottom-left vertex of an image is set as the origin of a two-dimensional coordinate system, and a rectangular area in an image is represented as $([x_1, x_2], [y_1, y_2])$.

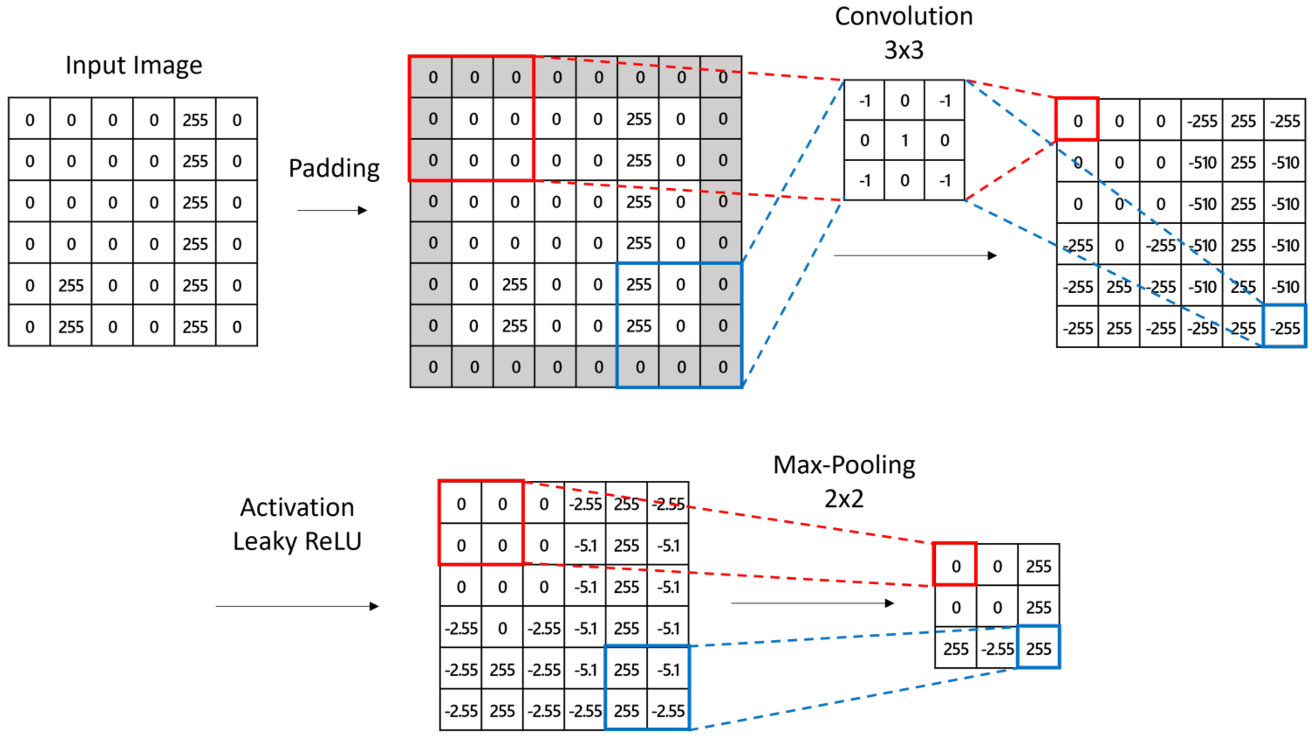


Figure 5. CNN Building Block: Padding, Convolution, Activation, and Max-Pooling. This figure displays how padding, convolution, activation, and max-pooling work in a CNN building block. The example input image is black-and-white and of size 6 × 6 pixels. The convolutional filter is of size 3 × 3 pixels. By using padding, the output after convolution has the same size of 6 × 6 pixels. The activation function is Leaky ReLU, which transforms an input value x to itself if $x > 0$ and $0.01x$ otherwise. The max-pooling (2 × 2 pixels) operation shrinks the input width and height to half by extracting the maximum element within a 2 × 2 pixels area and sliding through the image with a stride of 2.

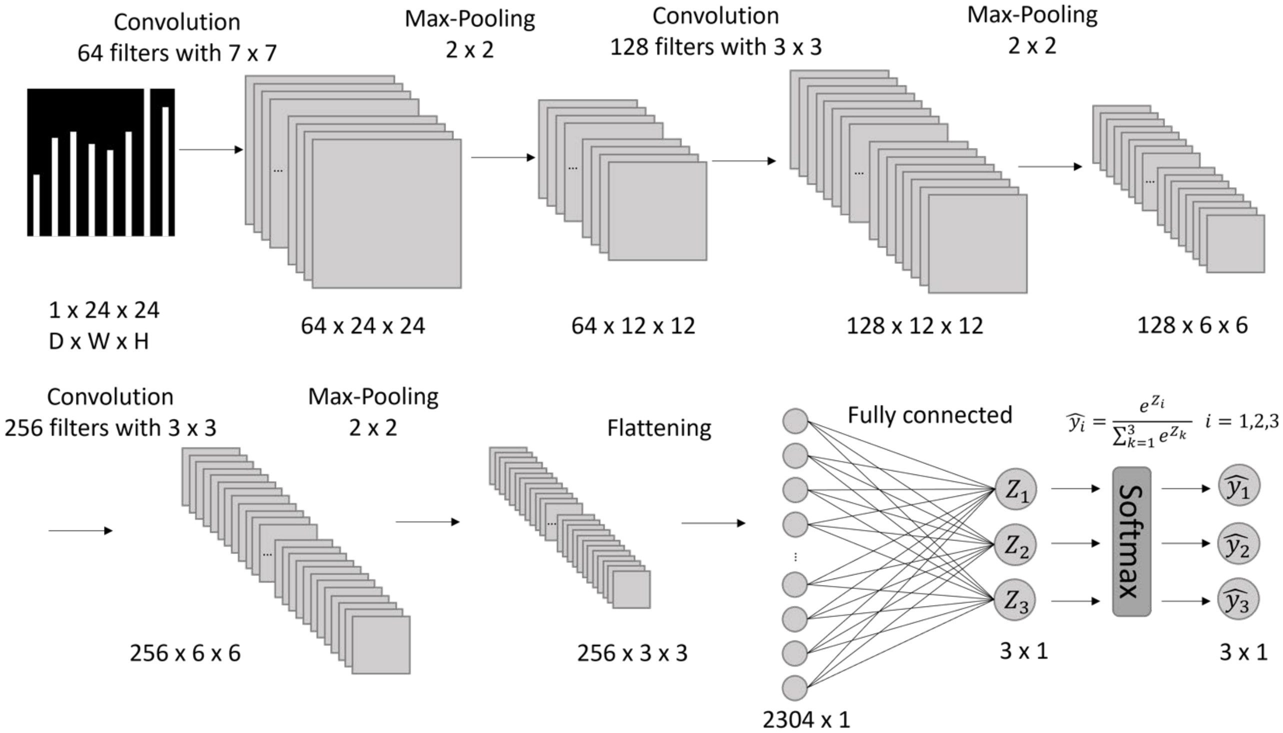


Figure 6. CNN Architecture Diagram. This figure displays the architecture of the CNN model. The notation $D \times W \times H$ represents the size of an image/feature map, where D is the depth, W is the width, and H is the height. The input black-and-white image is of size 24 × 24 pixels. There are three "blocks" in the model, with each block consisting of a convolutional layer and a max-pooling layer. The first convolutional layer has 64 filters of size 7 × 7 pixels, the second convolutional layer has 128 filters of size 3 × 3 pixels, and the third convolutional layer has 256 filters of size 3 × 3 pixels. After convolution, the output has the same width and height as those of the input due to padding, while its depth increases to the number of filters in the convolutional layer. After max-pooling, the output has half the width and height of the input, while its depth is the same as that of the input. Flattening refers to the process of converting the elements in a series of matrices into a vector. The fully connected layer linearly transforms the values in the vector to produce three "scores" of the three labels (label 1 = "sell", label 2 = "hold", label 3 = "buy"). Finally, the Softmax function transforms the three scores to three probabilities ($\widehat{y_1}, \widehat{y_2}, \widehat{y_3}$) that sum to one, and $\widehat{y_3}$ is the *CNN buy* probability.

