Journal of Organizational and End User Computing
Volume 36 • Issue 1

Analyzing E-Commerce Market Data Using Deep Learning Techniques to Predict Industry Trends

Wei Qian, School of Economics and Management, Harbin University, China
Yijie Wang, School of Digital Commerce, Zhejiang Institute of Mechanical and Electrical Engineering, China*

ABSTRACT

Faced with challenges in sales predicting research, this article combines the capabilities of deep learning algorithms in handling complex tasks and unstructured data. Through analyzing consumer behavior, it selects factors influencing sales, including images, prices and discounts, and historical sales, as input variables for the model. Three different types of neural network models-fully connected neural networks, convolutional neural networks, and recurrent neural networks-are employed to process structured data, image data, and sales sequence data, respectively. This forms a deep neural network for feature representation. Subsequently, based on the outputs of these three types of deep neural networks, a fully connected neural network is employed to train the sales prediction model. Ultimately, experimental results demonstrate that the proposed sales prediction method outperforms exponential regression and shallow neural networks in terms of accuracy.

KEYWORDS

Bi-LSTM, CNN, Fully connected neural networks, Sales predicting

INTRODUCTION

With the rapid expansion of the e-commerce market, the rise of online retail has become an indispensable component of corporate sales, especially for some e-commerce retail enterprises. Online retail sales not only occupy a significant position but even constitute the entire sales revenue for these enterprises. As businesses increasingly rely on e-commerce platforms for sales, accurately predicting and assessing market demand has become a key factor in formulating appropriate production plans. In this context, the use of deep-learning technology to analyze e-commerce market data, particularly forecasting online sales trends, becomes crucial.

Historically, research has focused primarily on methods for forecasting product sales by analyzing historical sales data using time-series methods to identify changes in sales trends (Hong, 2021; Zhang et al., 2022). However, traditional time-series methods impose high requirements on the stationarity of data, and consumer behavior in e-commerce markets typically exhibits randomness and disorder. This makes time-series methods perform poorly when facing certain factors that influence user

DOI: 10.4018/JOEUC.342093 *Corresponding Author

purchasing decisions, such as promotional strategies. In practical situations, the purchase-decision behaviors of individual users eventually converge into overall sales. However, due to the difficulty in acquiring factors related to individual purchase decisions in traditional sales-forecasting methods, these factors are rarely used for sales prediction. In the current internet environment, user browsing and purchasing behaviors are comprehensively recorded, providing businesses with the opportunity to utilize this data to identify factors influencing user purchasing decisions. Subsequently, deep-learning technology can be applied for product-sales forecasting.

In the e-commerce environment, the presentation of multimedia information is a crucial feature of transactions, especially in virtual environments where users find it challenging to directly observe and interact with actual products (Angeli et al., 2018; Gao et al., 2023b). User perception of products relies heavily on multimedia information presented on websites, including text, images, audio, and video. Among these, images, as the most intuitive and widely adopted form of information display, significantly influence user purchasing behavior through factors such as color and emotional appeal, making them a key determinant in user purchase decisions. Past studies have demonstrated that factors such as brightness, color, and emotions in product images can impact user purchase decisions. While research on utilizing image information for product-sales forecasting is currently limited, the rapid development of deep learning offers new methods for effectively handling image information. This development also provides technical support for integrating static and dynamic factors in product-sales forecasting.

Building upon the aforementioned analysis, this study focuses on leveraging deep-learning techniques to conduct in-depth analysis of e-commerce market data and predict industry development trends. The key contributions of this paper are outlined as follows:

(1) Diverging from traditional sales-forecasting methods, this study employs three different types of neural networks—convolutional neural network (CNN), bidirectional long short-term memory (Bi-LSTM), and Fourier CNN (FCNN)—to abstract and represent market features at multiple levels. By combining dynamic and static features, this paper establishes an innovative sales-prediction model, offering a more accurate and comprehensive predictive tool for the industry.
(2) This study maximizes the advantages of deep-learning methods in extracting image features and possessing strong learning capabilities. By applying these methods to sales forecasting, we achieve an end-to-end solution from raw data to final sales predictions. This approach facilitates the resolution of demand-forecasting issues in various industries, advancing research on industry development trends.
(3) The proposed sales-forecasting method in this study relies entirely on learning from historical data, eliminating the impact of subjective judgments on predictions. Since the model design is based solely on product and sales data and excludes errors that may arise from managerial subjective factors, the predictions are more objective and trustworthy. This method allows for periodic sales forecasting, providing businesses with more reliable decision support.

BACKGROUND

Time Series–Based Prediction Method

The most common and classic methods in demand forecasting start from the changes in demand itself. Based on historical sales data, the variation patterns are explored to predict the time series of sales for a future period. Due to the multitude of factors influencing sales, scholars believe that identifying the factors affecting sales is very challenging. Therefore, market demand is considered as a random process, focusing only on the time and quantity of market demand (Chern et al., 2023).

Exponential-smoothing forecasting methods, as a crucial component of time series–related techniques, have garnered attention from numerous scholars (Hsieh, 2019). The exponential-smoothing forecasting method assigns different weights to historical sales data, giving higher weights to data from neighboring forecast periods and lower weights to more distant data. Essentially, it is a moving average method with unequal weights. Gardner (1985) pointed out that exponential-smoothing forecasting has high accuracy in fitting mid-term to short-term data. This method effectively utilizes data from each historical period, enabling it to overcome errors introduced by random disturbances. Furthermore, as it assigns higher weights to recent data, the forecasting results can better reflect recent changes, considering the timeliness of time-series data (Gardner, 1985).

The multivariate regression method is a common approach in time-series analysis. Scholars in the research selected economic indicators such as gross domestic product, the first industrial output, and the second industrial output as variables. Combining the multivariate regression method has yielded good results in logistics-forecasting problems. The Grey Forecasting Model GM(1,1) is a prevalent forecasting method in predicting sales of fashion products. Xie et al. (2005) pointed out that the Grey Forecasting Model can still perform relatively well when the data are insufficient. For fashion products, given their short sales cycles, the Grey Forecasting Model tends to outperform exponential-smoothing methods in such cases. The mentioned methods are all forecasting models within the framework of time-series methods. While exponential-smoothing methods handle random disturbances effectively, they may struggle with sales changes due to price fluctuations. Multivariate regression methods use variables influencing the prediction target for forecasting, but achieving a good predictive model requires a longer time series for fitting. GM(1,1), while reducing the demand for historical data, is not adept at handling nonlinear predictive models.

Machine Learning–Based Prediction Methods

Another important category of demand-forecasting methods is machine learning–based sales-prediction methods. Scholars studied the Bass model under the influence of two factors, service and purchase intention, and constructed a multifactor impact product-sales prediction model using support vector machines (SVM) (Lee et al., 2014). As support vector machine regression is influenced only by samples closest to the support vector, it can reduce the demand for data. In situations with severely limited data, SVM tend to exhibit significant overfitting. In comparison, neural networks can better control the degree of model fitting (Nassibi et al., 2023).

The aforementioned studies that incorporated consumer-influencing factors did not utilize neural networks for sales prediction. Thomassey & Happiette (2007) addressed the issue of clothing sales in the context of the French clothing industry. The authors proposed a method using neural networks for clustering and prediction. Due to the absence of sales data for currently available products in the experiment, the authors employed a self-organizing map neural network for clustering historical sales data to obtain the sales-behavior characteristics of products. Probability neural networks were then used for prediction, resulting in favorable forecasting outcomes. The research demonstrated that neural networks can learn and summarize patterns of sales variations, enabling the utilization of these patterns for predicting new data. Brzęczek (2016) and Sano et al. (2014) utilized the autoencoding function of neural networks to encode and reduce the dimensions of categorical variables such as shipment locations, effectively lowering the dimensionality of high-dimensional data and uncovering correlations among different categories. Jian et al. (2020) and Tian and Wang (2022) indicated that the accuracy of phased sales predictions exceeded that of direct predictions using neural networks.

The predictive performance of machine learning–based forecasting methods depends on the variables used during prediction. If all the variables influencing sales cannot be identified, it may lead to a decrease in the effectiveness of predictions. Scholars have augmented their research by incorporating factors such as price and season that impact user purchasing behavior but have neglected the features of sales sequences. Consequently, many researchers have attempted to combine machine learning with time-series methods for prediction.

