Hindawi
Journal of Mathematics
Volume 2021, Article ID 3870657, 10 pages
https://doi.org/10.1155/2021/3870657

Research Article

Deep Learning-Based Economic Forecasting for the New Energy Vehicle Industry

Bowen Cai

Beijing Jiaotong University, Beijing 100044, China

Correspondence should be addressed to Bowen Cai; brooven@cumt.edu.cn

Received 22 November 2021; Revised 8 December 2021; Accepted 11 December 2021; Published 31 December 2021

Academic Editor: Naeem Jan

Recently the issues of insufficient energy and serious air pollution around the world have been rising. Henceforth, there is a need to carry out a research of new energy. Soon, new energy vehicles will be the mainstream trend, which can not only reduce the burden of consumers due to rising fuel prices but also solve the air pollution problem caused by the exhaust emissions of fuel vehicles. With the rapid development of science and technology, deep learning continues to make breakthroughs, and, in the field of economy with huge information data, we have more powerful weapons available to predict and research important economic data with infinite value, which can not only provide reference information to policy makers but also help enterprises and even economic markets to develop more healthily and sustainably. Therefore, this article uses deep learning algorithms to forecast and analyze the new energy industry, starting from the financial information released by new energy vehicle companies in their annual reports, in order to make basic judgments and help policy makers and enterprises in the new energy vehicle industry.

1. Introduction

Since the industrial revolution, a series of ecological and environmental problems [1] brought about by excessive human production and consumption [2] have come to a point where we have to pay attention to them and take measures to remedy and correct them. In this context, the concept of low-carbon development [3], which balances environmental protection and development, has gradually emerged and become a global trend. From personal lifestyles to government policies, the shadow of low-carbon development can be seen everywhere. Economic and social development and environmental protection have always been a dilemma for human beings [4], especially after the industrialization era. After years of practical exploration, mankind has finally found the road to low-carbon development, which is to integrate development into the construction of ecological civilization and environmental protection into economic development. From the current energy production and consumption situation, the development of new energy and energy-efficient technologies and products is an important step to ensure the sustainable development of the global economy.

The automotive industry is not only a pillar industry driving the national economy [5] but also a high-consumption and high-emission industry [6], which plays an important role in macroeconomic development and low-carbon economic transformation. The rapid development of China's economy and market has attracted many automobile manufacturers from developed countries to come to China, not only bringing advanced products and technologies but also making China face huge pressure of environmental protection, such as the oil crisis, urban environmental pollution, and traffic deterioration. The traditional development mode of automobile industry has obvious obstructive effect on economic sustainability, so the automobile industry must transform to low carbon with technological innovation as the core. New energy vehicles are the inevitable product of the development of the times, and the development of new energy vehicles is the inevitable path for the global automotive industry.

The industrial change triggered by the development of new energy vehicles will be a complete reshuffle; however, the development of the new energy vehicle industry will also be accompanied by a lot of uncertainty, especially because new energy vehicle companies have significant differences from the traditional automobile industry. For this reason, both the policy makers of the new energy industry and the practitioners of the new energy vehicle enterprises always need to assess and control the industry accurately. In the current fast-moving 21st century, scientists and researchers have provided us with many new techniques and analytical tools to study the industrial economy. Not only has deep learning become one of the most important key technologies in the field of AI, but also it has attracted a lot of attention from researchers in related business economies [7]. Deep learning has a wide range of applications in various fields and industries [8], driving industrial innovation and breakthrough development. In the field of economy and finance, people are also increasingly aware of the importance of economic data to enterprises, which determines the development and future of enterprises [9]. As the volume of economic data increases dramatically and the forms of data become more and more diversified, the approach of deep learning provides a new research idea of finding patterns from big data and learning the potential features behind the data through deep learning models [10]. So it is of great importance to apply deep learning to the field of economy and finance.

The main theme and focus of this article are on the use of deep learning methods to analyze and predict the corporate financial reports of new energy vehicle companies, especially the net profit of a company and the net cash flow from operating activities. The end-product of the operations of an enterprise is the net profit, that is, the principle sign of the working productivity of the enterprise. Net profit is utilized to replicate the operation and profitability of the enterprise. However, the value of the revenues should be evaluated along with other comprehensive factors. The most significant factor is the net cash flow from operating activities in the cash flow statement. Free income is created in the business exercises based on guaranteed normal operation and consistent/reliable reinvestment of the enterprise, which guarantees the free circulation and investment in a safe situation and is the actual fund that can be called by the enterprise, and operating cash flow is the most important part that constitutes free cash flow. The analysis of net income and net cash flow from operating activities not only reflects a company's ability to operate on a sustainable basis but also conveys information about the possible existence of abnormalities in the long-term operation of the company. By analyzing the matching ratio between net profit and operating cash flow, we can make a basic judgment on the sustainable operating ability and competitiveness of the new energy vehicle company.

The organization of the rest of the paper is as follows: Section 2 reviews some of the related literature. In the third section, the deep learning prediction models are build, which are based on GRU and CNN. Section 4 presents the experimental analysis and the results achieved by the trials. In Section 5, the final results and achievements are deliberated as a conclusion.

2. Related Work

2.1. Data Resources. Nowadays, both business operators and ordinary investors attach great importance to the data of economic and financial fields such as company annual reports, stocks, and funds, and, accordingly, major financial websites collect and display various announcements and data information. In this paper, we need to obtain the annual reports of companies in the new energy automobile industry and crawl them to further process and analyze the important information in the annual reports.

The process of crawling PDF financial statements from financial websites is roughly as follows: (1) Using crawler technology, we obtain the IDs of all companies, generate Uniform Resource Locators (URLs) for the main pages of all company information, and put them into the queue of URLs to be crawled. (2) The crawler reads from the queue of URLs to be crawled in turn, resolves the URLs through a domain name resolver (DNS), and converts the link addresses to the IP address of the web server, and then the web page can be downloaded. (3) We use Python's Request module to crawl to the main page of the company's information, and the URL of the company's annual report in PDF format is in the web page content. According to the characteristics of the annual report URL, generate a regular expression, and then use the Re module to search for the company's annual report URL. (4) Finally, use the Request module to download the annual report in PDF format, and use the company ID as the file name for storage for subsequent use.