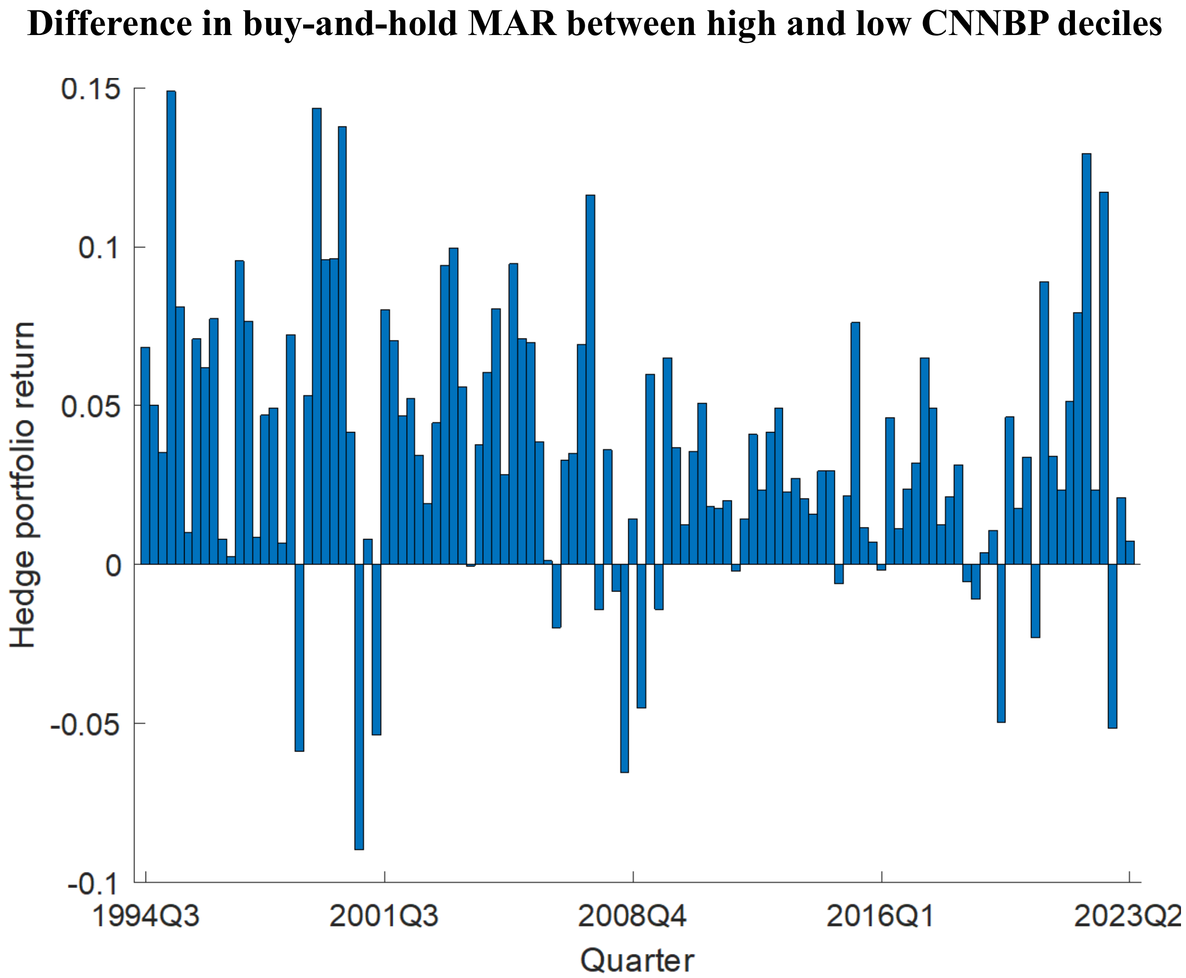


Figure 7. Time Stability of CNN Out-Of-Sample Drift-Predicting Performance. This figure depicts the difference in the average 63-day buy-and-hold market-adjusted returns (MAR) between high and low *CNN buy* probability (CNNBP) deciles in each quarter during the out-of-sample period (1994Q3 to 2023Q2, 116 quarters). The decile cutoffs are based on the distribution of the previous quarter's *CNN buy* probability.

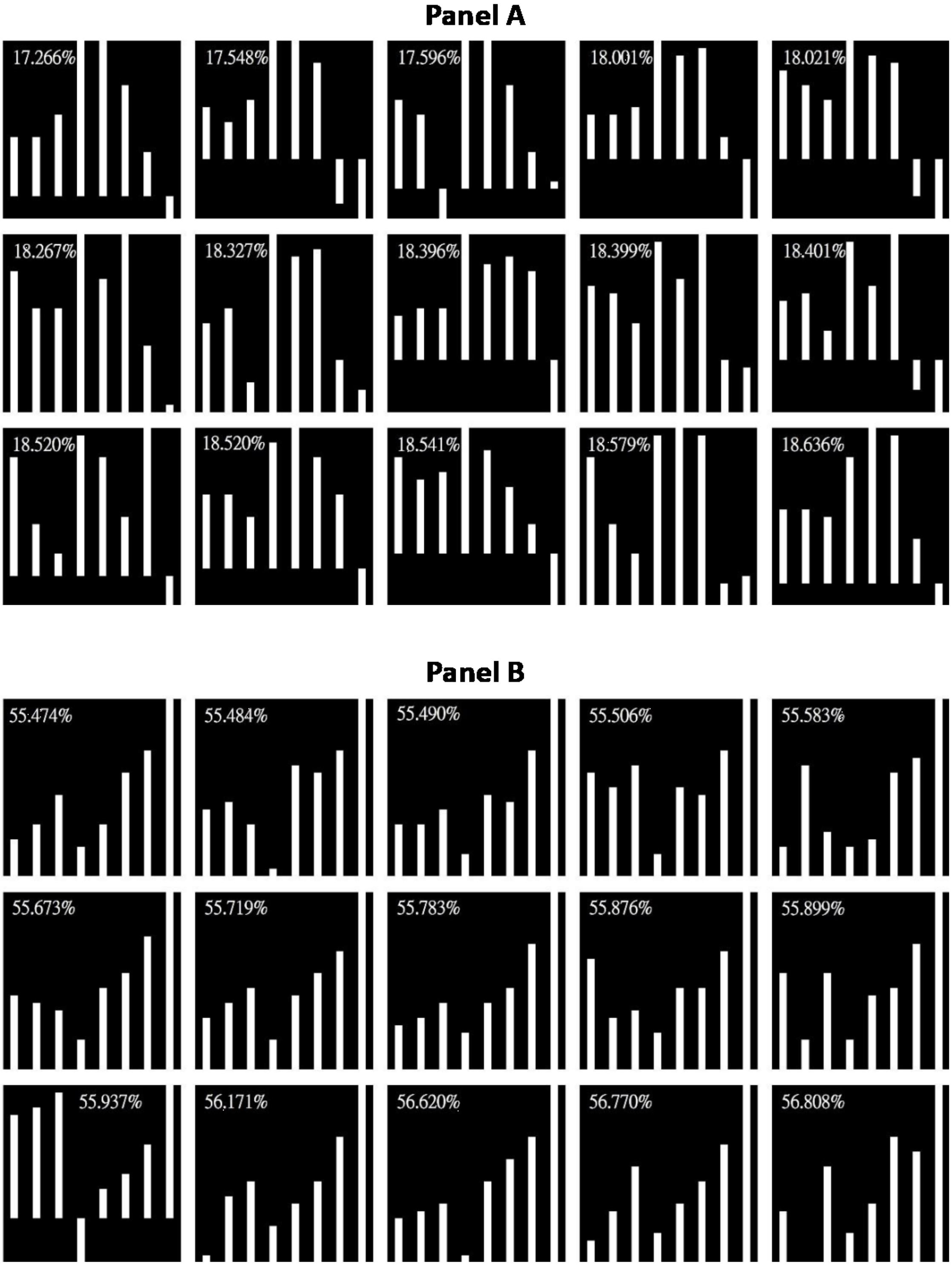


Figure 8. Earnings Images of the Lowest and Highest *CNN Buy* Probabilities. Panels A and B present earnings images whose *CNN buy* probabilities rank in the top 15 and bottom 15 among those of all earnings images in the out-of-sample period (1994Q3 to 2023Q2, 116 quarters), respectively. The corresponding *CNN buy* probabilities are also reported in each earnings image.