Ensemble-Prediction Model-Based Predicting Method

Some scholars simultaneously utilize time-series and machine-learning methods for sales forecasting, aiming to achieve more accurate predictions. Based on the combination forecasting concept proposed by Bates and Granger (1969), certain scholars have achieved strong fitting capabilities for simulated data through nonlinear combinations of different Bayesian models. However, this forecasting method lacks empirical evidence with real data and comparisons with other forecasting methods. After comparison with non-combination forecasting models, it has been demonstrated that combination forecasting models exhibit higher accuracy. Scholars (Vavliakis et al., 2021) argue that different forecasting models provide varied useful information and that prediction accuracy and emphasis often differ. Simply choosing one forecasting method or discarding methods with larger prediction errors may lead to the loss of useful information and result in resource waste. In practical forecasting, a more scientific approach has been discovered: combining different forecasting methods with certain structures and parameters for prediction, known as combination forecasting methods. Research (Fallah Tehrani & Ahrens, 2018) has shown that using logistic regression results with different parameters for sales forecasting, increasing the number of models, and altering the emphasis of prediction models can effectively enhance the accuracy of combination forecasting models. However, training multiple models often requires more training time. In contrast, using a single-layer dropout neural network can achieve the same results more quickly. However, the research on the combination forecasting models mentioned above used only historical sales sequence data.

The aforementioned studies primarily involve structured data, while unstructured data often contains more information. For instance, when studying the impact of online reviews on sales, relying solely on the displayed positive review count on a website may not accurately represent consumer satisfaction. Consumers often point out dissatisfaction in their comments but may still choose to give a positive rating. Therefore, there is a need to directly study unstructured data. Despite the strong learning capabilities of machine-learning methods, their dependence on data quantity and practical complexity still poses unavoidable challenges. Scholars (Aras et al., 2017) have investigated several common methods in demand forecasting, summarizing the strengths and weaknesses of various forecasting approaches. These methods are mainly divided into two categories: traditional statistical-based forecasting methods and machine-learning forecasting methods. In short-term forecasting, autoregressive models exhibit stronger adaptability and better prediction performance for nonstationary data. However, a common issue with time-series methods is their focus solely on fitting existing data, disregarding the influencing factors on demand. Machine learning–based forecasting methods, on the other hand, consider various factors in predicting sales, resulting in better fitting compared to traditional statistical forecasting methods. However, machine learning–based forecasting methods are more complex to operate, and the models have a time-sensitive nature, requiring timely updates. With the advancement of forecasting tools, machine learning–based forecasting methods are becoming increasingly user-friendly and feasible. To further improve the accuracy of demand forecasting, Giri & Chen (2002) proposed combining two or more forecasting models for ensemble forecasting.

In the aforementioned sales-forecasting research methods, traditional time series–based sales-forecasting methods have a relatively simple calculation process, but their predictive performance and applicability are limited. Many scholars (Hsu et al., 2020; Wang & Chen, 2022) have turned to machine-learning methods to study sales forecasting, enhancing the fitting capability of forecasting models. Ensemble forecasting models further improve predictive outcomes. Several studies by scholars indicate that there are factors beyond sales volume itself that can be used for sales forecasting. Only a few scholars incorporate factors that may affect sales as independent variables in sales forecasting. These factors are often structured data, unstructured data are rarely used. Particularly, the vast amount of online marketing information in the form of images is significantly underutilized (Mosteller et al., 2014). In addition to considering the variation in historical sales, this study utilizes other marketing information influencing sales as independent variables for sales forecasting. Specifically, unstructured data such as images, which numerous studies have demonstrated to be the most crucial online marketing information, have a profound impact on user purchasing-decision behavior.

METHOD

Construction of Neural Network for Sales Prediction

Different types of data require selecting different models for processing based on their characteristics. Structured data, where there is no inherent order between features and each feature represents a different meaning (for example, product category and product price) require only traditional data cleaning, such as handling missing values through methods like singular value decomposition, to be used for prediction. Sequential data represents ordered data of certain features changing over time; the variation of sequential data itself is crucial for prediction. For example, in time-series methods, when predicting sequential data, it is necessary to identify its trends and periodicities, as they are inherent characteristics of sequential data changes. Image data is represented as a three-dimensional tensor, depicting the length, width, and color-channel information of the image. Each value in this tensor represents the color depth at its specific position and channel; the size of each value and its relative position contain all the information about the image.

In other words, structured data, sales sequence data, and image data inherently exist in different feature spaces, making direct computations between them impractical. By employing fully connected, Bi-LSTM, and convolutional neural networks to respectively map these three types of data and aiming to minimize the loss function values, this study's ultimate goal is to map data from three distinct feature spaces into a common feature space.

To systematically address these three distinct types of data, this study integrates three different neural network models for feature representation. Through the combination of neural networks with different characteristics in parallel, an asymmetrical neural network structure is formed, allowing targeted utilization of networks with different features to implement a sales-forecasting neural network model.

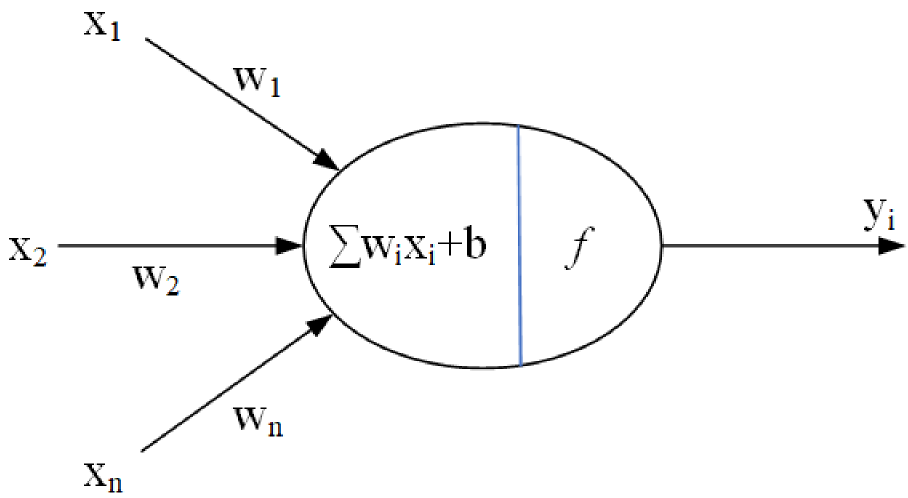
Structured Data Feature Representation of Products

Structured data of products refers to static numerical data such as the price, discount, and launch time of the products. The relationship between the structured information of products and the goal of sales forecasting is often not a simple linear one. Therefore, it is necessary to extract features from such numerical data for feature representation. In a fully connected neural network, each neuron contains a nonlinear function that can map the original features nonlinearly. After appropriate training, it can construct a nonlinear model (Gao et al., 2023a; Tian et al., 2023). Additionally, fully connected neural networks are typically limited by computational resources and are often of a smaller scale, making them suitable for extracting features from features that are easily represented. For features like price and discount, which have clear meanings themselves, feature representation is relatively straightforward. Therefore, this study uses a fully connected neural network for the feature representation of structured data related to products.

The fully connected network is a stacked network structure, where each node in the network is an artificial neuron and each artificial neuron is composed of input layer weights, neuron bias, and an activation function. The structure of an artificial neuron is illustrated in Figure 1.

The input of a single neural node can be from the input layer or the output of the previous layer in the neural network; the input to a neuron can consist of multiple values, so its input can be seen as a vector $\mathbf{X}$. Each input to the neuron corresponds to an input weight, making the neuron's input also a vector $\mathbf{W}$. As there are some variables that each neuron may not have considered, a bias term b is added. After summing the weighted inputs, a nonlinear activation function is applied to enable

Figure 1. Neuron node diagram



the neural network to represent nonlinear functions. The activation function can be any nonlinear function, and the commonly used activation functions are as shown in 1–4:

$$\text{sigmoid}\left(x\right) = \frac{1}{1 + e^{-x}} \tag{1}$$

$$\text{Tanh}\left(x\right) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \tag{2}$$

$$\text{ReLU}\left(x\right) = \max\left(0, x\right) \tag{3}$$

$$\text{Leaky ReLU}\left(x\right) = \max\left(\alpha x, x\right) \tag{4}$$

The above activation functions all exhibit nonlinear characteristics and can be used as activation functions in neural networks. Among them, the sigmoid function is the most widely used. According to the fully connected neural network structure shown in Figure 2, the output of any neuron in a layer can be considered as the input for the next layer. This paper utilizes the gradient descent method for training the neural network. According to the chain rule, the gradient of the current node needs to be multiplied by the gradient of the next layer. As the number of layers in the network increases, the gradient near the hidden layers close to the input layer may approach 0 (gradient vanishing) or tend to infinity (gradient exploding). However, both sigmoid function and tanh function encounter the problem of gradient vanishing during training, leading to slow learning in neural networks. ReLU (rectified linear unit) function, based on its functional properties, can avoid the issues of gradient vanishing and exploding. ReLU is a simple yet effective nonlinear activation function. It sets all negative input values to zero and keeps positive values unchanged. One of the main advantages of ReLU is that it does not introduce excessive computational complexity during the model-training process and helps alleviate the gradient-vanishing problem. The simplicity of ReLU makes it widely used in the field of deep learning, as it can effectively learn complex feature mappings. Compared to other activation functions such as sigmoid or tanh, ReLU's computations are lighter and it does not suffer from gradient-saturation issues, making the model easier to train. Leaky ReLU function can also address the issue of gradient vanishing. However, ReLU function is computationally simpler and more lightweight compared to Leaky ReLU function.