2.2. PDF Analysis and Table Data Extraction. In this article, we want to study the sustainability of new energy vehicle companies, and the important economic data needed are net profit and net cash flow from operating activities, so we need to extract the income statement and cash flow statement from the PDF annual reports of listed companies for subsequent research and analysis. We need to get a number of companies as well as multiple years of data, so you want to efficiently and accurately analyze the PDF, and you need to crop a long PDF, according to keywords to get the specific location of the target form, making the analysis more convenient. As shown in Figure 1, the specific steps will be explained in detail below.

2.2.1. Location Information Based on Keywords. In order to streamline and intercept PDF to obtain more accurate positioning and analysis, to improve efficiency, the first step is the need to get the target data according to the keyword page number and location. This paper uses Java to write ITextpdf method to locate the target data. When we provide the PDF file path, store the target location information to get the file path, and match the content of the target with keywords of

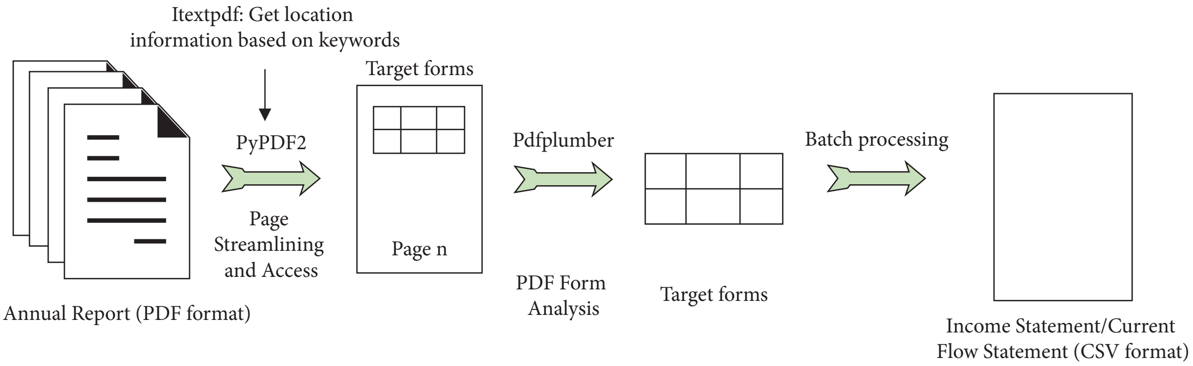
Itextpdf: Get location information based on keywords
PyPDF2
Page Streamlining and Access
Target forms
Page n
Pdfplumber
PDF Form Analysis
Target forms
Batch processing
Annual Report (PDF format)
Income Statement/Current Flow Statement (CSV format)

Figure 1: Data extraction process.

the three parameters, we can achieve the target location and page number, and batch management for subsequent use.

2.2.2. PDF Page Streamlining and Interception. In this paper, we use the Python library PyPDF2 to streamline and intercept the pages of the annual reports of listed companies with a large number of pages. According to the previous section of the method to obtain the target location and page number, call addPage() and other methods to add the loop to the object created by the PdfFileWriter class, in the output stream to add PDF pages; finally, we want to streamline and intercept the target page written to a new PDF file and want to achieve batch operations and management in order to follow the analysis of the work.

2.2.3. PDF Form Parsing and Data Batch Grabbing. After obtaining the streamlined PDF, we were able to target the important economic data in the tables and use the Pdfplumber technique in the Python third-party library to parse the PDF and extract the income statements and cash flow statements of several new energy companies and multiple years. Finally, the extracted results were modified and collated to obtain the income statements and cash flow statements for the annual reports of the new energy companies needed for this study and sorted by company and chronological order according to the different industries and collated into an easy-to-use CSV file.

2.3. Experimental Data Preprocessing and Sample Generation. After parsing and extracting the data from the PDF annual reports of the new energy vehicle companies, the data is obtained in a convenient CSV format for use. In order to obtain the dataset for the final model requirements, this paper also requires preprocessing of the data, including various data cleaning, normalization, feature selection, and finally sample generation for the experimental requirements.

2.3.1. Data Cleaning. Data cleaning is an essential and critical step in data analysis [11], and the quality of the data obtained after cleaning will directly affect the subsequent model effects and experimental conclusions. In the first step of missing value cleaning, the proportion of missing values for each feature is firstly counted. If there are too many missing values, the validity of the features is lost, so the feature columns with more than 80% missing values are directly deleted in this paper. Then the missing values are filled in. As each economic indicator has a certain standard, if it is filled in randomly, it may cause the data to be unrealistic, and there are many 0 values in many features, so, according to experience, this paper fills in the missing values with 0. The second step is to clean the formatted content. The format of the data will affect the data import as well as the experimental process; for example, in this paper, the accounting time in the income statement and cash flow statement is cleaned in format.

2.3.2. Normalization. In real data, different features have different ranges of values, and thus it may happen that individual features with larger values in the feature space have a dominant effect on the sample. In order to have all features in the same scale, it is important to map them to the same scale so that the accuracy of the model can be improved as well as the speed of the fit, so this experiment is to normalize the sample [12]. We use the maximum-minimum normalization method, which scales the dataset equally and ends up being represented as a value between [0, 1], as shown in the following equation:

$$X_{\text{scale}} = \frac{x - \min}{\max - \min}. \tag{1}$$

2.3.3. Feature Selection. Feature selection is a very important step in the actual processing of data [13]. Not only does it help to further understand the characteristics and relationships between data features but it also reduces feature dimensionality, reduces overfitting to improve model generalization, and improves the performance of algorithms and models. We have experimentally chosen the Randomized Lasso algorithm of the top-level feature selection algorithm, which is a kind of stability selection. Lasso regression means that the penalty function used in solving the objective function of the regression coefficients is the L1 parametrization, as shown in the following equation:

$$\widehat{\beta}_{\text{Lasso}} = \arg\min_{\beta\in R^d}\left(\|Y - X\beta\|^2 + \lambda \sum_{j=1}^{d} \left|\beta_j\right| \right). \tag{2}$$

2.3.4. Sliding Sample Generation. Through the above data preprocessing, the effective features are selected using the feature selection algorithm; this paper uses a time span of 2 years as a sample and sliding order to select samples, with each company individually divided for sample generation. The 2-year time span is the best time span to be tested. Through analysis, if the time span is too small, it will not be easy to grasp the trend pattern, and if the time span is too large, the number of samples will be too small, which will make the prediction effect poor. The specific operation process is shown in Figure 2, and finally a dataset sample with multiple characteristics over multiple time spans can be obtained. As our aim is to study the data in the annual reports of new energy vehicle companies, not only are there a limited number of new energy vehicle companies, but also a limited number of years of annual reports can be obtained, so a sliding approach to sample generation not only allows for a more detailed delineation of features while maintaining the time-series characteristics but also increases the number of samples and is suitable for smaller datasets.