Table 1.

Sample Selection

All Compustat firm-quarters with matched CRSP Permno (SHRCD = 10 or 11; EXCHCD = 1, 2 or 3) whose earnings announcement date (Compustat item RDQ) is between 1974/01/01 and 2023/06/30	815,907
Drop observations with missing RDQ in the most recent eight quarters	(86,724)
Drop observations with earnings announcements on the same date for the same firm in the most recent eight quarters	(3,349)
Drop observations with RDQ less than 30 days away from the previous quarter RDQ in the most recent eight quarters	(17,028)
Drop observations with RDQ before or more than 180 days after the quarter fiscal period end date in the most recent eight quarters	(2,971)
Drop observations with missing earnings (Compustat item IBQ) in the most recent eight quarters	(74,655)
Drop observations whose CRSP daily price at the current quarter RDQ is missing or ≤$1	(102,680)
Drop financial firms (SIC codes between 6000 and 6999) and utility firms (SIC codes between 4900 and 4949)	(109,418)
Drop observations with non-positive BM or missing SIZE	(14,423)
Drop observations with more than 30 missing CRSP daily returns in the 120-day window ([-150, -31]) prior to the current quarter RDQ	(24)
Total observations	404,635
In-sample dataset: observations whose current quarter RDQ is between 1974/01/01 and 1993/12/31	124,413
Out-of-sample dataset: observations whose current quarter RDQ is between 1994/07/01 and 2023/06/30 and have non-missing SUE, EA, TREND, RET[−1, 1], RET[−30, −2], PERSIST, VOL, GP, OP, OA, TA, and AG	240,844

This table reports the sample selection procedures. The in-sample dataset is for CNN model training and validation. The out-of-sample dataset is for testing the out-of-sample CNN model performance, and thus serves as the dataset for all empirical analyses throughout the paper. See Appendix for variable definitions.

Table 2.

***CNN Buy* Probability and Post-Earnings Announcement Drift: Univariate Portfolio Analysis.**

	The 63-day post-announcement buy-and-hold abnormal return					
CNNBP deciles	MAR	SAR	FF4	FF6	HMXZ5	DHS3
Low [25.5%]	−0.006	−0.015***	−0.007***	−0.008***	−0.007**	0.001
	(−1.137)	(−5.210)	(−3.364)	(−3.622)	(−2.458)	(0.293)
2 [28.8%]	−0.003	−0.012***	−0.003	−0.002	−0.001	0.004
	(−0.624)	(−4.714)	(−1.385)	(−1.201)	(−0.584)	(0.919)
3 [30.6%]	0.004	−0.006**	0.001	0.002	0.006*	0.012**
	(0.651)	(−2.610)	(0.270)	(0.512)	(1.979)	(2.150)
4 [31.9%]	0.003	−0.006*	−0.001	0.001	0.004	0.013*
	(0.449)	(−1.689)	(−0.215)	(0.116)	(1.104)	(1.833)
5 [33.1%]	0.006	−0.004	0.002	0.003	0.007*	0.015**
	(0.820)	(−1.359)	(0.518)	(0.680)	(1.929)	(2.123)
6 [34.3%]	0.011	0.001	0.007	0.007*	0.010***	0.019***
	(1.408)	(0.232)	(1.560)	(1.714)	(2.823)	(2.667)
7 [35.5%]	0.014*	0.004	0.009*	0.009**	0.013***	0.021***
	(1.729)	(1.204)	(1.946)	(2.085)	(3.407)	(2.792)
8 [37.1%]	0.015**	0.005	0.011***	0.011***	0.015***	0.022***
	(2.117)	(1.624)	(2.901)	(2.813)	(3.754)	(3.238)
9 [39.4%]	0.024***	0.014***	0.017***	0.016***	0.021***	0.029***
	(3.850)	(5.323)	(5.371)	(5.226)	(6.388)	(4.909)
High [44.4%]	0.030***	0.020***	0.024***	0.024***	0.027***	0.036***
	(4.994)	(5.093)	(6.555)	(6.830)	(7.375)	(6.039)
High-Low [18.9%]	0.036***	0.035***	0.031***	0.032***	0.034***	0.034***
	(7.213)	(7.111)	(8.092)	(9.035)	(8.378)	(8.192)

This table reports the average 63-day buy-and-hold abnormal return (BHAR) after earnings announcements, including market-adjusted return (MAR), size-adjusted return (SAR), and factor-adjusted returns (FF4, FF6, HMXZ5, andDHS3) for portfolios formed based on *CNN buy* probability (CNNBP) deciles in the out-of-sample period (1994Q3-2023Q2, 116 quarters). The CNNBP decile cutoffs are based on the distribution of the previous quarter's CNNBP. The average CNNBP for each CNNBP decile is reported in brackets. See Appendix for variable definitions. Newey and West [1987] *t*-statistics with three lags are reported in parentheses, and ***, **, and * indicate significance at the 1%, 5%, and 10% level, respectively.

Table 3.

Firm Characteristics and Post-Earnings Announcement Drift: Univariate Portfolio Analysis

	The 63-day post-announcement buy-and-hold abnormal return difference between the highest and lowest variable deciles					
Variable	MAR	SAR	FF4	FF6	HMXZ5	DHS3
CNNBP	0.036***	0.035**	0.031***	0.032***	0.034***	0.034***
	(7.213)	(7.111)	(8.092)	(9.035)	(8.378)	(8.192)
SUE	0.026***	0.026***	0.021***	0.021***	0.021***	0.020***
	(5.025)	(5.142)	(5.558)	(6.161)	(5.559)	(4.691)
EA	0.024***	0.024***	0.022***	0.023***	0.021***	0.024***
	(6.969)	(6.839)	(6.877)	(6.827)	(6.044)	(7.489)
TREND	0.020***	0.018***	0.014***	0.014***	0.016***	0.015***
	(3.353)	(3.154)	(2.974)	(3.073)	(3.188)	(2.789)
RET[-1, 1]	0.035***	0.035***	0.038***	0.037***	0.035***	0.033***
	(5.551)	(5.967)	(7.711)	(7.813)	(6.499)	(5.660)
RET[-30, -2]	−0.003	0.002	0.005	0.006	0.002	−0.004
	(−0.291)	(0.268)	(0.812)	(1.056)	(0.217)	(−0.572)
PERSIST	−0.003	−0.003	−0.004	−0.002	−0.007*	−0.002
	(−0.983)	(−0.876)	(−1.127)	(−0.536)	(−1.792)	(−0.459)
VOL	−0.006	−0.003	−0.018**	−0.021***	−0.022***	−0.021***
	(−0.769)	(−0.413)	(−2.406)	(−2.898)	(−2.770)	(−2.740)
SIZE	−0.017	−0.012*	−0.026**	−0.027**	−0.032***	−0.034**
	(−1.376)	(−1.956)	(−2.041)	(−2.293)	(−2.828)	(−2.517)
BM	0.019*	0.016*	0.022***	0.017**	0.025***	0.020**
	(1.692)	(1.685)	(2.625)	(2.177)	(2.899)	(2.073)
GP	0.020***	0.020***	0.024***	0.019***	0.010*	0.016***
	(2.910)	(3.222)	(4.011)	(3.091)	(1.677)	(2.842)
OP	0.017	0.018**	0.017	0.015	0.001	0.010
	(1.518)	(2.352)	(1.615)	(1.528)	(0.100)	(0.895)
OA	−0.005	−0.005	0.000	−0.002	−0.003	−0.004
	(−0.770)	(−0.763)	(−0.064)	(−0.441)	(−0.762)	(−0.743)
TA	−0.012	−0.010	−0.006	−0.007	−0.014**	−0.011
	(−1.406)	(−1.502)	(−0.734)	(−1.013)	(−2.140)	(−1.307)
AG	−0.019**	−0.017**	−0.018**	−0.016**	−0.018**	−0.017*
	(−2.108)	(−2.236)	(−2.166)	(−2.047)	(−2.532)	(−1.936)

This table reports the average 63-day post-announcement buy-and-hold abnormal return (BHAR) difference between the highest and lowest variable deciles in the out-of-sample period (1994Q3-2023Q2, 116 quarters). We use market-adjusted return (MAR), size-adjusted return (SAR), and factor-adjusted returns (FF4, FF6, HMXZ5, and DHS3) as BHAR measures. The variables include the *CNN buy* probability (CNNBP), standardized unexpected earnings (SUE), earnings acceleration (EA), trend in gross profitability (TREND), market capitalization (SIZE), book-to-market ratio (BM), earnings announcement return (RET[−1, 1]), pre-announcement return (RET[−30, −2]), earnings persistence (PERSIST), earnings volatility (VOL), gross profitability (GP), operating profitability (OP), total accruals (TA), operating accruals (OA), and asset growth (AG). The quintile cutoffs for quarterly variables are based on the distribution of these variables in the previous quarter, and the quintile cutoffs for annual variables from July in year t to June in year $t+1$ are based on the distribution of these variables at the end of June in year t. Newey and West [1987] t-statistics with three lags are reported in parentheses, and ***, **, and * indicate significance at the 1%, 5%, and 10% level, respectively.