Therefore, this study employs a fully connected neural network with ReLU as the activation function to represent structured data. The hierarchical structure of the neural network and the nonlinear activation function enable effective nonlinear mapping of structured information. The outcome of this mapping is the feature representation of structured data. However, training large neural networks not only requires a significant number of computational resources but also necessitates a sufficiently large amount of training data. This makes it challenging to apply fully connected neural networks to image data.

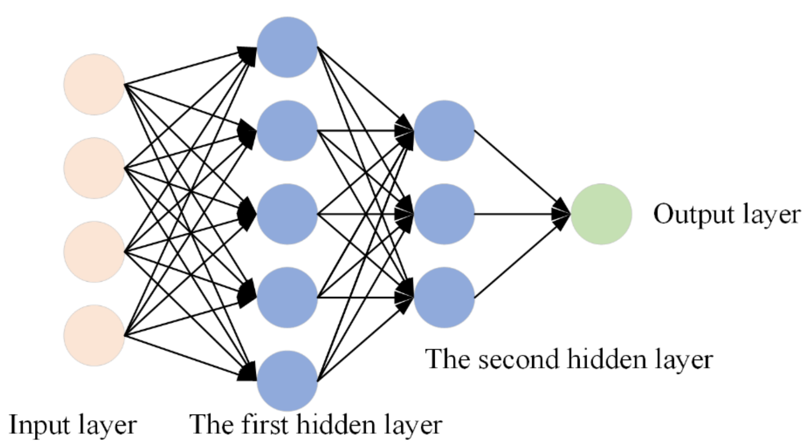
Feature Representation of Product-Image Data

The image data of products are themselves a high-dimensional dataset, and such high-dimensional data cannot be directly used for sales forecasting. Therefore, this study utilizes CNNs to extract features from product images. In image data, the relative positions of object contours, textures, and color features are often crucial. Using a fully connected neural network may not effectively capture features with relative positions. For example, the features of a bird include the beak and eyes, and the presence or absence of the beak and eyes themselves is an important feature. Additionally, the relative positions of the beak and eyes determine whether a bird is present in the image. In a batch of bird images, the positions of birds are typically not fixed, making relative positional features important for image recognition. In the case of product images, due to the presence of various types of products within the same category, manually defining all features is challenging. CNNs, by observing the sliding window on the image, extract features. The convolutional kernels inherently capture the relative positional features of the images. Moreover, with random initialization and the training process, the neural network can autonomously define the features it extracts. As long as the network size is sufficient, it can comprehensively capture important features. Therefore, CNNs can effectively extract features from the image data in this study.

In the structure diagram of a fully connected neural network, we can see that each input is individually connected to a node in the next layer using a separate weight. Assuming a colored image with width W and height H as input, constructing a fully connected neural network with n nodes would result in a large number of weights, specifically $W \times H \times 3 \times n$. Such a large-scale neural network is not practical. Moreover, past research has found that the more parameters a neural network has, the more susceptible it is to overfitting. Therefore, for image data, this study employs CNNs for processing.

This study constructs multiple convolutional filters, where each filter is a three-dimensional weight tensor with width w, height h, and depth d, along with an activation function. Here, $w \leq W$,

Figure 2. Fully connected neural network structure



$h \leq H$, and d is the third dimension of the input tensor (e.g., 3 for colored images). The convolutional layer with n filters has a parameter count of $w \times h \times d \times n$, which is significantly smaller than a fully connected neural network. The convolutional filters slide and move across the image, producing an activation value with each slide. Finally, they output a three-dimensional tensor, as illustrated in Figure 3.

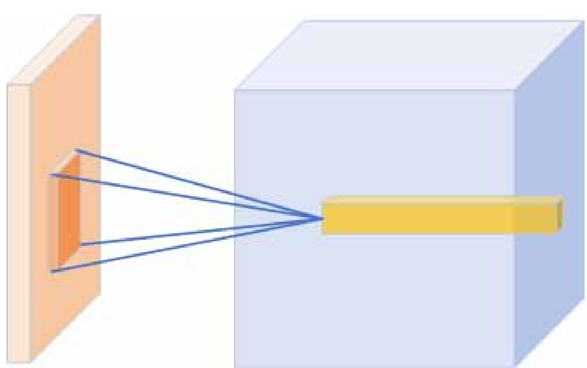
As the connection weights of neural networks are randomly initialized, during the training process of backpropagation, different convolutional filters become sensitive to different local features. This enables convolutional neural networks to automatically extract useful features in image classification. Although convolutional neural networks can effectively handle image data, they require uniform dimensions for all input data. For time-series data, where the quantity of data varies, it is challenging to process inputs of variable lengths effectively. Moreover, it fails to address the issue of long-term dependencies in sequence data.

Feature Representation of Sales Sequence Data

The features of sales sequence data are challenging to extract; the early changes in sales can provide a reference basis for later sales predictions to some extent. In sales sequence data, the sales at each moment are important features. The order of the same numerical values represents different meanings; the changes in sales between multiple moments are also crucial features due to the ordered nature of sales sequence data. This implies the presence of a long-term dependency issue in sales-forecasting problems (Ji & Zhang, 2023; Zhang & Shankar, 2023). As a type of recurrent neural network, the Bi-LSTM neural network introduces two hidden states for each time step on the basis of long short-term memory (LSTM). Each hidden state in the Bi-LSTM model aligns with either the forward or the backward LSTM, enabling the network to assimilate contextual information from both temporal directions. This architectural choice enhances the model's ability to comprehend intricate dependencies within the sequence. Consequently, Bi-LSTM often excels in tasks demanding a nuanced understanding of contextual information. In the context of this study, a Bi-LSTM neural network is employed to adeptly capture and represent the intricate features embedded in sales sequence data.

The traditional statistical learning methods and three-layer back-propagation neural networks face a fundamental problem: they are unable to capture long-term dependencies in data (**Merzbacher et al., 2003;** Bengio et al., 1994; Ungureanu & Cziker, 2021). To address the challenge of long-term dependencies and comprehensively understanding the relationships within sequences, this study utilizes Bi-LSTM neural networks for feature representation in sales sequence data. The issue of long-term dependencies refers to the difficulty in learning the impact of distant past data on future predictions. For instance, in sales forecasting, the initial order by a company may significantly underestimate market demand, leading to stockouts and a period of zero sales in the historical data. In

Figure 3. Convolution operation



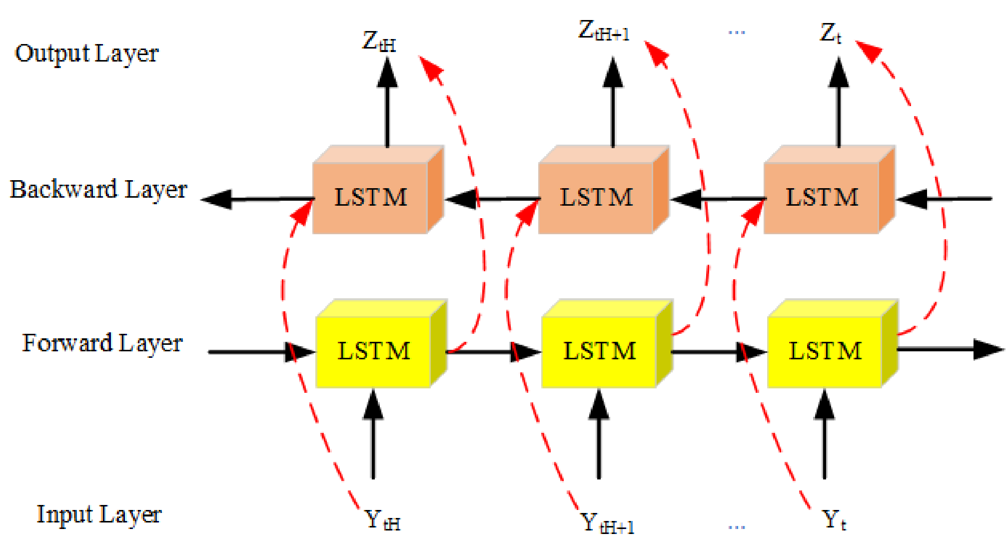
such cases, meaningful guidance for future sales predictions should be derived from the initial sales and sales variations when the product was first introduced, rather than recent sales. Both time-series methods and traditional statistical learning methods find it challenging to handle such prediction problems. However, the Bi-LSTM in recurrent neural networks can effectively learn from such data (Zhang et al., 2022). (Zheng et al., 2023)

The Bi-LSTM network employed in this study is capable of learning bidirectional dependencies between sequence data and time steps without the need for increased data volume. Specifically, the network consists of the following layers: input layer, forward LSTM layer, backward LSTM layer, hidden layer, and output layer. In the input layer, sequence data are received as input, potentially containing information about time steps and other relevant features. The forward LSTM layer comprises multiple LSTM units, each equipped with three gates—input gate, forget gate, and output gate. Each unit controls the flow of information, selectively remembering or forgetting prior information. Additionally, each LSTM unit maintains a cell state responsible for preserving long-term information. The backward LSTM layer is analogous to the forward LSTM layer but propagates information in the reverse direction. The hidden layer combines the outputs of the forward and backward LSTM layers through concatenation or other methods, forming the final hidden representation, enabling the network to comprehensively consider bidirectional information at each time step. The output layer produces a sequence containing predicted values for each time step. In the context of sales forecasting, each time step's predicted value represents the anticipated sales quantity at a future moment. The specific architecture of the Bi-LSTM neural network is illustrated in Figure 4.