3. Building Deep Learning Prediction Models Based on GRU and CNN

Deep learning is applied to the annual reports of new energy vehicle companies, the prediction and analysis of the net profit in the income statement, which reflects the operation and profitability of the enterprise, and the net cash flow from operating activities in the cash flow statement, which reflects the quality of profitability. Finally the basic judgment of the sustainability of individual enterprises and the industry as a whole is of certain research and practical significance.

For the prediction of important economic data with time-series characteristics, this paper first uses machine learning models that have the advantages of being simple, efficient, and stable, such as the MLR model [14], which is the simplest multiple linear regression model that can handle linear relationships well, and the SVR model [15], which has multiple kernel functions that can be applied to data with a variety of characteristics. However, as they do not adequately consider and reflect the time-series characteristics of the data nor can they extract the underlying features between the data in depth, the prediction results are rather mediocre. However, these two problems that arise can be well improved using deep learning methods. The GRU model [16], a variant of the recurrent neural network (RNN) gated recurrent unit [17], is particularly suitable for time-series data and also addresses the long-time dependence problem that is important for predictive regression results, coupled with the convolutional neural network (CNN) model that can effectively extract potentially important features between data. The combination of GRU and CNN models [18] is therefore ideally suited to our prediction and regression of corporate financial reports.

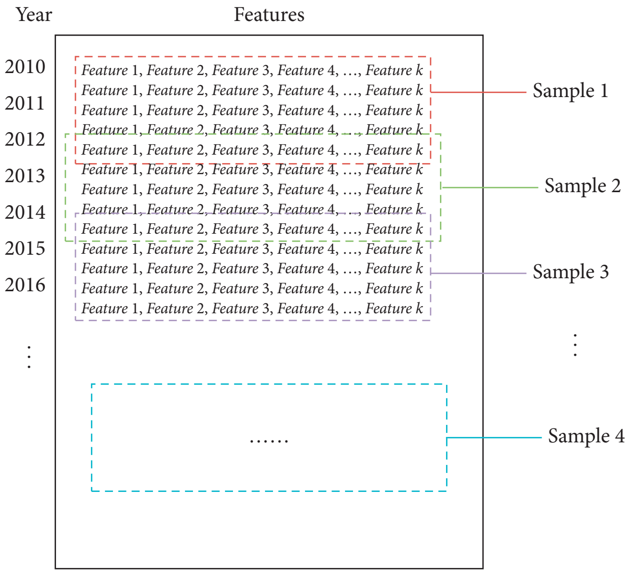
Year
Features
2010
2011
2012
2013
2014
2015
2016
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Feature 1, Feature 2, Feature 3, Feature 4, ..., Feature k
Sample 1
Sample 2
Sample 3
Sample 4
......

Figure 2: Sliding sample generation.

3.1. CNN Model. CNNs have great advantages in extracting local high-level abstract features due to their local perception, weight sharing, and pooling layer downsampling [19]. CNN uses forward propagation to calculate the output values and gradient descent and backpropagation to train the model, adjusting the weights, biases, and so forth. The forward calculation formula for the convolutional layer of the CNN is as follows:

$$X_j^l = f\left(\sum_{i\in M_j} X_i^{l-1} * K_{ij}^l + b_j^l \right), \tag{3}$$

where K is the convolution kernel, which contains $n * m$ weights. The feature mapping map of layer $L-1$ is convolved with K in a dot product operation and then summed and a bias is added to prevent overfitting. Finally, the activation function is used to obtain the feature map of the Lth layer. In order to reduce the computational complexity and extract the main features, the feature mapping map needs to be compressed, so the downsampling formula for the pooling layer is as follows:

$$X_j^l = f\left(\beta_j^l \text{down}\left(X_i^{l-1} \right) + b_j^l \right), \tag{4}$$

where down() is a function that samples the feature values of layer $L-1$, for example, to find the maximum value and the average value, and then adds a bias to the activation function to obtain a feature map of the specified size after compression of the Lth layer. When training in the training set using the backpropagation algorithm, weights and biases are obtained, which are continuously adjusted to meet the training objectives. These set weights and biases can then be used to obtain regression predictions when the test set is tested.

The exact operation of a CNN is roughly shown in Figure 3.

In the input layer, each feature in the input sample is treated as a neuron. After setting the number of convolutional kernels, size, convolutional move step, and so