Table 4.

***CNN Buy* Probability and Post-Earnings Announcement Drift: Double Sorts**

Panel A: Two-way sorting, controlling for standardized unexpected earnings (SUE)

	CNN buy probability quintiles					
SUE quintiles	Low	2	3	4	High	High-Low
Low	−0.007	0.000	−0.008	−0.004	0.007	0.013
	(−1.295)	(0.025)	(−0.920)	(−0.385)	(0.541)	(1.208)
2	−0.005	−0.003	0.006	0.001	0.007	0.012
	(−0.967)	(−0.565)	(0.612)	(0.088)	(0.587)	(1.265)
3	−0.005	0.005	0.010	0.016**	0.021***	0.025***
	(−0.836)	(0.567)	(1.325)	(2.147)	(3.040)	(4.088)
4	0.002	0.011	0.015*	0.024***	0.028***	0.027***
	(0.291)	(1.336)	(1.936)	(3.137)	(4.368)	(4.044)
High	0.009	0.009	0.017**	0.021***	0.029***	0.019***
	(1.425)	(1.217)	(2.377)	(3.087)	(5.399)	(3.311)
High-Low	0.016***	0.009	0.025***	0.024***	0.021*	
	(3.972)	(1.280)	(4.269)	(3.478)	(1.748)	

Panel B: Two-way sorting, controlling for earnings acceleration (EA)

	CNN buy probability quintiles					
EA quintiles	Low	2	3	4	High	High-Low
Low	−0.006	0.002	0.008	0.007	0.029***	0.035***
	(−0.634)	(0.176)	(0.726)	(0.594)	(2.921)	(5.031)
2	−0.006	0.000	0.001	0.009	0.021***	0.027***
	(−1.121)	(0.044)	(0.220)	(1.406)	(3.465)	(5.954)
3	0.000	0.000	0.007	0.009**	0.016***	0.016***
	(0.028)	(0.014)	(1.574)	(1.992)	(3.158)	(2.711)
4	0.001	0.000	0.008	0.014**	0.020***	0.019***
	(0.175)	(−0.019)	(1.440)	(2.397)	(3.998)	(3.746)
High	−0.013	0.017	0.016	0.027***	0.041***	0.054***
	(−1.072)	(1.179)	(1.450)	(2.684)	(5.286)	(5.505)
High-Low	−0.007	0.015**	0.008*	0.020***	0.012**	
	(−0.744)	(2.025)	(1.709)	(4.476)	(2.021)	

Panel C: Two-way sorting, controlling for trend in gross profitability (TREND)

	CNN buy probability quintiles					
TREND quintiles	Low	2	3	4	High	High-Low
Low	−0.009	−0.002	0.004	0.012	0.023***	0.032***
	(−1.470)	(−0.197)	(0.480)	(1.300)	(3.193)	(6.792)
2	−0.002	0.001	0.001	0.008	0.024***	0.025***
	(−0.332)	(0.136)	(0.089)	(1.157)	(4.092)	(5.332)
3	−0.005	−0.001	0.005	0.016**	0.021***	0.026***
	(−0.924)	(−0.262)	(0.963)	(2.017)	(3.389)	(5.801)
4	−0.001	0.007	0.011	0.016**	0.025***	0.027***
	(−0.272)	(1.125)	(1.534)	(2.392)	(3.910)	(5.750)
High	−0.002	0.014	0.018*	0.018*	0.037***	0.039***
	(−0.352)	(1.157)	(1.684)	(1.940)	(4.740)	(6.632)
High-Low	0.007	0.016**	0.014**	0.005	0.013**	
	(1.424)	(2.206)	(2.357)	(0.963)	(2.037)	

This table reports the average 63-day buy-and-hold market-adjusted returns (MAR) after earnings announcements for portfolios formed based on the *CNN buy* probability quintiles and one of the seven firm characteristics including standardized unexpected earnings (SUE), earnings acceleration (EA), trend in gross profitability (TREND), earnings announcement return (RET[-1, 1]), book-to-market ratio (BM), gross profitability (GP), and asset growth (AG) using independent two-way sorting in the out-of-sample period (1994Q3-2023Q2, 116 quarters). The quintile cutoffs for quarterly variables are based on the distribution of these variables in the previous quarter, and the quintile cutoffs for annual variables from July in year t to June in year $t+1$ are based on the distribution of these variables at the end of June in year t. See Appendix for variable definitions. Newey and West [1987] t-statistics with three lags are reported in parentheses, and ***, **, and * indicate significance at the 1%, 5%, and 10% level, respectively.

Table 4.

***CNN Buy* Probability and Post-Earnings Announcement Drift: Double Sorts (continued)**

Panel D: Two-way sorting, controlling for earnings announcement return (RET[−1, 1])

	CNN buy probability quintiles					
RET[−1, 1] quintiles	Low	2	3	4	High	High-Low
Low	−0.011	−0.003	−0.005	0.003	0.017**	0.028***
	(−1.555)	(−0.292)	(−0.493)	(0.275)	(2.008)	(5.094)
2	−0.004	0.001	0.005	0.005	0.019***	0.023***
	(−0.852)	(0.090)	(0.677)	(0.814)	(3.083)	(4.832)
3	−0.002	0.003	0.003	0.014**	0.019***	0.021***
	(−0.487)	(0.511)	(0.542)	(2.156)	(3.681)	(5.625)
4	−0.003	0.011	0.012*	0.016**	0.025***	0.028***
	(−0.574)	(1.510)	(1.757)	(2.374)	(4.220)	(6.207)
High	0.001	0.010	0.029***	0.030***	0.044***	0.044***
	(0.088)	(1.123)	(2.966)	(3.038)	(5.706)	(5.913)
High-Low	0.011**	0.012*	0.035***	0.027***	0.027***	
	(2.033)	(1.861)	(6.177)	(3.847)	(4.030)	

Panel E: Two-way sorting, controlling for book-to-market ratio (BM)

	CNN buy probability quintiles					
BM quintiles	Low	2	3	4	High	High-Low
Low	−0.003	−0.004	0.000	0.007	0.017***	0.020***
	(−0.660)	(−0.575)	(0.045)	(1.146)	(2.876)	(3.111)
2	0.000	0.002	0.005	0.012**	0.019***	0.019***
	(0.036)	(0.373)	(0.687)	(2.083)	(3.578)	(4.478)
3	−0.003	0.007	0.008	0.013*	0.025***	0.027***
	(−0.477)	(0.869)	(1.333)	(1.841)	(3.884)	(6.189)
4	−0.004	0.006	0.011	0.014	0.029***	0.033***
	(−0.639)	(0.679)	(1.323)	(1.548)	(4.276)	(7.875)
High	−0.013*	0.010	0.017	0.020*	0.044***	0.057***
	(−1.787)	(0.863)	(1.468)	(1.796)	(4.274)	(7.456)
High-Low	−0.010	0.014	0.017	0.013	0.027**	
	(−1.407)	(1.326)	(1.632)	(1.322)	(2.532)	

Panel F: Two-way sorting, controlling for gross profitability (GP)

	CNN buy probability quintiles					
GP quintiles	Low	2	3	4	High	High-Low
Low	−0.016**	−0.006	−0.005	0.002	0.018***	0.034***
	(−2.215)	(−0.504)	(−0.555)	(0.185)	(2.184)	(4.747)
2	−0.007	0.007	0.012*	0.010	0.025***	0.032***
	(−1.238)	(0.944)	(1.722)	(1.476)	(3.790)	(7.520)
3	−0.004	0.003	0.015*	0.014**	0.030***	0.035***
	(−0.853)	(0.462)	(1.931)	(2.007)	(4.698)	(6.854)
4	0.005	0.006	0.014*	0.019***	0.028***	0.024***
	(0.921)	(0.985)	(1.947)	(2.699)	(4.563)	(4.837)
High	−0.002	0.009	0.010	0.026***	0.036***	0.038***
	(−0.299)	(1.219)	(1.387)	(2.765)	(5.585)	(8.569)
High-Low	0.015**	0.014*	0.016***	0.024***	0.019**	
	(2.490)	(1.751)	(2.725)	(3.718)	(2.340)	

Table 4.