Sales-Forecasting Neural Network Structure

For the structured data of products, such as price, discounts, and release time, FCNN can capture their complex feature relationships through nonlinear mappings. In sales forecasting, sales data are typically in the form of time-series data, where Bi-LSTM can effectively learn and represent temporal dependencies, including sales trends and periodicity. Additionally, product image data may contain visual information relevant to sales. Through CNN, features in the images, such as color and shape, can be effectively captured, providing more comprehensive information for sales predictions. By combining these three models, features of products can be captured and represented in different aspects, thereby enhancing the overall performance of the sales-forecasting model. 窗体顶端

Figure 4. Bi-LSTM structure diagram



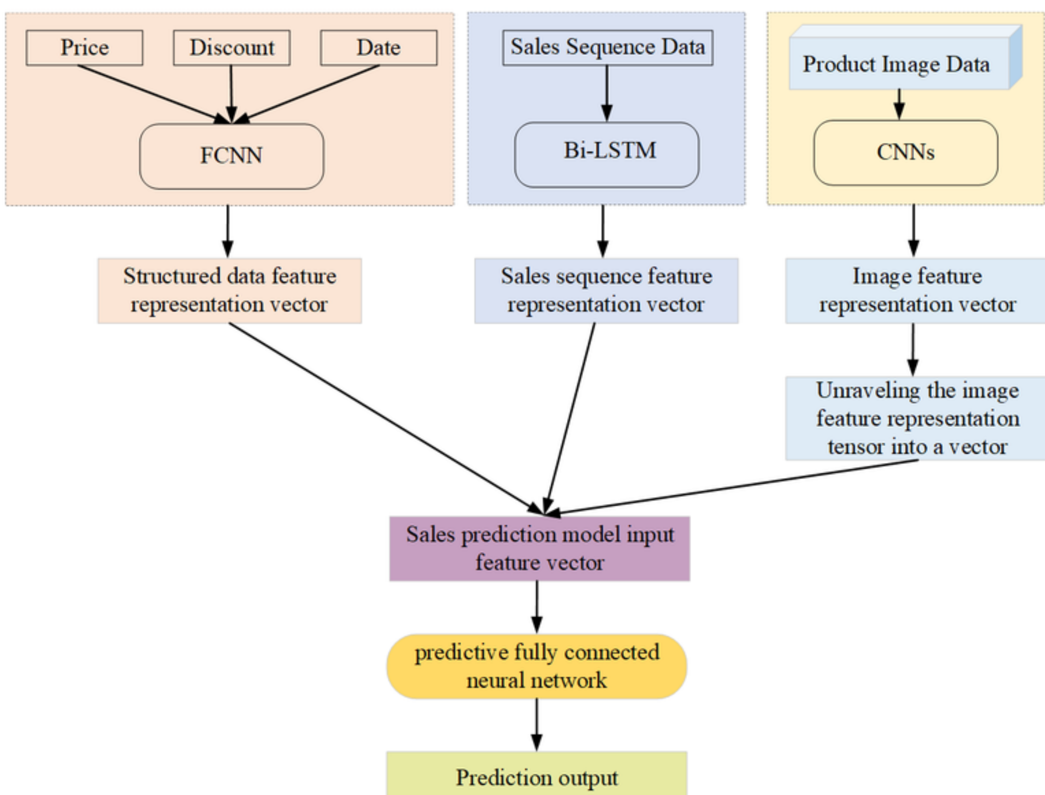
After the three aforementioned feature representations are obtained, the product's structured data, image data, and sales-sequence data are each transformed into vectors of varying lengths through neural networks. These three vectors are then concatenated to form a unified input vector for the sales-forecasting neural network. Since all three types of data are captured as static features, the vector representing the features at a given moment remains a fixed-length vector when predicting sales at a specific time point. Therefore, a fully connected neural network, adept at handling structured numerical data, is used as the sales-forecasting neural network. This step is also the realization of integrating static and dynamic features.

The neural network structure proposed in this paper is an asymmetric neural network, as shown in Figure 5. As it is composed of three different types of neural network architecture models, the predictive model presented in this paper can utilize various types of data and make more comprehensive use of data from different sources within the enterprise.

Neural Network Training Process

In this study, ReLU is used as the activation function. According to the chain rule, in parts with gradients, the gradient is 1, and with the cumulative multiplication of local gradients, the gradient will not tend toward 0 or infinity. The initial position may lead the neural network into a local optimum, making it difficult to train. Therefore, this study adopts some methods to avoid the issue of local optimum caused by the initial position. In addition, neural networks have strong learning and memorization capabilities, resulting in significant overfitting problems. Neural networks can even predict one set of random numbers using another set of random numbers, making it easy to perform

Figure 5. Neural network structure diagram for sales-forecasting model



well on training data but poorly on testing data. Hence, some measures need to be taken to address the overfitting problem.

Stochastic Batch Gradient Descent

Many machine-learning training processes are based on the batch gradient descent method, where all training data are input into the model for training during each iteration. For the overall data, the existence of saddle points is possible, and the training of the model may get stuck in a saddle point, leading to a local-optimum problem. An effective method is to randomly extract a batch of data for training in each iteration instead of using the entire dataset. Although this training process is unlikely to directly descend along the direction of the steepest gradient, it increases the likelihood of escaping saddle points, or local optima. Moreover, for neural networks, with a large number of weights and computationally intensive steps, optimizing using the entire dataset in each iteration incurs significant time costs. Therefore, using the stochastic batch gradient descent method can reduce training overhead.

Batch Normalization

Due to the activation function altering the data distribution, deep neural networks impose high demands on the weights of neuron connections during training. Consequently, only a relatively small learning rate can be set during training to prevent fitting issues, resulting in slow learning rates for deep neural networks. To address this problem, adding a batch normalization (BN) layer before each layer allows the use of a relatively higher learning rate and more lenient weight initialization requirements. As shown in Figure 6, based on test results on the MNIST dataset, using BN in neural networks demonstrates faster learning efficiency compared to networks without BN, and it stabilizes the inputs for each layer. The MNIST dataset was created by the National Institute of Standards and Technology (NIST) in the United States and is widely used for testing new classification algorithms and models.

Early Stopping

Early stopping is a common training method to prevent overfitting. During the learning process, neural networks may prioritize learning more general patterns under the influence of other measures to prevent overfitting. However, as learning progresses, neural networks may still exhibit overfitting to the training data, as shown in Figure 7. To avoid overfitting to the training data, the dataset is first divided into training and testing data and the loss values for both are observed. Training is stopped when the loss value for the testing data no longer decreases. However, since the optimization process of stochastic batch gradient descent does not necessarily always optimize along the global optimum direction, it is necessary to record the optimal loss value for the testing data. Training is stopped when there is no improvement in the loss value for n consecutive training iterations and the loss value for the training data continues to decrease.

EXPERIMENTAL DESIGN AND RESULTS ANALYSIS

Experimental Data

In predicting the trends of the fashion industry, this study employed data representative of three distinct domains within the sector, namely down jackets, sportswear, and casual fashion. These datasets were used as inputs for the model, and trend predictions were generated for each of the three representative domains. The comprehensive forecast for the industry trend is derived from the average of trend predictions associated with these three representative clothing categories. This approach seeks to provide a holistic industry trend forecast by considering representative areas and aims to offer valuable insights into the overall trajectory of the fashion industry.