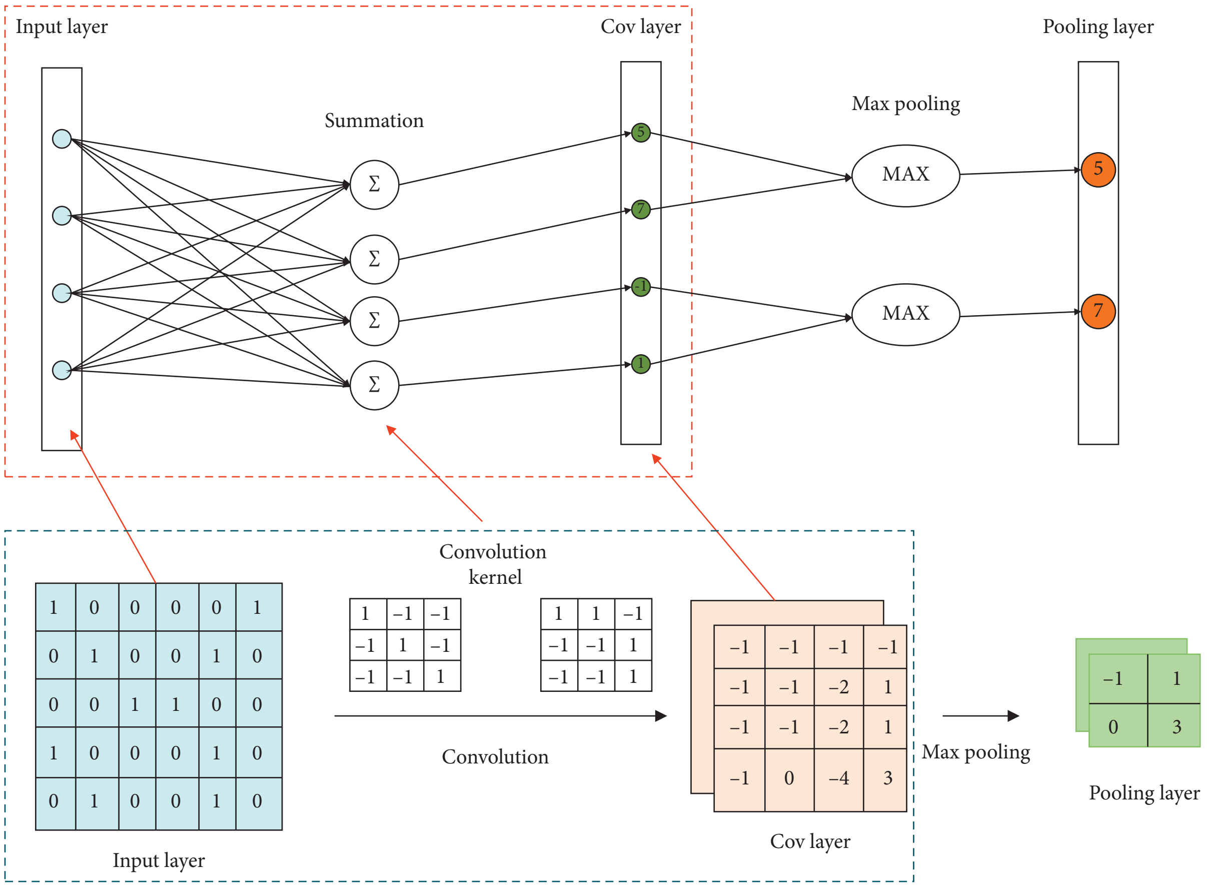

FIGURE 3: The schematic diagram of CNN.

forth and using the weight parameters of the convolutional kernels calculated by the backpropagation algorithm, the convolutional operations and summation are performed with the neurons in the input layer to obtain the feature maps of the convolutional layer composed of the feature values. For the pooling layer, we use the maximum pooling algorithm, which sets the size of the pooling kernel and the sliding step to calculate the new feature values and form the compressed feature maps of the pooling layer.

3.2. GRU Model. The GRU model not only has the power of an RNN for time-series data but also has the advantage of an LSTM network that is good at dealing with long- and short-term dependencies. The GRU model works as shown in Figure 4 and equations (5) to (12).

GRU has two important gates, the Update Gate and the Reset Gate. As shown in Figure 4, z_t represents the Update Gate. Equation (5) connects input x_t and the two vectors of h_{t-1} of the previous hidden layer and then performs a dot product with the weight matrix W_z. Finally, the result is compressed between 0 and 1 by the sigmoid activation function. The closer z_t is to 0, the more information from the previous hidden layer should be forgotten in that hidden layer, and the closer it is to 1, the more information needs to be retained in that hidden layer. r_t represents the Reset gate, which is similar to z_t, except that the weight matrix is W_r, as shown in equation (6). The more r_t is closer to 0, the more information from the previous hidden layer should be forgotten in the current memory content, and the closer it is to 1, the more information needs to be retained in the current memory content. The closer it is to 1, the more information needs to be in the current memory content that will continue to be retained.

$$z_t = \sigma\left(W_z \bullet \left[h_{t-1}, x_t\right]\right), \tag{5}$$

$$r_t = \sigma\left(W_r \bullet \left[h_{t-1}, x_t\right]\right). \tag{6}$$

In Figure 4, $\widetilde{h}_t$ represents the candidate hidden layer state (Candidate Activation), which is expressed by multiplying the previous moment hidden layer h_{t-1} with the Reset gate r_t, which is used to determine how much of the previous moment hidden state is to be forgotten in the current memory content. This is then linked to the input x_t matrix, dotted with the weight matrix $W_{\widetilde{h}}$, and finally scaled to −1 to 1 by the tanh activation function. $\widetilde{h}_t$ finally stores all the important information recorded by GRU by calculating the important information in the hidden layer at the previous moment as well as the important information in the current input, as shown in the following equation:

$$\widetilde{h}_t = \tanh\left(W_{\widetilde{h}} \bullet \left[r_t * h_{t-1}, x_t\right]\right). \tag{7}$$

Finally, it is necessary to calculate the current moment hidden layer h_t, as expressed in equation (8) by multiplying

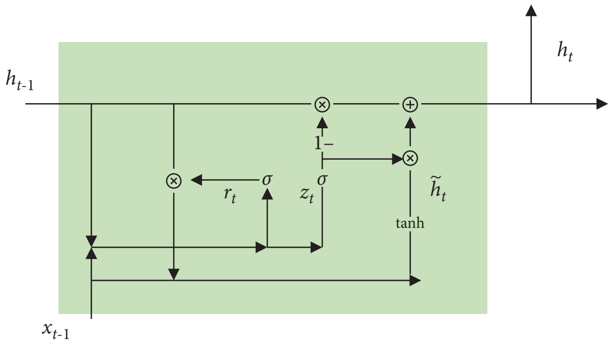

Figure 4: The structure diagram of GRU.

all the important information $\tilde{h}_t$ in the candidate hidden layer state with the Update gate to obtain the updated important information that needs to be obtained. The information that continues to be retained with the hidden layer at the previous moment is summed to obtain the hidden layer h_t. Finally, by dotting the product with the weight matrix W_o and then inputting the sigmoid activation function, result y_t is obtained, as shown in equation (9).