***CNN Buy* Probability and Post-Earnings Announcement Drift: Double Sorts (continued)**

Panel G: Two-way sorting, controlling for asset growth (AG)

	CNN buy probability quintiles					
AG quintiles	Low	2	3	4	High	High-Low
Low	−0.006	0.009	0.014	0.018	0.037***	0.042***
	(−0.689)	(0.830)	(1.316)	(1.633)	(4.198)	(8.064)
2	−0.003	0.003	0.013*	0.017**	0.033***	0.036***
	(−0.599)	(0.429)	(1.696)	(2.204)	(4.643)	(7.288)
3	−0.002	0.006	0.008	0.016**	0.026***	0.028***
	(−0.491)	(1.089)	(1.305)	(2.566)	(3.877)	(6.338)
4	0.000	0.002	0.011	0.013*	0.023***	0.024***
	(−0.092)	(0.324)	(1.588)	(1.855)	(4.508)	(5.149)
High	−0.010	−0.003	−0.005	0.005	0.019***	0.029***
	(−1.478)	(−0.323)	(−0.644)	(0.670)	(3.295)	(5.409)
High-Low	−0.004	−0.012	−0.019***	−0.013*	−0.018***	
	(−0.690)	(−1.600)	(−2.878)	(−1.796)	(−2.773)	

Table 5.

***CNN Buy* Probability and Post-Earnings Announcement Drift: Regression Analysis**

	(1) MAR	(2) SAR	(3) FF4	(4) FF6	(5) HMXZ5	(6) DHS3
Intercept	0.011*	0.000	0.007**	0.007**	0.010***	0.019***
	(1.707)	(0.216)	(2.098)	(2.308)	(3.677)	(3.068)
CNNBP	0.014***	0.014***	0.010***	0.011***	0.012***	0.014***
	(3.529)	(3.535)	(3.507)	(3.652)	(3.971)	(4.344)
SUE	0.011***	0.012***	0.011***	0.010***	0.010***	0.009**
	(2.951)	(3.009)	(3.406)	(3.146)	(2.840)	(2.389)
EA	0.013***	0.013***	0.012***	0.013***	0.013***	0.014***
	(4.892)	(4.713)	(4.681)	(5.133)	(4.833)	(5.417)
TREND	0.016***	0.016***	0.012***	0.013***	0.012***	0.013***
	(4.769)	(4.717)	(3.797)	(3.679)	(3.961)	(4.291)
RET[-1, 1]	0.021***	0.023***	0.024***	0.024***	0.023***	0.021***
	(5.734)	(6.520)	(7.126)	(6.861)	(7.008)	(5.788)
RET[-30, -2]	−0.011**	−0.006	−0.003	−0.003	−0.006	−0.010**
	(−2.113)	(−1.207)	(−0.745)	(−0.685)	(−1.106)	(−2.094)
PERSIST	0.000	0.000	−0.001	−0.001	−0.003	0.001
	(−0.155)	(−0.117)	(−0.594)	(−0.281)	(−1.409)	(0.385)
VOL	0.014	0.014	0.009	0.005	0.009	0.010
	(1.281)	(1.356)	(1.217)	(0.824)	(1.207)	(1.212)
SIZE	−0.028*	−0.023**	−0.035**	−0.035**	−0.040***	−0.042**
	(−1.697)	(−2.298)	(−2.216)	(−2.283)	(−2.789)	(−2.562)
BM	0.014	0.014	0.017***	0.011**	0.011***	0.008
	(1.359)	(1.356)	(2.755)	(2.089)	(1.267)	(1.388)
GP	0.009	0.010	0.012*	0.008	0.006	0.008
	(1.552)	(1.612)	(1.838)	(1.237)	(1.086)	(1.319)
OP	0.030***	0.030***	0.033***	0.031***	0.022***	0.027***
	(3.925)	(3.975)	(3.896)	(3.452)	(3.522)	(3.700)
OA	0.001	0.001	0.001	0.001	0.000	0.000
	(0.322)	(0.431)	(0.388)	(0.224)	(0.002)	(−0.089)
TA	−0.008**	−0.008**	−0.004	−0.005	−0.005	−0.006
	(−2.218)	(−2.184)	(−0.906)	(−1.388)	(−1.407)	(−1.388)
AG	−0.011**	−0.011**	−0.012***	−0.010**	−0.008**	−0.007*
	(−2.373)	(−2.439)	(−2.633)	(−2.248)	(−2.026)	(−1.667)
Adj. R^2	0.045	0.038	0.032	0.028	0.030	0.036
obs.	240,844	240,844	240,844	240,844	240,844	240,844

The table presents results of quarterly weighted Fama and MacBeth [1973] regressions in the out-of-sample period (1994Q3-2023Q2, 116 quarters) using the 63-day buy-and-hold abnormal return (BHAR) after earnings announcements, including market-adjusted return (MAR), size-adjusted return (SAR), and factor-adjusted returns (FF4, FF6, HMXZ5, and DHS3), as the dependent variables. The weights correspond to the number of observations used in each quarterly cross-sectional regression. The independent variables include the *CNN buy* probability (CNNBP), standardized unexpected earnings (SUE), earnings acceleration (EA), trend in gross profitability (TREND), market capitalization (SIZE), book-to-market ratio (BM), earnings announcement return (RET[−1, 1]), pre-announcement return (RET[−30, −2]), earnings persistence (PERSIST), earnings volatility (VOL), gross profitability (GP), operating profitability (OP), total accruals (TA), operating accruals (OA), and asset growth (AG). All variables except for six measures of BHAR (MAR, SAR, FF4, FF6, HMXZ5, and DHS3) are converted into scaled ranks ranging from −0.5 to 0.5 with a mean of zero. See Appendix for variable definitions. Newey and West [1987] t-statistics with three lags are reported in parentheses, and ***, **, and * indicate significance at the 1%, 5%, and 10% level, respectively.

Table 6.

