Computer Science and Information Systems 22(3):1197–1228 https://doi.org/10.2298/CSIS241123038L

Smart Home Management Based on Deep Learning: Optimizing Device Prediction and User Interface Interaction

Xuan Liang[1,2], Meng Liu[1,4], Hezhe Pan[3,*]

[1] Art College, Chongqing Technology And Business University, Chongqing, 400067, China
liangxuan@ctbu.edu.cn
[2] School of Design, Hunan University, Changsha, 410082, China
[3] Loudi Vocational and Technical College, Loudi, 417000, China
panhezhe001@163.com
[4] Chongqing Vocational College of Media, Chongqing, 400020, China
liumeng6@ctbu.edu.cn

Abstract. This work aims to address the challenges faced by smart home systems, including the accuracy of device status prediction, user interface design, system stability, and response speed. As smart home devices become more widely used, the need for accurate predictions of their operational status has increased. This includes predicting the switch states, faults, and performance metrics of devices such as smart lights, thermostats, and security systems. To address this demand, an innovative multimodal prediction model combining the Convolutional Neural Network and Long Short-Term Memory network is proposed to enhance the accuracy of smart device status predictions. Cloud computing technology is used for the user interface design to create an intuitive and user-friendly interface, ensuring both system stability and fast response times. The experiments compare the performance of the proposed model with traditional models in predicting the status of smart devices. The results demonstrate that the proposed system reduces the Mean Squared Error and Mean Absolute Error by 20% and 15%, respectively, significantly improving prediction performance. Furthermore, user satisfaction surveys indicate a 25% increase in satisfaction with the system. The proposed system also reduces the utilization rates of the Central Processing Unit, memory, Graphics Processing Unit, and network bandwidth by 15%, 18%, 25%, and 20%, respectively. These findings highlight the system's advantages in accuracy, user satisfaction, and resource utilization efficiency, providing strong support for the design and application of smart home systems.

Keywords: artificial intelligence; cloud computing; smart home; multimodal prediction model; user satisfaction.

1. Introduction

1.1. Research Background and Motivations

The rapid advancement of technology has made smart home systems an integral part of daily life [1-3]. As user expectations for these systems rise, especially in terms of the accuracy of device status predictions, current smart home technologies face significant challenges. These include limited capacity to process multimodal data and poor real-time response performance. Many traditional models struggle to predict device statuses accurately in complex environments, which impacts the user experience and limits the potential applications of these systems [4-6]. However, innovations in artificial intelligence (AI) and the improved processing capabilities of cloud computing have created new opportunities for home automation [7]. AI technologies, particularly machine learning and deep learning algorithms, enable home devices to learn and adapt to user preferences, providing personalized services [8-10]. Cloud computing offers powerful computational and storage support, facilitating remote connections and seamless data sharing among devices [11-13]. The integration of AI and cloud computing offers great potential for developing intelligent, efficient, and secure home management systems [14-16].

Despite significant advancements in both academia and industry, several challenges persist in the development of smart home systems [17]. One of the primary issues is the diversity of home devices and the lack of standardized protocols, which have led to interoperability problems [18, 19]. Additionally, the real-time performance and accuracy of intelligent algorithms still require improvement [20, 21]. Another critical concern is the safeguarding of user privacy and data security [22, 23]. Traditional smart home systems are limited by the capabilities of their intelligent algorithms, particularly in terms of real-time performance and accuracy. These limitations hinder the system's ability to make prompt and precise decisions, reducing both the user experience and the system's overall effectiveness. To address this, the focus of this work is on integrating machine learning and deep learning algorithms, optimizing them to improve the system’s real-time responsiveness and enhance its ability to adapt accurately to user habits in varied environments. Furthermore, the smart home market is fragmented, with various device brands and standards contributing to interoperability challenges. This issue makes it difficult for devices from different manufacturers to work seamlessly together, and compromises the integrated functioning of the system. Finally, smart home systems handle large amounts of sensitive data, such as information about household routines and user preferences, raising significant concerns regarding privacy and data security.

In summary, current smart home systems still face many challenges in device status prediction, user interface design, system stability, and response speed. With the widespread adoption of smart home devices, accurately predicting the operational status of devices (such as smart lighting, thermostats, security system switches, faults, and performance indicators) has become an urgent problem to address. However, existing smart home systems still have certain shortcomings in these aspects, particularly in the accuracy of device status prediction and the user-friendliness of the interface. Therefore, this work aims to propose an innovative multimodal prediction model that combines the Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) network to improve the accuracy of device status prediction. Additionally, it incorporates cloud computing technology into the user interface design to enhance system stability and response speed, ensuring a better user experience.

The research motivation arises from an in-depth analysis of the shortcomings of current smart home systems, particularly the inaccuracy of device status prediction and the complexity of user interfaces, which affect user experience and system efficiency. Through the proposed multimodal model and optimized user interface design, this work aims not only to improve prediction accuracy but also to make significant improvements in user experience, system stability, and resource utilization efficiency. This innovative approach is expected to provide strong support for the design and application of smart home systems and promote the popularization and development of smart home technology in practical applications. This work prioritizes the design of a secure and reliable system to protect user privacy and mitigate potential threats. It incorporates encryption technologies, access control, and secure transmission protocols, while also implementing robust security measures within a cloud computing environment to defend against cyberattacks and data breaches. By leveraging an innovative multimodal predictive model, cloud computing for user interface design, and optimizing resource utilization, this work offers a comprehensive solution for developing whole-home intelligent management systems. The goal is to integrate AI technology with cloud computing to create a highly intelligent, secure, and reliable system. This system not only enables the intelligent control of home devices but also facilitates seamless information sharing and decision-making across devices. It significantly enhances the user experience in smart homes and lays a strong foundation for the continued evolution of smart home technologies.