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t, \tag{8}$$

$$y_t = \sigma(W_o \bullet h_t). \tag{9}$$

The above process is the GRU forward propagation process, where W_r, W_z, $W_{\tilde{h}}$, and W_o are the parameters to be trained, and W_r, W_z, and $W_{\tilde{h}}$ are stitched together by two vector matrices, respectively. This is shown in the following equations:

$$W_r = W_{rx} + W_{rh}, \tag{10}$$

$$W_z = W_{zx} + W_{zh}, \tag{11}$$

$$W_{\tilde{h}} = W_{\tilde{h}x} + W_{\tilde{h}h}. \tag{12}$$

Then the model is trained by backpropagation algorithm and gradient descent regularization. The parameters, such as weights and biases, are adjusted, updated, and iterated until the loss converges; that is, the training is completed, and the test set data can be fed into the GRU for prediction.

3.3. CNN-GRU Model Construction. Taking advantage of the efficient extraction of potential relationships between features by CNN and the powerful ability of GRU to handle long- and short-term dependencies of time-series data, the construction process of CNN-GRU [20], the first model combining CNN and GRU in this paper, is shown in Figure 5.

The first input sample size is a matrix of $5 * k$, where 5 represents a sample of 5 years as a time step. k represents the number of features of the sample after feature selection. 25 features from the income statement of the listed company are selected for training when predicting net profit, so k is 25. When predicting net cash flow from operating activities, the cash flow statement is used to obtain 29 features after feature selection. After feature selection, 29 features are available, so k is equal to 29 currently.

Then the sample data is input into CNN for abstraction of local features, and the CNN of Convolution2D is selected. Firstly, the first convolutional layer is entered for training, in which the convolutional kernel size is set to $2 * 3$ rectangle, and the number of convolutional kernels filters is set to 32, which can get 32 layers of feature maps, used to extract different kinds of potential relationships and features; the convolutional step strides are equal to 1. By sliding the convolution from top to bottom in left-to-right order, the padding attribute is set to same, because the boundary information is chosen to be discarded when padding is equal to another valid value. When set to same, the boundary information is preserved, and the input and output sizes are kept consistent because the boundary is complemented by adding 0. This layer is learned using the ReLU activation function, which is easier to learn to optimize, and is then fed to the second convolutional layer in order to extract the features in more depth. In this layer, the convolutional kernel size is set to a rectangle of $3 * 5$, and the number of filters is set to a larger 64 to store the extracted features, resulting in thicker 64-layer feature maps, with the convolutional step length strides still equal to 1. In the pooling layer, the window size of the pooling kernel is set to a $2 * 2$ rectangle, the window sliding steps are set to 2, and the ReLU activation function is used. After pooling, the number of layers of 64 is unchanged but the features are compressed to obtain the feature maps, and, finally, a dropout mechanism is used to temporarily disable the neurons in the hidden layer at a random rate of 0.25 to effectively alleviate the overfitting situation.

After the features of the samples are extracted by the CNN network, the extracted features can be tiled in temporal order by the TimeDistributed(Flatten()) layer to preserve the temporal order of the features. The features are then fed to the GRU layer, which takes advantage of the GRU's ability to process temporal features efficiently and to handle long- and short-term dependency problems. Finally, a dense fully connected layer is connected, and the final prediction results are output using linear activation function.

3.4. CNN + GRU Fusion Model Construction. The last column of features in the sample of this paper is called direct features, and the other columns of features are called indirect features. In other words, when forecasting net profit through the past income statements of new energy vehicle companies, the last column of characteristics is the net profit of previous years, and the other columns are the indirect characteristics of operating income and sales expense of previous years. When forecasting the operating cash flow from operating activities through the cash flow statement, the last column of features is the operating cash flow of previous years, and the other columns are indirect features such as tax refunds received in previous years and net increase in cash and cash equivalents. Since different features have different potential characteristics, in this paper, we

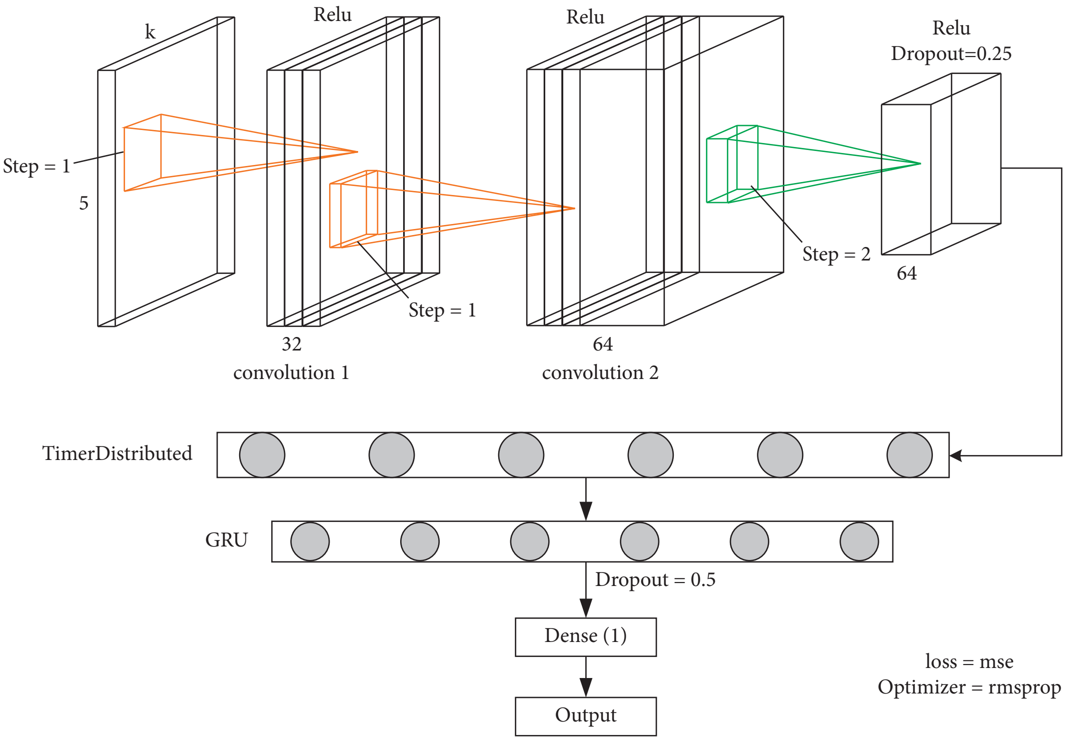

Figure 5: The overview of CNN-GRU.