Figure 6. Neural network input distribution with or without BN Processing (a) Comparison of training speed (b) Hidden layer input without BN (c) Hidden layer input with BN

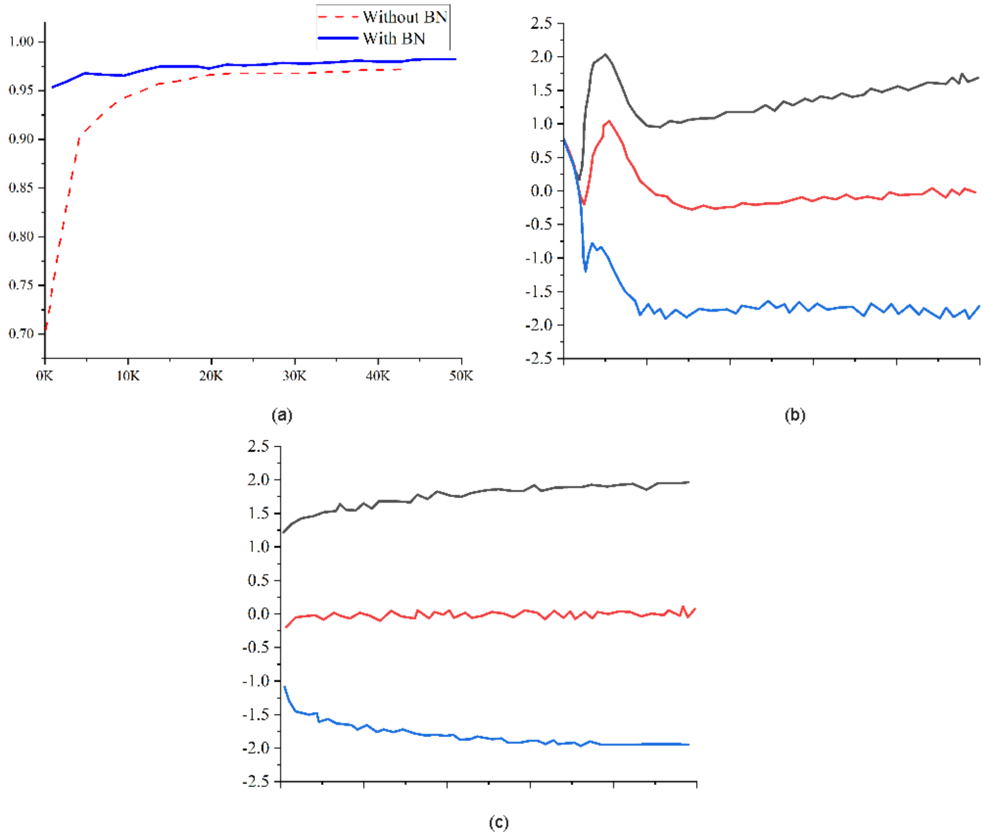
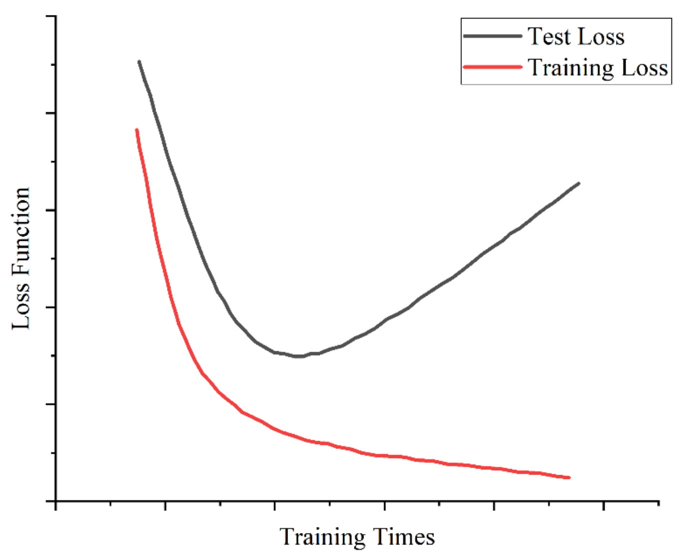


Figure 7. The relationship between training error, test error, and training times



This study gathered a total of 9,286,419 down jacket order records, 8,187,412 sportswear order records, and 10,187,417 casual fashion order records from the JD.com platform spanning from September 2013 to December 2017. Additionally, a collection of 4,352 images for down jackets, 6,235 images for sportswear, and 8,236 images for casual fashion, all available for sale during this period, were also compiled.

Experimental Design

Data Preprocessing

From the data description, it can be inferred that the dataset contains unreasonable dirty data, such as product prices being 0. Typically, products with a price of 0 in these orders are considered as complimentary items. Additionally, there are instances where product pricing is lower or higher than expected, attributed to system testing errors. Since the correct values for the mentioned dirty data cannot be traced, and they constitute a small proportion of the overall dataset, it was decided to remove this portion of the data.

The sales cycle of the product is approximately three months, indicating a relatively short sales duration. There is significant variability in the discount rates applied to the product, and a portion of users did not receive any discounts. Therefore, this study applies a logarithmic transformation to the sales volume and discounts of the product to mitigate the impact of extreme values on the model. Deep neural networks have high demands on input data, and input distributions close to normal distributions can prevent divergence during the neural network training process. Moreover, a distribution closer to normality enhances the learning rate, thereby accelerating the model-training speed. Consequently, standardization is applied to all features. The standardization formula is as follows:

$$\text{standardization}\left(x\right) = \frac{x - x_{min}}{X_{max} - X_{min}} \tag{5}$$

CNNs for processing image data require input images to have the same resolution. In the original dataset, there are instances where the resolution of certain images differs, necessitating the resizing of the original images. In this study, all images with a resolution less than 200 x 200 pixels underwent bicubic interpolation, while images with a resolution exceeding 200 x 200 pixels underwent down sampling. This ensures that all images in the training data are standardized to a resolution of 200 x 200 pixels.

In addition, categorical data cannot be directly computed. In this study, one-hot encoding is applied to encode categorical data, transforming categorical variables into numerical vectors to make them eligible for mathematical operations. For example, the season attribute includes four categories—spring, summer, autumn, and winter. Their encoding is represented as follows: spring [1, 0, 0, 0], summer [0, 1, 0, 0], autumn [0, 0, 1, 0], winter [0, 0, 0, 1].

Data Representation

This article's core approach to constructing a sales-prediction model using TensorFlow involves building a computational graph. The computational graph is a directed graph where each node represents a numerical computation and edges represent data. This study involves diverse data types with varying tensor dimensions. To address the issue of local optima, stochastic gradient descent is employed. Each neural network training iteration involves extracting a certain number of samples for training, where the batch size indicates the number of samples drawn in a single training iteration. Structured features, akin to relational tables in a database, represent a two-dimensional tensor, with each row representing a record. Therefore, structured data is a second-order tensor. Sales data serves as a feature input for the past three years, where only the sales figures themselves are considered as variables for each year. Consequently, sales data is treated as a third-order tensor. Image data encompasses the length, width, and RGB values of each pixel, resulting in a four-dimensional tensor. The input features of this model include structured features, historical sales, and image features, with dimensions of [batch size, 10], [batch size, 3, 1], and [batch size, 200, 200, 3], respectively. The dimensions of the input data are summarized in Table 1.

Representation of Structured Feature Data

In FCNN, each node in every hidden layer is connected to every node in the preceding layer close to the input layer. Suppose there are nx nodes in the upper layer, corresponding to nx inputs. Let $\mathbf{X}$ be the input vector, and $\mathbf{W}$ be the weight vector with nx weights. The output of neuron node j is calculated as:

$$\text{output} = \text{ReLU}\left(\sum_{i=1}^{nx} X_i W_i + b\right) \tag{6}$$

The collection of weight vectors for a hidden layer containing m nodes forms a second-order tensor of shape [nx , m]. The final output of the hidden layer is a matrix of shape [batch size, nx]. The input layer for structured data in this study consists of a total of 15 elements. Following the empirical formula for the number of nodes in a neural network hidden layer, we set the hidden layer to include eight nodes.

Representation of Sales Data Features

This study aggregates the collected data on a weekly basis, obtaining the sales data for each product for every week since its release. For the Bi-LSTM, the sales sequence for the three weeks preceding the prediction time is extracted and used as features. The input for the Bi-LSTM needs to include three dimensions: batch size, the number of time steps, and the number of variables at each time step. In this study, the number of time steps is set to 3, and the number of variables at each time step is set to 1. During the actual experimental process, reducing the number of time steps in the input would lower accuracy, while increasing the number of time steps does not necessarily improve accuracy. The number of neuron nodes is still set to 8 based on empirical formulas and experimental observations.

Representation of Image Feature Data

This study constructs a four-layer convolutional neural network. During training, the input tensors for each layer are either images or the output from the previous layer, both in the form of [batch size, image width, image height, 3] or [batch size, output tensor width, output tensor height, number of filters in the previous convolutional layer]. In the convolutional operations, padding is applied in this study. The dimensions of the image or input tensor are increased at the second and third axes to avoid having fewer activation values at the tensor edges compared to other positions. The convolutional kernel is shaped as per the common 5×5×(input tensor's fourth dimension) found in the literature. The stride for movement is set to 1, and the fourth dimension of the tensor output after convolution

Table 1. Feature–Dimension reference

Feature	Dimension
Structured Features (Price, Discount, Season, Category)	[batch size, 10]
Historical Sales	[batch size, 3, 1]
Images	[batch size, 200, 200, 3]

is equal to the number of kernels in the current layer. Additionally, max pooling is employed, with the pool size matching the convolutional kernel size.