***CNN Buy* Probability, Firm Characteristics, and Historical Earnings**

	(1) CNNBP		(2) CNNBP
Intercept	−0.002	Intercept	−0.004***
	(−1.598)		(−3.040)
SUE	0.510***	ROA_q	0.644***
	(77.045)		(41.254)
EA	0.040***	ROA_{q-1}	0.038***
	(6.851)		(6.965)
TREND	0.004	ROA_{q-2}	−0.073***
	(0.792)		(−14.535)
RET[-1, 1]	0.050***	ROA_{q-3}	−0.333***
	(20.524)		(−62.953)
RET[-30, -2]	0.024**	ROA_{q-4}	−0.428
	(7.350)		(−48.276)
PERSIST	−0.044***	ROA_{q-5}	0.079***
	(−8.605)		(16.625)
VOL	0.246***	ROA_{q-6}	0.044***
	(21.668)		(12.322)
SIZE	−0.274***	ROA_{q-7}	−0.021***
	(−18.569)		(−4.567)
BM	0.037***		
	(5.876)		
GP	0.010**		
	(2.155)		
OP	−0.023***		
	(−3.938)		
OA	0.001		
	(0.186)		
TA	−0.019***		
	(−8.196)		
AG	0.047***		
	(9.380)		
Adj. R^2	0.328	Adj. R^2	0.300
obs.	240,844	obs.	240,043

The table presents results of quarterly weighted Fama and MacBeth [1973] regressions in the out-of-sample period (1994Q3-2023Q2, 116 quarters) using the *CNN buy* probability (CNNBP) as the dependent variable. The weights correspond to the number of observations used in each quarterly cross-sectional regression. The independent variables in specification 1 are standardized unexpected earnings (SUE), earnings acceleration (EA), trend in gross profitability (TREND), market capitalization (SIZE), book-to-market ratio (BM), earnings announcement return (RET[−1, 1]), pre-announcement return (RET[−30, −2]), earnings persistence (PERSIST), earnings volatility (VOL), gross profitability (GP), operating profitability (OP), total accruals (TA), operating accruals (OA), and asset growth (AG). The independent variables in specification 2 are the return on assets (ROA) in the most recent eight quarters (quarters $q-7$ to q). All variables are converted into scaled ranks ranging from −0.5 to 0.5 with a mean of zero. See Appendix for variable definitions. Newey and West [1987] t-statistics with three lags are reported in parentheses, and ***, **, and * indicate significance at the 1%, 5%, and 10% level, respectively.

Table 7.
***CNN Buy* Probability, Future Earnings Growth, and Future Three-Day Abnormal Returns around Earnings Announcements**

	(1)	(2)	(3)	(4)
	SUE_{q+1}	SUE_{q+1}	$\text{RET}[-1,1]_{q+1}$	$\text{RET}[-1,1]_{q+1}$
Intercept	0.002	0.002	0.002***	0.002***
	(0.779)	(0.802)	(3.564)	(3.586)
CNNBP	0.113***	0.133***	0.004***	0.004***
	(13.731)	(20.601)	(4.282)	(5.185)
SUE	0.377***	0.346***	−0.002	−0.002
	(30.493)	(31.877)	(−1.319)	(−1.531)
EA	−0.090***	−0.086***	0.001	0.001
	(−24.490)	(−24.216)	(0.866)	(1.088)
TREND	0.021***	0.024***	0.000	0.001
	(4.331)	(6.614)	(0.165)	(1.470)
RET[-1, 1]	0.078***	0.077***	0.004***	0.004***
	(37.487)	(37.871)	(4.348)	(3.899)
RET[-30, -2]	0.067***	0.066***	0.001	0.000
	(16.162)	(16.506)	(0.548)	(−0.110)
PERSIST	0.001	−0.002	−0.002***	−0.001**
	(0.138)	(−0.510)	(−3.011)	(−2.029)
VOL	0.009**	−0.076***	−0.001	−0.002
	(2.237)	(−8.943)	(−0.904)	(−1.283)
SIZE		0.119***		0.003
		(10.848)		(1.253)
BM		−0.018***		0.007***
		(−3.278)		(5.733)
GP		0.012**		0.006***
		(2.107)		(4.640)
OP		−0.022***		0.005***
		(−3.443)		(4.847)
OA		−0.016***		0.000
		(−5.093)		(0.521)
TA		−0.002		−0.001
		(−0.728)		(−1.326)
AG		−0.015***		−0.002**
		(−3.934)		(−2.187)
Adj. R^2	0.214	0.228	0.003	0.007
obs.	237,118	237,118	236,239	236,239

The table presents results of quarterly weighted Fama and MacBeth [1973] regressions in the out-of-sample period (1994Q3-2023Q2, 116 quarters) using one-quarter-ahead standardized unexpected earnings (SUE_{q+1}) or the three-day abnormal return around the next earnings announcement date ($\text{RET}[-1, 1]_{q+1}$) as the dependent variable. The weights correspond to the number of observations used in each quarterly cross-sectional regression. The independent variables include *CNN buy* probability (CNNBP), standardized unexpected earnings (SUE), earnings acceleration (EA), trend in gross profitability (TREND), market capitalization (SIZE), book-to-market ratio (BM), earnings announcement return (RET[−1, 1]), pre-announcement return (RET[−30, −2]), earnings persistence (PERSIST), earnings volatility (VOL), gross profitability (GP), operating profitability (OP), total accruals (TA), operating accruals (OA), and asset growth (AG). All variables are converted into scaled ranks ranging from −0.5 to 0.5 with a mean of zero. See Appendix for variable definitions. Newey and West [1987] *t*-statistics with three lags are reported in parentheses, and ***, **, and * indicate significance at the 1%, 5%, and 10% level, respectively

Table 8.
Test of Market Efficiency for the *CNN Buy* Features Effect

Parameter	3-day	Quarter-Long
γ_1	0.136***	0.136***
	(19.783)	(19.816)
γ_1^*	0.062***	0.010
	(3.814)	(0.233)
β	0.062***	0.010***
	(38.571)	(18.547)

Market efficiency test $(\gamma_1 = \gamma_1^*)$	3-day	Quarter-long
Likelihood ratio statistic	8.218***	21.444***

This table reports the regression results from nonlinear generalized least squares estimation of the following two equations in the out-of-sample period (1994Q3-2023Q2, 116 quarters)

$$\text{Forecasting equation: } \text{SUE}_{q+1} = \alpha + \gamma_1 \text{CNNBP}_q + \sum \gamma_c \text{Controls}_q + \delta_{q+1}$$
$$\text{Pricing equation: } \text{AR}_{q+1} = \beta(\text{SUE}_{q+1} - \alpha^* - \gamma_1^* \text{CNNBP}_q - \sum \gamma_c^* \text{Controls}_q) + \varepsilon_{q+1}.$$

The control variables include standardized unexpected earnings (SUE), earnings acceleration (EA), trend in gross profitability (TREND), market capitalization (SIZE), book-to-market ratio (BM), earnings announcement return (RET[−1, 1]), pre-announcement return (RET[−30, −2]), earnings persistence (PERSIST), earnings volatility (VOL), gross profitability (GP), operating profitability (OP), total accruals (TA), operating accruals (OA), and asset growth (AG). AR is the abnormal return from a 3-day window around quarter $q + 1$'s earnings or the quarter-long window starting two days after the quarter q earnings and ending on the next announcement date. All variables except for the abnormal return AR are converted into scaled ranks ranging from −0.5 to 0.5 with a mean of zero. *t*-statistics based on firm and quarter double-clustered standard errors are reported in parentheses. The likelihood ratio statistic for testing $\gamma_1 = \gamma_1^*$ is distributed asymptotically as $\mathcal{X}^2(1)$. ***, **, and * indicate significance at the 1%, 5%, and 10% level, respectively.

Table 9.