build different models for each of the different characteristics for training and then finally merge and splice the different abstract features obtained from each training and then carry out the next training together to achieve better and more ideal prediction results. The construction process of our second model combining CNN and GRU, that is, CNN + GRU with CNN and GRU fusion [21], is shown in Figure 6.

First, the sample of size $5 * k$ matrix is input, 5 still represents the best time step of 5 years after experimental validation, k represents the number of effective features after feature selection, k takes 25 when predicting net profit, and k equals 29 when predicting operating cash flow. Then the samples have to be divided into samples, and the first $k - 1$ indirect features are input into the CNN model as shown in the left-dashed box for training, and the last column of direct features FEATURE k is input into the GRU model as shown in the right-dashed box for training.

In the left CNN model, in order to extract the potential high-level abstraction relationship between indirect features, firstly enter the first convolution layer Convolution2D, set the size of convolution kernel as a rectangle of $2 * 3$, and perform the convolution operation sequentially in the order of top to bottom, left to right, and step size of 1. The number of 32 convolution kernels is set to extract different kinds of features, and finally the feature map of 32 layers can be obtained in this layer. Then we enter the second convolution layer of Convolution2D, the size of the convolution kernel in this layer is $3 * 5$, and the number of convolution kernels is increased to 64 to store the extracted features. The activation functions of both convolutional layers use the efficient ReLU activation function. In order to reduce the parameters to extract more important and higher-level features and relationships, a pooling layer is connected next, using a maximum pooling algorithm, each time selecting the maximum value in a matrix with a pooling kernel size of $2 * 2$, with the aim of reflecting the extraction of the most important features in it. Then use Dropout mechanism, and set the parameter to 0.5, which randomly makes some neurons in the hidden layer of the neural network fail, in order to improve the model generalization ability and other effects. Finally, the flatten layer is used to tile the feature information extracted by the CNN for subsequent training.

In the GRU model on the right, for the predicted values of previous years, they are directly input into the GRU model that is very suitable for processing time-series data and extracting the long- and short-term dependencies in the time-series data enables better analysis of this part of the features. So the last column of direct features of the sample is input to the GRU with 128 neurons, and the input shape is set to (5,1), where 5 represents a time step of 5 years and 1 represents a column of features. The return sequence is set to False to ensure that only the last target value is input at the end after training according to the whole time step. The same Dropout mechanism with parameter of 0.5 is used after the GRU training is completed. Finally, the important features extracted by the GRU model are obtained.

After the CNN and GRU models are trained to extract different kinds of features, respectively, it is necessary to use the most critical Merge fusion mechanism provided by Keras to stitch the two models together and set the parameter as concat to get all features after the whole sample extraction.

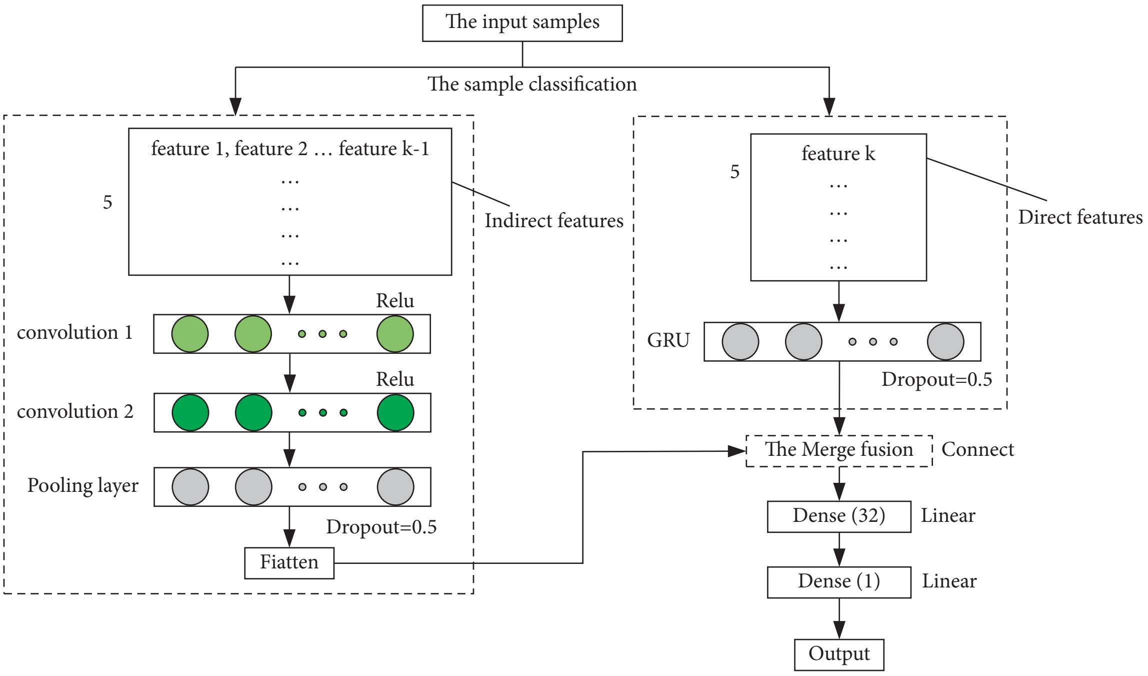

Figure 6: The overview of CNN + GRU.

Then all features are input to the dense fully connected layer for training together and finally input to a fully connected layer for final regression prediction using the linear activation function.