The Fusion of Predictive Model Features

The three different types of neural networks described above correspond to receiving and processing three different types of features in the data. Leveraging the capability of neural networks to automatically extract features from data, each network is designed to extract features from different types of data. The three neural networks will produce tensors of different shapes. To fuse the data features extracted by the three types of neural networks, we introduce a fully connected neural network. This network combines and processes the outputs of the three networks to achieve sales prediction. In this study, all dimensions beyond the first are flattened. The outputs of all three networks are flattened into tensors of shape [batch size, N], and these flattened tensors are concatenated into a single tensor. This combined tensor is then input into the fully connected neural network for further processing.

Training of Predictive Models

Before training the model, 20% of the data were randomly selected as the test set, and the remaining data were used as the training set. The model was trained exclusively on the training set, while the test set was utilized to monitor the training process and validate the model's effectiveness. Reasonable initialization methods can help avoid the vanishing-gradient problem in neural networks using ReLU as the activation function but cannot address the gradient-explosion problem that arises when the network has many layers.

Gradient explosion causes the weights of deep neural networks to oscillate repeatedly, hindering convergence. To mitigate this issue, this study employed the gradient-clipping method. Gradient clipping involves limiting the gradients after computing them for each layer, preventing excessively large gradients. After multiple experiments, it was found that setting the maximum and minimum values of the gradients to [-0.01, 0.01] helps prevent gradient explosion while trying to maintain training speed. To enhance training speed and prevent some nodes' outputs from becoming too large or too small, potentially affecting the learning of the next layer, this study added BN layers between the two hidden layers. The BN layer is a method of transforming the data from the previous layer, ensuring that the data become a distribution with a mean of 0 and unit variance before input.

The designed neural networks in this study, including the fully connected neural network for structured data, the Bi-LSTM neural network for sales data, and the convolutional neural network for image data, all consist of a single hidden layer with relatively few neurons. The image data convolutional neural network, however, comprises four hidden layers and more neurons. Consequently, the convolutional neural network for image data requires more training cycles. During joint training, all three neural networks and the final prediction neural network are trained simultaneously. As a result, when the fully connected neural network for structured data and the Bi-LSTM neural network for sales sequences converge, the convolutional neural network for image data may not have trained sufficiently to extract effective features from the data. Therefore, as the training progresses, the learning process of the neural networks tends to minimize the activation of the convolutional neural network for image data. To address this issue, a staged and modular training approach is proposed in this study. In the initial phase of training, this study freezes the weights of the fully connected neural network for structured data and the Bi-LSTM neural network for sales sequences. Only the connection weights of the convolutional neural network for image data and the fully connected neural network for sales prediction are trained. To prevent overfitting, the study concurrently monitors the error on the test set. If there is no lower error observed on the test set in 50 consecutive training cycles, the early stop is applied to halt the training.

Results Analysis

Model Performance Analysis

This study plotted the line charts of training loss, test loss, and learning rate for clothing industry trend-prediction models, as shown in Fig. 8. The study uniformly selected values ranging from 0.0001 to 0.01 as learning rates for testing.

From Figure 8, it was observed that with the decrease in learning rate, the loss values showed a certain degree of decline. This indicates that the optimal learning rate for this model is below 0.01. When the learning rate is below 0.005, the model cannot effectively converge during training, and the loss values rapidly increase. This suggests that when the learning rate is below 0.005, the neural network's fitting speed becomes excessively slow. Additionally, it indicates that the prediction models for the clothing industry achieve relatively good results with a learning rate in the range of [0.005, 0.008]. Finally, based on the trend chart of learning rate to and loss values, this study determined the learning rate parameters for the prediction models in the clothing industry to be 0.007.

Additionally, this study employs the commonly used dropout technique in deep-learning models to address the issue of overfitting. The key parameter of dropout is the retention probability of neuron nodes, indicating the probability of whether a neuron node has an output in a particular training instance. A higher retention probability results in a more complete structure of the neural network for a single training, while a lower retention probability introduces more noise into the neural network, leading to greater differences in the features learned by neuron nodes. An excessively low retention probability may introduce excessive noise during the training process, causing slow convergence and poor fitting performance of the neural network. Conversely, an excessively high retention probability may diminish the regularization effect of dropout, failing to achieve the goal of resolving overfitting issues. Therefore, this study adopted a retention probability of 0.5 during the training of the model.

Model-Performance Verification

To further demonstrate the verification of the feature-representation effectiveness in each module of the three prediction models, this study modified the model structure and trained four models to

Figure 8. Learning rate and loss function

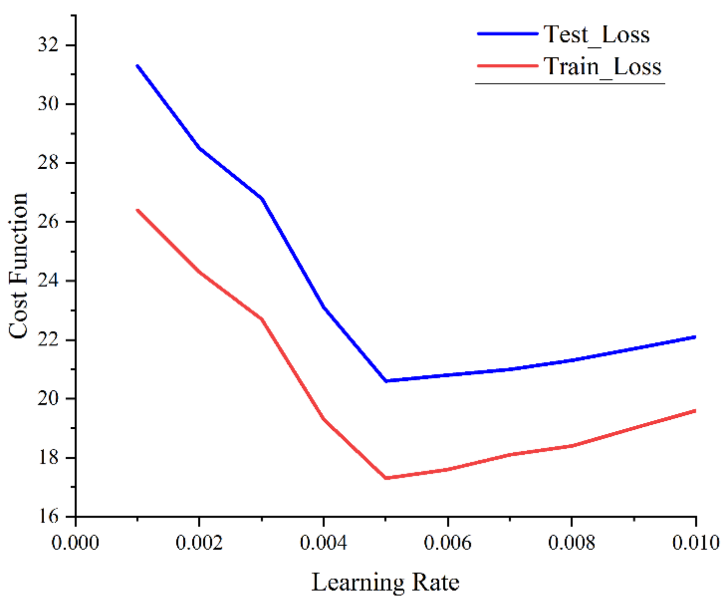
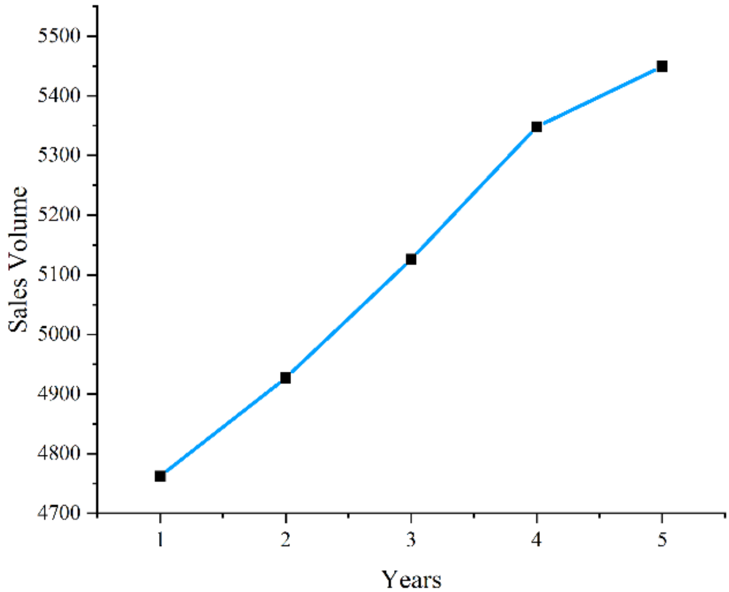


Figure 9. Dropout rate and loss function value



compare their predictive performance. The structure of all four predictive fully connected neural networks remained unchanged, but modifications were made to the neural network structure before the predictive fully connected neural network input. The first model is the sales-prediction model proposed in this paper. The second model removes structured data, the third model removes sales sequence data, and the fourth model removes image data.

In the predictive model for the clothing industry, from the predictive performance of the four different neural networks, it can be observed that removing the structured data in the sales-forecasting method significantly increases the mean and variance of prediction errors. This indicates that features such as the product's launch time, type, and price are crucial for predicting sales. The method without historical sales data leads to a significant increase in sales errors. The prediction accuracy of the method without image data is similar to the proposed sales-prediction model in this paper. However, the inclusion of product-image information can still improve the model's prediction accuracy to some extent, demonstrating that product images are indeed one of the influencing factors on sales. Although the feature representation of image data can enhance the accuracy of sales predictions, its impact on improving the model's prediction accuracy is limited. The specific results are shown in Table 2.