Month-Based Rebalancing Strategy

	Equal-weighted returns					Value-weighted returns				
CNNBP deciles	Excess return	FF4 alpha	FF6 alpha	HMXZ5 alpha	DHS3 alpha	Excess return	FF4 alpha	FF6 alpha	HMXZ5 alpha	DHS3 alpha
Low	0.006*	−0.001	−0.002***	−0.002	0.000	0.006**	0.000	−0.001	−0.001	−0.001
	(1.955)	(−0.868)	(−2.681)	(−1.289)	(0 041)	(2.412)	(−0.189)	(−1.445)	(−0.765)	(−0.725)
2	0.007**	0.000	−0.001	0.000	0 001	0.007***	0.001	0.000	0.000	0.000
	(2.289)	(0.045)	(−0.827)	(0.166)	(0 914)	(3.003)	(1.018)	(−0.268)	(−0.025)	(0.253)
3	0.009**	0.002*	0.002*	0.003**	0 004**	0.007***	0.001	0.000	0.001	0.000
	(2.546)	(1.881)	(1.667)	(2.056)	(2 121)	(3.093)	(0.591)	(−0.285)	(0.501)	(0.058)
4	0.009**	0.002	0.002*	0.004**	0 005**	0.007***	0.000	0.000	0.000	−0.001
	(2.336)	(1.331)	(1.728)	(2.105)	(2 019)	(2.693)	(0.118)	(−0.195)	(−0.286)	(−0.462)
5	0.011***	0.003**	0.003***	0.005***	0 006**	0.008***	0.001	0.001	0.000	0.001
	(2.811)	(2.560)	(2.674)	(3.248)	(2.586)	(3.344)	(1.263)	(0.681)	(0.111)	(1.052)
6	0.011***	0.004***	0.004***	0.005***	0.006**	0.009***	0.002	0.002	0.002	0.003**
	(2.872)	(2.958)	(3.148)	(2.789)	(2.531)	(3.183)	(1.549)	(1.251)	(1.337)	(2.058)
7	0.012***	0.004***	0.003***	0.005***	0.006***	0.007***	0.000	−0.001	−0.002	0.001
	(3.223)	(3.615)	(3.182)	(3.285)	(3.025)	(2.826)	(0.162)	(−0.491)	(−1.191)	(0.771)
8	0.013***	0.005***	0.004***	0.005***	0.006***	0.008***	0.000	0.000	0.001	0.000
	(3.335)	(3.897)	(3.788)	(3.867)	(2 948)	(2.699)	(0.175)	(0.364)	(0.721)	(0.096)
9	0.015***	0.006***	0.005***	0.006***	.0 008***	0.011***	0.003**	0.004***	0.003**	0.005***
	(4.081)	(5.032)	(4.604)	(5.165)	(4 082)	(3.793)	(2.198)	(2.798)	(2.268)	(3.077)
High	0.016***	0.006***	0.006***	0.007***	.0 009***	0.012***	0.003**	0.003**	0.004**	0.004***
	(4.518)	(4.689)	(4.620)	(4.458)	(4.461)	(3.758)	(2.070)	(2.467)	(2.373)	(3.417)
High-Low	0.010***	0.007***	0.008***	0.008***	0.009***	0.006***	0.003*	0.005***	0.004**	0.005***
	(6.055)	(5.284)	(6.237)	(5.274)	(6.298)	(3.115)	(1.852)	(2.911)	(2.294)	(3.155)

At the end of each month t in the out-of-sample period (1994Q3-2023Q2, 116 quarters), we sort firms into deciles based on their most recent *CNN buy* probability (computed using firms' most recent eight quarterly earnings). For a firm to enter the portfolio formation at the end of month t, we require that announcement date of its most recent earnings to be within three months prior to portfolio formation. This table reports the average equal-weighted and value-weighted excess returns, as well as factor-adjusted returns (FF4 alpha, FF6 alpha, HMXZ5 alpha, and DHS3 alpha) in the subsequent month $t + 1$ for each *CNN buy* probability decile. Newey and West [1987] t-statistics with three lags are reported in parentheses, and ***, **, and * indicate significance at the 1%, 5%, and 10% level, respectively.

Table 10.
Alternative Modeling Choices in Training

Panel A: Two-dimensional CNN model

Alternative modeling choices	Difference in the 63-day BHAR between the highest and lowest CNNBP deciles					
	MAR	SAR	FF4	FF6	HMXZ5	DHS3
3 blocks; filter size = (5 × 5, 3 × 3, 3 × 3)	0.034***	0.034***	0.029***	0.031***	0.034***	0.033***
	(6.537)	(6.456)	(7.255)	(7.677)	(7.582)	(7.528)
3 blocks; filter size = (3 × 3, 3 × 3, 3 × 3)	0.035***	0.035***	0.031***	0.031***	0.035***	0.034***
	(7.414)	(7.367)	(8.516)	(9.154)	(8.932)	(8.678)
2 blocks; filter size = (7 × 7, 3 × 3)	0.033***	0.032***	0.027***	0.029***	0.030***	0.031***
	(7.532)	(7.425)	(7.300)	(7.971)	(8.371)	(7.859)
2 blocks; filter size = (5 × 5, 3 × 3)	0.035***	0.034***	0.029***	0.030***	0.032***	0.033***
	(7.910)	(7.802)	(7.752)	(8.560)	(8.879)	(9.057)
2 blocks; filter size = (3 × 3, 3 × 3)	0.034***	0.033***	0.029***	0.030***	0.032***	0.032***
	(7.715)	(7.506)	(8.312)	(8.871)	(8.687)	(9.005)
Dropout rate = 0	0.032***	0.031***	0.027***	0.027***	0.030***	0.031***
	(6.215)	(6.182)	(6.600)	(7.050)	(6.446)	(6.967)
Batch normalization = no	0.034***	0.033***	0.029***	0.030***	0.032***	0.032***
	(7.425)	(7.205)	(8.400)	(9.328)	(8.491)	(8.723)
Xavier initialization = no	0.035***	0.034***	0.030***	0.031***	0.034***	0.033***
	(6.433)	(6.372)	(7.476)	(7.917)	(7.539)	(7.394)
Activation = ReLU	0.035***	0.034***	0.031***	0.032***	0.033***	0.033***
	(7.343)	(7.162)	(8.594)	(9.466)	(8.164)	(8.050)

Panel B: One-dimensional CNN model

Alternative modeling choices	Difference in the 63-day BHAR between the highest and lowest CNNBP deciles					
	MAR	SAR	FF4	FF6	HMXZ5	DHS3
3 blocks; filter size = (1 × 7, 1 × 3, 1 × 3)	0.007*	0.007*	0.009**	0.009**	0.010***	0.010***
	(1.905)	(1.926)	(2.305)	(2.454)	(2.663)	(2.751)
3 blocks; filter size = (1 × 5, 1 × 3, 1 × 3)	0.006	0.005	0.005	0.006*	0.006*	0.007**
	(1.504)	(1.475)	(1.420)	(1.660)	(1.676)	(2.012)
3 blocks; filter size = (1 × 3, 1 × 3, 1 × 3)	0.002	0.001	0.003	0.003	0.001	0.003
	(0.773)	(0.567)	(1.268)	(1.140)	(0.359)	(0.972)
2 blocks; filter size = (1 × 7, 1 × 3)	0.001	0.000	0.001	0.001	0.002	0.002
	(0.430)	(0.013)	(0.210)	(0.230)	(0.549)	(0.839)
2 blocks; filter size = (1 × 5, 1 × 3)	0.001	0.001	−0.001	−0.002	0.001	0.000
	(0.311)	(0.234)	(−0.508)	(−0.574)	(0.329)	(0.041)
2 blocks; filter size = (1 × 3, 1 × 3)	0.004	0.005	0.005	0.004	0.008*	0.007
	(0.901)	(1.049)	(1.084)	(0.937)	(1.738)	(1.571)

This table reports the last row of Table 2 under different modeling choices in training. The baseline modeling choices in Table 2 are: two-dimensional CNN model; 3 blocks; filter size = (7 × 7, 3 × 3, 3 × 3); dropout rate = 0.5; batch normalization = Yes; Xavier initialization = Yes; activation = Leaky ReLU. See Table 2 and Section 6.2 for more details. Newey and West [1987] t-statistics with three lags are reported in parentheses, and ***, **, and * indicate significance at the 1%, 5%, and 10% level, respectively.

Table IA1.
***CNN Buy* Probability and Post-Earnings Announcement Drift: Double Sorts**

Panel A: Two-way sorting, controlling for pre-announcement return (RET[−30, −2])

	CNN buy probability quintiles					
RET[−30, −2] quintiles	Low	2	3	4	High	High-Low
Low	0.006	0.009	0.013	0.016	0.036***	0.030***
	(0.628)	(0.849)	(1.057)	(1.393)	(3.880)	(5.139)
2	0.000	0.004	0.009	0.016**	0.020***	0.020***
	(−0.030)	(0.676)	(1.332)	(2.385)	(3.404)	(5.544)
3	−0.004	0.004	0.004	0.012*	0.022***	0.026***
	(−0.811)	(0.813)	(0.691)	(1.677)	(4.187)	(6.515)
4	−0.009*	0.003	0.008	0.014**	0.023***	0.031***
	(−1.750)	(0.413)	(1.115)	(2.189)	(3.979)	(6.877)
High	−0.016**	−0.005	0.011	0.014	0.036***	0.052***
	(−2.203)	(−0.551)	(1.145)	(1.536)	(4.102)	(8.324)
High-Low	−0.022***	−0.015*	−0.002	−0.002	0.000	
	(−2.659)	(−1.781)	(−0.216)	(−0.313)	(−0.047)	