4. Experimental Results and Analysis

4.1. Evaluation Criteria. In this paper, in order to evaluate the experimental results, RMSE and $R2$ are firstly selected as evaluation criteria. Both of these rubrics are common criteria for judging the results of prediction and regression analysis.

Root Mean Square Error (RMSE) is the value obtained after the root of the variance between the predicted value and the true value, and the RMSE is a clearer measure of the prediction result than the MSE and better represents the deviation between the true value and the predicted value, because the root of the error unit can be kept constant. The calculation formula is shown in (13), where $\hat{y}_t$ represents the predicted value and y_i represents the true value.

$$\mathrm{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(y_i - \hat{y}_i\right)^2}. \tag{13}$$

The coefficient of determination (R-squared, $R2$), also known as goodness of fit, is an indicator that varies between 0 and 1. $R2$ reflects the predictive effect by the change in the data, comparing the predicted value to the actual mean value only, as shown in the following equation:

$$R2 = 1 - \frac{\sum_i\left(\hat{y}_i - y_i\right)^2}{\sum_i\left(\overline{y_i} - y_i\right)^2}. \tag{14}$$

The numerator represents the predicted value minus the sum of the true values squared, like the mean squared, that is, all errors in the trained model prediction. The denominator is the mean value minus the sum of the squares of the true values, which is similar to the variance, that is, the guess is the mean of the true values. $r2$ takes the value range [0,1]; when the result is 0, it means that the model prediction deviates a lot; if $r2$ is 1, it means that the fit is quite accurate, so the closer $r2$ is to 1, the better the model prediction result is. In addition, the larger $R2$ means that the model prediction results are closer to the true value.

4.2. Model Experimental Results. In this paper, we built models and conducted experiments based on Python 3.6 using the high-level neural network module Keras, with Theano as its back end. We built the MLR and support vector regression (SVR) models of the deep learning approach and the NN, CNN, RNN, LSTM, and GRU models of the general neural network, convolutional neural network, recurrent neural network, long short-term memory network, and gated recurrent unit of the deep learning approach as comparison models for experimental comparison. When building the machine learning model, the third-party module Sklearn is used, and, in the MLR model, the samples are tiled and input to the model LinearRegression() for training. In the SVR model, the SVR is initialized using the radial basis kernel function. When building the deep learning model, the Sequential() model is used, and the specific model is created by adding the add() function and

Table 1: Experimental results of forecasting net profit.

Dataset	Model	*R*2	RMSE
Net income dataset	MLR	0.6048	0.0987
	SVR	0.6900	0.0759
	NN	0.7241	0.0203
	CNN	0.7907	0.0575
	RNN	0.8046	0.0230
	LSTM	0.8731	0.0195
	GRU	0.9104	0.0180
	CNN-GRU	0.8919	0.0161
	CNN + GRU	**0.9059**	**0.0150**

passing a series of different layers. In the NN model, the samples are tiled and fed to only one fully connected layer for training. In the CNN model, two convolutional layers and one maximum pooling layer are used. In the RNN model, it is created by a layer of SimpleRNN (128). The LSTM model and the GRU model are created by LSTM/GRU (128, input_shape, return_sequences = False), respectively. These comparison models were trained and tested with the CNN-GRU model and CNN + GRU fusion model constructed in this paper, using the same training and test sets. The experimental results are shown specifically in Tables 1 and 2.

Table 2: Experimental results of forecasting operating cash flow.

Dataset	Model	*R*2	RMSE
Operating cash flow dataset	MLR	0.4298	0.1357
	SVR	0.3113	0.1983
	NN	0.8902	0.0203
	CNN	0.8619	0.0239
	RNN	0.7405	0.0266
	LSTM	0.8939	0.0197
	GRU	0.9104	0.0173
	CNN-GRU	0.9202	0.0158
	CNN + GRU	**0.9452**	**0.0114**

The following summary can be made through Tables 1 and 2 evaluation index results data:

(1) From the overall view of each evaluation index, machine learning MLR and SVR fit the least well, but SVR is slightly better than MLR in general. Among the deep learning algorithms, the general neural network (NN) performs generally, and CNN and RNN have improved significantly, while LSTM and GRU can achieve more satisfactory results in general. However, the CNN-GRU and CNN + GRU fusion models constructed in this paper basically achieve the most desirable results in each evaluation index; in particular, the CNN + GRU fusion model has the best overall performance in each dataset.

(2) From the *R*2 goodness-of-fit evaluation index, the CNN-GRU and CNN + GRU models can reach 0.8 or even 0.9 or more on both datasets, which also has about 5% improvement over the best results in other comparison models; and the *R*2 value of the CNN + GRU model will be about 2% higher than that of CNN-GRU in general.

(3) From the Root Mean Square Error (RMSE), the CNN-GRU and CNN + GRU models can reach about 0.015 on each dataset and achieve the lowest RMSE value.

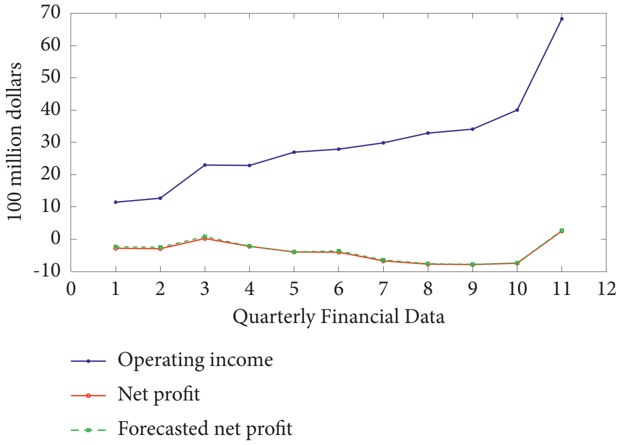

Figure 7: Financial real value and forecast of a new energy vehicle company.

4.3. Analysis of Model Results. In order to carry out the comparison, we analyzed the actual data of a new energy company and proposed CNN + GRU fusion model for prediction. Figure 7 portrays the graphical comparison between the net income, net profit, and the proposed forecasted net profit. The data in Tables 1 and 2 and the following graphs show that the net income is greater than 0, the net profit gradually changes from negative to positive, and the overall trend is up. The net profit predicted by the CNN + GRU fusion model also fits well with the actual net profit and can predict the trend well. This new energy vehicle enterprise has more room for development.