From Table 2, it can be observed that the omission of structured data for sales prediction markedly amplifies both the mean and the variance of prediction errors, underscoring the critical role of structured data in enhancing prediction accuracy. Furthermore, the exclusion of historical sales data

Table 2. Error statistics for different structural models in the clothing industry

	Complete Model	Lacking Structured Feature	Lacking Historical Sales Data Feature	Lacking Image Data Feature
Mean Prediction Error	40.03	47.53	82.45	43.24
Prediction Error Variance	30.31	105.26	435.70	35.62
Maximum Prediction Error	325.17	812.23	1,135.75	368.21

results in a significant surge in sales-prediction errors, emphasizing the indispensable contribution of historical sales information to the overall predictive capability of the model. Intriguingly, when image data are removed, the prediction accuracy remains comparable to the performance exhibited by the sales-prediction model proposed in this paper. However, it is crucial to note that the complete model, incorporating all three components of structured data, historical sales data, and image data, consistently demonstrates the most robust prediction performance. Therefore, this phenomenon suggests that the complete model performs best among the four different neural network models.

Analysis of Predictions from Different Models

To validate the effectiveness of the forecasting model proposed in this paper, this section predicts the sales of winter down jackets using actual data based on the forecasting model for the clothing industry. A comparison is made with the exponential-regression model and the shallow neural network model. When the sales-forecasting models are constructed using the other two methods, the exponential-regression prediction method cannot utilize structured data and image data and the shallow neural network cannot use image data, with sales history truncated to the first three periods as structured data.

The limitations of the exponential-regression model become evident, as it relies solely on historical sales data without incorporating information on discounts during the prediction process. Consequently, when confronted with fluctuations in sales patterns, the exponential-regression model experiences substantial delays in adapting to these changes. In contrast, both the shallow neural network and the deep-learning models exhibit comparable performance across various metrics, with the deep-learning model demonstrating a slightly superior ability to make accurate predictions compared to the shallow neural network. Upon closer examination of individual products, it becomes apparent that the shallow neural network tends to yield relatively large prediction errors, particularly in the early stages of forecasting for newly launched products. This discrepancy can be attributed to the fact that, during the initial phases of product launch, the sales history is incomplete and fails to adequately reflect the product's market performance. The shallow neural network, relying on factors such as launch time, category, and price, lacks the capacity to capture the visual features of the product. This deficiency hampers its accuracy in making predictions, particularly for products with limited historical sales data. Therefore, it is challenging to make accurate assessments of the product, and in the early stages of product launch, the deep-learning model demonstrates better predictive performance. The predicted results of the three models are shown in Figure 10.

In Figure 10, the vertical axis represents the sales volume in pieces, and the horizontal axis represents time in week. The sales are represented by the red line, while the blue line corresponds to the sales predictions of the exponential-regression model. Similarly, the brown line represents the sales predictions of the neural network model, and the green line represents those of the deep-learning model. From the observations in Figure 10, it can be inferred that the deep-learning model performs the best, demonstrating the most outstanding sales-forecasting effectiveness.

Clothing Industry Trend Predicting

The above experimental results clearly demonstrate the excellent performance of the proposed model for predicting the sales of down jackets. Based on this model, the paper also predicts the sales of sportswear and casual wear and extrapolates the trends in the clothing industry for the next five years, as shown in Figure 11.

According to Figure 11, it can be observed that the sales of sportswear are lower than the sales of casual wear, but overall, the sales of both types of clothing show an increasing trend. This trend can be ascribed primarily to recent economic growth, leading to heightened consumer spending, particularly on new clothing. Additionally, the influence of seasonal and fashion trends, along with the surge in e-commerce, where a growing number of consumers prefer online shopping, contributes significantly to the expansion of the apparel sector. Hence, this provides strong support for the validity of the model proposed in this paper.

Figure 10. Predicted results of the three models

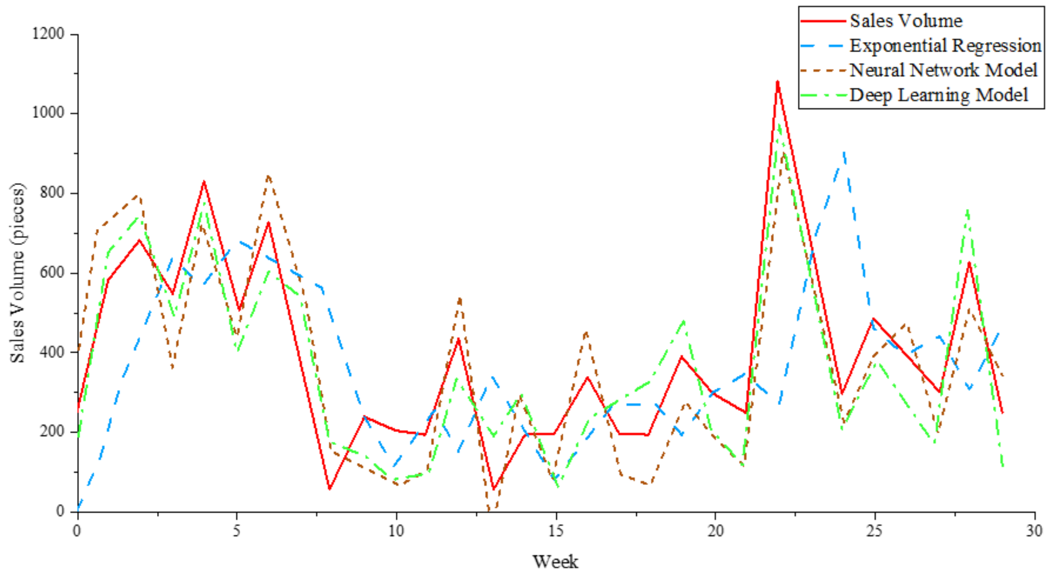
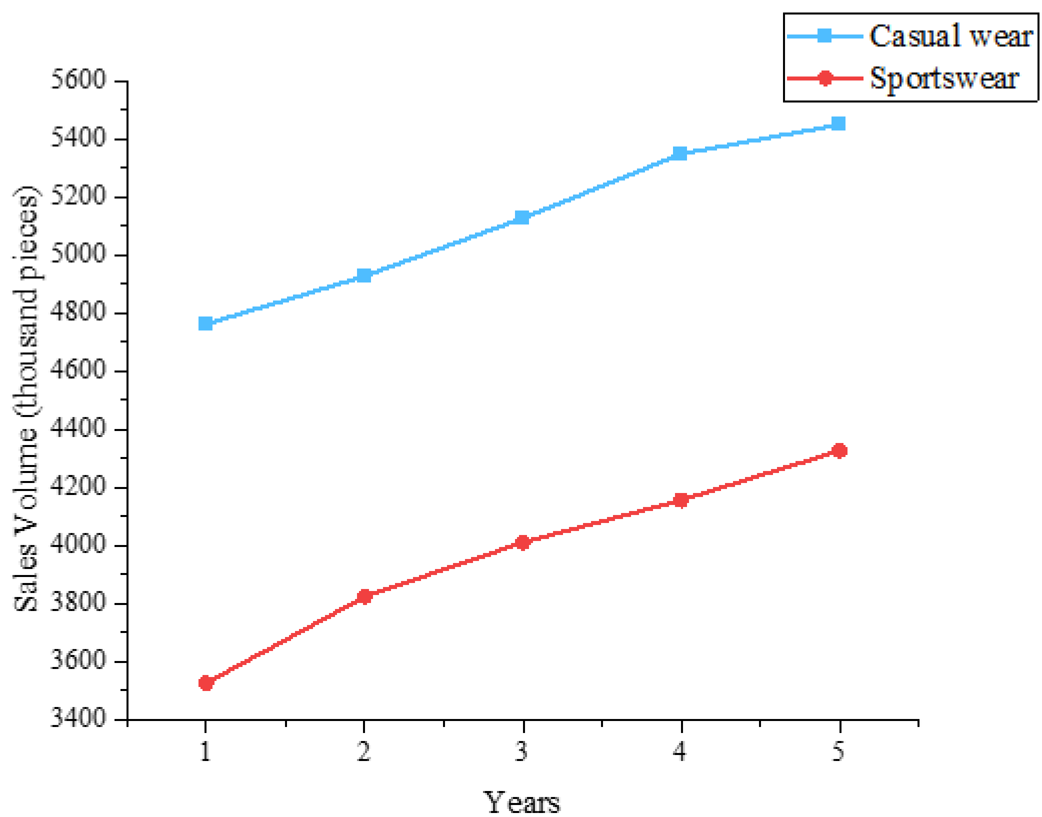


Figure 11. Clothing industry trend prediction



CONCLUSION

This study leverages the capabilities of deep learning in handling both image and sequence information. It utilizes product images and sales sequences as variables to train a deep neural network model for predicting product sales and subsequently forecasting industry trends. Based on the characteristics of structured data, sequence data, and image data, this study employs three types of neural network models: FCNN, Bi-LSTM, and CNNs to construct an asymmetric neural network prediction model. By effectively handling and using the information from structured data, sales data, and image data, the proposed prediction model can better capture the complexity and comprehensiveness of the input data, providing more accurate predictive outcomes. It. Throughout the training process, this study rigorously validated various methods in deep learning and carefully controlled the model's training procedure. Furthermore, the removal of certain modules results in varying degrees of decline in sales-prediction accuracy, emphasizing the necessity of each module in the proposed sales-prediction model. Finally, based on the introduced deep-learning model, in-depth predictions of sales volumes for the clothing industry were conducted, demonstrating the rationality and feasibility of the model. The deep-learning approach employed in this study achieves high prediction accuracy; however, the interpretability of the model is limited. Extracting easily understandable features from the neural network could be a future research direction. Additionally, the training of the predictive model in this study requires a substantial amount of time. Enhancing the efficiency of the model-training process remains an area for potential improvement in this work.