Panel B: Two-way sorting, controlling for earnings persistence (PERSIST)

	CNN buy probability quintiles					
PERSIST quintiles	Low	2	3	4	High	High-Low
Low	−0.005	0.001	0.005	0.012	0.029***	0.034***
	(−1.033)	(0.204)	(0.636)	(1.574)	(4.461)	(5.995)
2	0.000	0.008	0.015*	0.015*	0.026***	0.026***
	(−0.025)	(1.067)	(1.946)	(1.902)	(4.749)	(5.770)
3	−0.005	0.009	0.009	0.017**	0.029***	0.034***
	(−0.970)	(1.205)	(1.128)	(2.091)	(4.515)	(7.839)
4	−0.006	0.003	0.008	0.015**	0.025***	0.031***
	(−1.052)	(0.277)	(1.038)	(2.081)	(3.918)	(5.331)
High	−0.004	0.000	0.006	0.009	0.023***	0.027***
	(−0.738)	(−0.065)	(0.661)	(1.190)	(3.239)	(4.647)
High-Low	0.001	−0.002	0.001	−0.003	−0.006	
	(0.360)	(−0.424)	(0.309)	(−0.586)	(−0.983)	

Panel C: Two-way sorting, controlling for earnings volatility (VOL)

	CNN buy probability quintiles					
VOL quintiles	Low	2	3	4	High	High-Low
Low	−0.005	0.001	0.005	0.012	0.029***	0.034***
	(−1.033)	(0.204)	(0.636)	(1.574)	(4.461)	(5.995)
2	0.000	0.008	0.015*	0.015*	0.026***	0.026***
	(−0.025)	(1.067)	(1.946)	(1.902)	(4.749)	(5.770)
3	−0.005	0.009	0.009	0.017**	0.029***	0.034***
	(−0.970)	(1.205)	(1.128)	(2.091)	(4.515)	(7.839)
4	−0.006	0.003	0.008	0.015**	0.025***	0.031***
	(−1.052)	(0.277)	(1.038)	(2.081)	(3.918)	(5.331)
High	−0.004	0.000	0.006	0.009	0.023***	0.027***
	(−0.738)	(−0.065)	(0.661)	(1.190)	(3.239)	(4.647)
High-Low	0.001	−0.002	0.001	−0.003	−0.006	
	(0.360)	(−0.424)	(0.309)	(−0.586)	(−0.983)	

This table reports the average 63-day buy-and-hold market-adjusted return (MAR) after earnings announcements for portfolios formed based on the *CNN buy* probability quintiles and one of the seven firm characteristics including pre-announcement return (RET[−30, −2]), earnings persistence (PERSIST), earnings volatility (VOL), market capitalization (SIZE), operating profitability (OP), total accruals (TA), and operating accruals (OA) using independent two-way sorting in the out-of-sample period (1994Q3-2023Q2, 116 quarters). The quintile cutoffs for quarterly variables are based on the distribution of these variables in the previous quarter, and the quintile cutoffs for annual variables from July in year *t* to June in year *t+1* are based on the distribution of these variables at the end of June in year *t*. See Appendix for variable definitions. Newey and West [1987] *t*-statistics with three lags are reported in parentheses, and ***, **, and * indicate significance at the 1%, 5%, and 10% level, respectively.

Table IA1.

***CNN Buy* Probability and Post-Earnings Announcement Drift: Double Sorts (continued)**

Panel D: Two-way sorting, controlling for market capitalization (SIZE)

	CNN buy probability quintiles					
SIZE quintiles	Low	2	3	4	High	High-Low
Low	−0.018*	0.009	0.012	0.020*	0.056***	0.074***
	(−1.945)	(0.613)	(0.998)	(1.837)	(5.679)	(13.846)
2	−0.006	0.005	0.012	0.018*	0.043***	0.049***
	(−0.939)	(0.518)	(1.179)	(1.698)	(4.934)	(8.833)
3	−0.002	0.006	0.013	0.017*	0.022***	0.024***
	(−0.320)	(0.828)	(1.598)	(1.910)	(3.338)	(5.235)
4	0.001	0.003	0.003	0.006	0.011**	0.010**
	(0.100)	(0.575)	(0.685)	(1.201)	(2.338)	(2.508)
High	0.000	−0.001	0.002	0.004	0.009**	0.009*
	(−0.103)	(−0.258)	(0.696)	(1.365)	(2.203)	(1.840)
High-Low	0.017**	−0.010	−0.010	−0.016*	−0.047***	
	(2.110)	(−0.691)	(−0.887)	(−1.682)	(−4.815)	

Panel E: Two-way sorting, controlling for operating profitability (OP)

	CNN buy probability quintiles					
OP quintiles	Low	2	3	4	High	High-Low
Low	−0.017	−0.001	0.002	0.001	0.032***	0.049***
	(−1.474)	(−0.111)	(0.167)	(0.102)	(3.234)	(6.128)
2	−0.008	0.007	0.009	0.015*	0.030***	0.037***
	(−1.268)	(0.886)	(1.199)	(1.677)	(3.628)	(6.455)
3	−0.005	−0.003	0.016*	0.019**	0.028***	0.033***
	(−0.887)	(−0.492)	(1.952)	(2.503)	(4.795)	(8.142)
4	−0.004	0.007	0.008	0.016**	0.024***	0.028***
	(−0.787)	(1.220)	(1.218)	(2.552)	(3.784)	(5.145)
High	0.004	0.010*	0.012*	0.021***	0.023***	0.019***
	(0.810)	(1.762)	(1.887)	(3.406)	(3.486)	(3.438)
High-Low	0.021**	0.012	0.010	0.020**	−0.008	
	(2.017)	(1.082)	(1.288)	(2.468)	(−0.963)	

Panel F: Two-way sorting, controlling for operating accruals (OA)

	CNN buy probability quintiles					
OA quintiles	Low	2	3	4	High	High-Low
Low	−0.008	0.004	0.009	0.014	0.033***	0.041***
	(−1.071)	(0.363)	(0.910)	(1.376)	(4.291)	(7.451)
2	0.001	0.006	0.010	0.014*	0.027***	0.026***
	(0.237)	(0.846)	(1.325)	(1.909)	(4.200)	(4.839)
3	−0.004	0.004	0.015*	0.018**	0.025***	0.029***
	(−0.735)	(0.752)	(1.896)	(2.195)	(3.969)	(7.300)
4	−0.004	0.007	0.006	0.015**	0.027***	0.031***
	(−0.820)	(0.975)	(1.055)	(2.560)	(4.901)	(7.319)
High	−0.007	−0.003	0.004	0.009	0.027***	0.034***
	(−1.184)	(−0.362)	(0.452)	(1.235)	(4.195)	(7.041)
High-Low	0.001	−0.007	−0.006	−0.006	−0.006	
	(0.176)	(−0.848)	(−1.038)	(−0.813)	(−1.217)	

Table IA1.

***CNN Buy* Probability and Post-Earnings Announcement Drift: Double Sorts (continued)**

Panel G: Two-way sorting, controlling for total accruals (TA)

	CNN buy probability quintiles					
TA quintiles	Low	2	3	4	High	High-Low
Low	−0.008	0.012	0.009	0.009	0.037***	0.045***
	(−0.894)	(0.910)	(0.828)	(0.831)	(4.012)	(6.271)
2	−0.003	−0.001	0.011	0.017**	0.031***	0.033***
	(−0.415)	(−0.073)	(1.299)	(2.207)	(4.606)	(7.087)
3	−0.002	0.002	0.009	0.015**	0.027***	0.029***
	(−0.335)	(0.333)	(1.448)	(2.073)	(3.879)	(6.916)
4	−0.003	0.004	0.011*	0.021***	0.027***	0.030***
	(−0.656)	(0.902)	(1.781)	(3.112)	(4.469)	(6.644)
High	−0.005	−0.002	0.004	0.008	0.017***	0.022***
	(−0.956)	(−0.348)	(0.538)	(1.282)	(2.833)	(4.321)
High-Low	0.003	−0.014	−0.005	−0.001	−0.021***	
	(0.375)	(−1.449)	(−0.811)	(−0.169)	(−3.186)	