5. Conclusion

In this paper, we firstly parse the annual reports of listed companies in PDF format, through ITextpdf, PyPDF2, and Pdfplumber technologies, to parse and extract PDF tables in a more standardized and accurate way and achieve the target data in batch for multiple years. Then the data is cleaned and normalized, and the Randomized Lasso algorithm is used for feature stability selection. We study the financial statements of new energy vehicle companies, and the data available to each company is limited, so this paper performs sliding sample generation to achieve a reasonable increase in data samples and maintain the temporal order among data features and finally obtains a dataset sample with multiple features over multiple time spans.

We have successfully constructed two regression prediction models, CNN-GRU and CNN + GRU fusion. By conducting experiments on different datasets and comparing with other models such as MLR, SVR, NN, RNN, and LSTM, the two models constructed in this paper achieve more satisfactory results on each dataset, with $R2$ values reaching 0.8 or even 0.9 or more and RMSE basically around 0.015, indicating that the trend in prediction is basically consistent with the actual situation, and in a comprehensive view, the CNN + GRU fusion model has the best overall performance.

In the actual demand, there are many new energy vehicle companies that have been established recently, so we can do further research and analysis on this issue in the subsequent study. Other data and information in the annual report are also very rich, and more data and indicators can be studied in more depth to provide more favorable help for investors and enterprises in the new energy vehicle industry based on deep learning.

Data Availability

The datasets used during the current study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The author declares that he has no conflicts of interest.

References

[1] S. Czaja, “Disputes over the concept of ecological economics and environmental economics in the light of the works of professor tomasz zylicz,” *Central European Economic Journal*, vol. 8, no. 55, pp. 201–211, 2021.

[2] M. Ahmed, M. Rauf, Z. Mukhtar, and N. A. Saeed, “Excessive use of nitrogenous fertilizers: an unawareness causing serious threats to environment and human health,” *Environmental Science and Pollution Research*, vol. 24, no. 35, pp. 26983–26987, 2017.

[3] Q. Song, N. Zhou, T. Liu, S. A. Siehr, and Y. Qi, “Investigation of a “coupling model” of coordination between low-carbon development and urbanization in China,” *Energy Policy*, vol. 121, pp. 346–354, 2018.

[4] Q. He, Z. Lei, R. An et al., “Environmental protection and social development: an optimization perspective from two polarities,” *E3S Web of Conferences*, vol. 269, Article ID 01001, 2021.

[5] A. G. Olabi, T. Wilberforce, and M. A. Abdelkareem, “Fuel cell application in the automotive industry and future perspective,” *Energy*, vol. 214, Article ID 118955, 2021.

[6] C. Zi and S. Zhenyao, “Industrial carbon emissions status and analysis based on energy consumption,” *Chinese Journal of Population Resources and Environment*, vol. 9, no. 1, pp. 33–41, 2011.

[7] Y. Lecun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[8] J. Ahmad, H. Farman, and Z. Jan, “Deep learning methods and applications,” *Deep Learning: Convergence to Big Data Analytics*, pp. 31–42, Springer, Singapore, 2019.

[9] R. Culkin and S. R. Das, “Machine learning in finance: the case of deep learning for option pricing,” *Journal of Investment Management*, vol. 15, no. 4, pp. 92–100, 2017.

[10] O. B. Sezer, M. U. Gudelek, and A. M. Ozbayoglu, “Financial time series forecasting with deep learning: a systematic literature review: 2005–2019,” *Applied Soft Computing*, vol. 90, Article ID 106181, 2020.

[11] T. Wang, H. Ke, X. Zheng, K. Wang, A. K. Sangaiah, and A. Liu, “Big data cleaning based on mobile edge computing in industrial sensor-cloud,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1321–1329, 2019.

[12] T. Y. Al-Naffouri and A. H. Sayed, “Transient analysis of data-normalized adaptive filters,” *IEEE Transactions on Signal Processing*, vol. 51, no. 3, pp. 639–652, 2003.

[13] J. Li, K. Cheng, S. Wang et al., “Feature selection: a data perspective,” *ACM Computing Surveys*, vol. 50, no. 6, pp. 1–45, 2017.

[14] R. Sandhu and F. Chen, “The multilevel relational (MLR) data model,” *ACM Transactions on Information and System Security*, vol. 1, no. 1, pp. 93–132, 1998.

[15] W. Chen, M. Hasanipanah, H. Nikafshan Rad, D. Jahed Armaghani, and M. M. Tahir, “A new design of evolutionary hybrid optimization of SVR model in predicting the blast-induced ground vibration,” *Engineering with Computers*, vol. 37, no. 2, pp. 1455–1471, 2021.

[16] J. X. Chen, D. M. Jiang, and Y. N. Zhang, “A hierarchical bidirectional GRU model with attention for EEG-based emotion classification,” *IEEE Access*, vol. 7, pp. 118530–118540, 2019.

[17] R. Dey and F. M. Salem, “Gate-variants of gated recurrent unit (GRU) neural networks,” in *Proceedings of the 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 1597–1600, IEEE, Boston, MA, USA, August 2017.

[18] S. Yu, D. Liu, W. Zhu, Y. Zhang, and S. Zhao, “Attention-based LSTM, GRU and CNN for short text classification,” *Journal of Intelligent and Fuzzy Systems*, vol. 39, no. 1, pp. 333–340, 2020.

[19] A. H. Abdulnabi, G. Wang, J. Lu, and K. Jia, “Multi-task CNN model for attribute prediction,” *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 1949–1959, 2015.

[20] M. Sajjad, Z. A. Khan, A. Ullah et al., “A novel CNN-GRU-based hybrid approach for short-term residential load forecasting,” *IEEE Access*, vol. 8, pp. 143759–143768, 2020.

[21] N. Dua, S. N. Singh, and V. B. Semwal, “Multi-input CNN-GRU based human activity recognition using wearable sensors,” *Computing*, vol. 103, pp. 1461–1478, 2021.