FUNDING

This work was supported by Social Science Planning Project of Zhejiang Province (21NDJC311YBM), Soft Science Research Project of Zhejiang Province (2021C35072).

REFERENCES

Angeli, A., Piccolomini, E. L., & Marfia, G. (2018). Learning about fashion exploiting the big multimedia data. In *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)* (pp. 48–51). IEEE. doi:10.1109/PIMRC.2018.8581037

Aras, S., Deveci Kocakoç, İ., & Polat, C. (2017). Comparative study on retail sales forecasting between single and combination methods. *Journal of Business Economics and Management*, *18*(5), 803–832. doi:10.3846/16111699.2017.1367324

Bates, J. M., & Granger, C. W. J. (1969). The combination of forecasts. *The Journal of the Operational Research Society*, *20*(4), 451–468. doi:10.1057/jors.1969.103

Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, *5*(2), 157–166. doi:10.1109/72.279181 PMID:18267787

Brzęczek, T. (2016). Using portfolio theory to predict the impact of reduction in product width on sales. *Journal of Business Economics and Management*, *17*(6), 1222–1236. doi:10.3846/16111699.2015.1120769

Chern, C. C., Chen, W. Y., & Hsiao, B. (2023). A time series analysis approach to forecast price fluctuation points. *International Journal of Management Science and Engineering Management*, *19*(1), 67–77. doi:10.1080/17509653.2023.2174199

Fallah Tehrani, A., & Ahrens, D. (2018). Enhanced predictive models for purchasing in the fashion field by applying regression trees equipped with ordinal logistic regression. In S. Thomassey & X. Zeng (Eds.), *Artificial intelligence for fashion industry in the big data era; Springer series in fashion business* (pp. 27–45). Springer. doi:10.1007/978-981-13-0080-6_3

Gao, T., Wang, C., Zheng, J., Wu, G., Ning, X., Bai, X., Yang, J., & Wang, J. (2023a). A smoothing Group Lasso based interval type-2 fuzzy neural network for simultaneous feature selection and system identification. *Knowledge-Based Systems*, *280*, 111028. doi:10.1016/j.knosys.2023.111028

Gao, Y., Wang, J., Li, Z., & Peng, Z. (2023b). The social media big data analysis for demand forecasting in the context of globalization: Development and case implementation of innovative frameworks. [JOEUC]. *Journal of Organizational and End User Computing*, *35*(3), 1–15. doi:10.4018/JOEUC.325217

Gardner, E. S. Jr. (1985). Exponential smoothing: The state of the art. *Journal of Forecasting*, *4*(1), 1–28. doi:10.1002/for.3980040103

Giri, C., & Chen, Y. (2022). Deep learning for demand forecasting in the fashion and apparel retail industry. *Forecasting*, *4*(2), 565–581. doi:10.3390/forecast4020031

Hong, J. K. (2021). LSTM-based sales forecasting model. *KSII Transactions on Internet and Information Systems*, *15*(4), 1232–1245. doi:10.3837/tiis.2021.04.003

Hsieh, P. H. (2019). A study of models for forecasting e-commerce sales during a price war in the medical product industry. In F. F. Nah & K. Siau (Eds.), Lecture Notes in Computer Science: Vol. 11588. *HCI in Business, Government and Organizations. eCommerce and Consumer Behavior: HCII 2019* (pp. 3–21). Springer. doi:10.1007/978-3-030-22335-9_1

Hsu, J. Y., Tseng, W. K., Hsieh, J. Y., Chang, C. J., & Chen, H. (2020). The recommending agricultural product sales promotion mode in e-commerce using reinforcement learning with contextual multiarmed bandit algorithms. *Mathematical Problems in Engineering*, *8836000*, 1–10. doi:10.1155/2020/8836000

Ji, B., & Zhang, Y. (2023). Few-shot relation extraction model based on attention mechanism induction network. [Science Edition]. *Journal of Jilin University*, *61*(4), 845–852.

Jian, Z., Qingyuan, Z., & Liying, T. (2020). Market revenue prediction and error analysis of products based on fuzzy logic and artificial intelligence algorithms. *Journal of Ambient Intelligence and Humanized Computing*, *11*(10), 4011–4018. doi:10.1007/s12652-019-01650-2

Lee, H., Kim, S. G., Park, H. W., & Kang, P. (2014). Pre-launch new product demand forecasting using the Bass model: A statistical and machine learning-based approach. *Technological Forecasting and Social Change*, *86*, 49–64. doi:10.1016/j.techfore.2013.08.020

Mosteller, J., Donthu, N., & Eroglu, S. (2014). The fluent online shopping experience. *Journal of Business Research*, *67*(11), 2486–2493. doi:10.1016/j.jbusres.2014.03.009

Nassibi, N., Fasihuddin, H., & Hsairi, L. (2023). Demand forecasting models for food industry by utilizing machine learning approaches. *International Journal of Advanced Computer Science and Applications*, *14*(3). doi:10.14569/IJACSA.2023.01403101

Sano, N., Yada, K., & Suzuki, T. (2014). Category evaluation method for business intelligence using a hierarchical Bayes model. In S. Patel, Y. Wang, W. Kinsner, D. Patel, G. Fariello, & L. A. Zadeh (Eds.), *2014 IEEE 13th International Conference on Cognitive Informatics and Cognitive Computing* (pp. 400–407). IEEE. doi:10.1109/ICCI-CC.2014.6921490

Thomassey, S., & Happiette, M. (2007). A neural clustering and classification system for sales forecasting of new apparel items. *Applied Soft Computing*, *7*(4), 1177–1187. doi:10.1016/j.asoc.2006.01.005

Tian, L., & Wang, X. (2022). A dynamic prediction neural network model of cross-border e-commerce sales for virtual community knowledge sharing. *Computational Intelligence and Neuroscience*, *2529372*, 1–11. doi:10.1155/2022/2529372 PMID:36262614

Tian, S., Li, W., Ning, X., Ran, H., Qin, H., & Tiwari, P. (2023). Continuous transfer of neural network representational similarity for incremental learning. *Neurocomputing*, *545*, 126300. doi:10.1016/j.neucom.2023.126300

Ungureanu, S., Topa, V., & Cziker, A. C. (2021). Deep learning for short-term load forecasting—Industrial consumer case study. *Applied Sciences (Basel, Switzerland)*, *11*(21), 10126. doi:10.3390/app112110126

Vavliakis, K. N., Siailis, A., & Symeonidis, A. L. (2021). Optimizing sales forecasting in e-commerce with ARIMA and LSTM models. In *Proceedings of the 17th International Conference on Web Information Systems and Technologies—WEBIST* (pp. 299–306). SciTePress. doi:10.5220/0010659500003058

Wang, C. H., & Chen, T. Y. (2022). Combining biased regression with machine learning to conduct supply chain forecasting and analytics for printing circuit board. *International Journal of Systems Science: Operations & Logistics*, *9*(2), 143–154. doi:10.1080/23302674.2020.1859157

Xie, N. M., & Liu, S. F. (2005). Discrete GM(1,1) and mechanism of grey forecasting model. *Systems Engineering—Theory & Practice, 25*(1), 93–99.

Zhang, C., Zhang, D., & Zheng, C. (2022). Research on the application of auto spare parts sales forecast in the age of big data. In *2022 International Conference on Computers and Artificial Intelligence Technologies (CAIT)* (pp. 48–52). IEEE. doi:10.1109/CAIT56099.2022.10072156

Zhang, Y., & Shankar, A. (2023). Enhancing supply chain transparency and risk management using CNN-LSTM with transfer learning. [JOEUC]. *Journal of Organizational and End User Computing*, *35*(1), 1–22. doi:10.4018/JOEUC.333472