1.2. Research Objectives

This work aims to design a highly intelligent home control system to enable automated control and intelligent scheduling of household devices. It also leverages cloud computing technology to create a secure and stable data platform, facilitating information sharing and remote control across devices. Through the development of smart algorithms, this work enables real-time monitoring and analysis of the home environment, providing personalized services. The approach enhances the system’s scalability and interoperability, ensuring compatibility and seamless integration with smart devices from various manufacturers. Finally, real-world scenario validation ensures the system’s stability and practicality. These innovations not only represent significant technological breakthroughs but also contribute to improved system performance, user experience, and resource utilization efficiency. They provide a solid foundation for the design and deployment of whole-house smart management systems, opening new possibilities for the future of smart home technology.

2. Literature Review

In the field of home automation, a wide range of commercial products has emerged, showcasing advanced smart control capabilities. In the area of intelligent lighting systems, products like the Philips Smart Lighting System and Yeelight Smart Bulbs allow users to remotely adjust brightness, color temperature, and other settings via smartphone apps or voice assistants. In the domain of smart security, products such as the Ring Smart Video Doorbell and Nest Secure Smart Security System offer features like video monitoring, intrusion detection, and intelligent doorbells, enabling real-time home security management through smartphones. In the smart home appliance sector, products like the LG ThinQ series and Samsung Smart Refrigerators not only allow for remote monitoring but also incorporate intelligent features that adapt to user habits. Smart speaker systems, such as the Amazon Echo and Google Home, integrate voice assistants. They enable users to control smart home devices through voice commands and access a wide range of information and entertainment services. Additionally, various commercial products are available across other smart home domains, including smart temperature control, home automation, smart curtains and windows, entertainment systems, health monitoring, and kitchen appliances, covering nearly every aspect of modern living. These products demonstrate that smart home technology has become a practical and accessible solution, offering users enhanced convenience and intelligence in their homes. A significant body of research is dedicated to developing smart home systems that incorporate AI technology and cloud computing [24]. For example, literature [25] introduced the concept of a "user-friendly Internet of Things (IoT) for everyday living" in their design approach, creating an IoT-based smart home system. This system enabled functions such as displaying temperature and humidity data collected from node boards on a personal computer (PC) via a web browser. It also allowed users to control the on/off states of Light Emitting Diode lights through the same interface. In a similar vein, scholars proposed a system connecting sensors, actuators, and other data sources to enable more complex home automation tasks [26]. They also developed a smartphone application that allowed users to control various household appliances and sensors remotely.

Literature [27] developed a powerful and intelligent floor monitoring system using highly reliable frictional electric encoding pads and DL-assisted data analysis. They further integrated deep learning-assisted data analysis to enhance the system’s capabilities for various smart home monitoring and interactions. Literature [28] proposed a fully operational 46-inch smart textile lighting/display system. This system incorporated embedded optical fiber devices designed to detect external stimuli. Literature [29] introduced a comprehensive smart home aggregation system based on IoT and edge computing. The system leveraged edge AI support technology and adhered to industry standards for fog computing, providing robust responses from connected IoT sensors in typical smart homes. Literature [30] designed a secure remote user authentication scheme called SecFHome. This scheme supports secure communication at the network edge and enables remote authentication in fog-based smart home systems. Literature [31] presented an IoT-based smart home management system. The system utilized sensors, actuators, smartphones, network services, and microcontrollers for enhanced functionality.

Research on data privacy protection in smart homes has garnered significant attention, particularly focusing on the application of the Deep Deterministic Policy Gradient (DDPG) algorithm as a core predictive model in modern power systems. This approach enhanced prediction accuracy [32]. The DDPG predictive model is later integrated into the federated learning framework. The resulting Federated Deep Reinforcement Learning (FedDRL) model mitigates privacy concerns by sharing model parameters instead of private data, ensuring accurate predictive models are obtained in a decentralized manner. Literature [33] proposed a False Data Injection Attack (FDIA) detection method based on secure federated deep learning. The method's effectiveness and superiority were demonstrated through extensive experiments on IEEE 14-bus and 118-bus test systems. Literature [34] addressed privacy concerns in smart home technology, particularly the risks of data leakage through wireless signal eavesdropping. They discussed "FTS (fingerprint and timing-based snooping)" attacks, a type of side-channel attack that could passively infer activity information within residences. These attacks can be executed remotely near the target house. Literature [35] applied the Sovereign design philosophy to enable communication between home IoT devices and applications via application-named data, directly protecting the data. The results indicated that Sovereign offers a systematic, user-controlled solution for self-contained smart homes, with minimal observable overhead when running on existing IoT hardware. Literature [36] proposed a location privacy security mechanism based on anonymous trees and box structures. This approach provided location privacy protection for services targeting smart terminals.

In the field of smart homes, numerous studies have examined the integration of AI technology with cloud computing, driving advancements in the design and implementation of smart home systems. These studies often focus on specific applications, such as smart lighting and environmental monitoring [37]. However, they tend to lack a comprehensive analysis and optimization of the entire smart home ecosystem. This gap indicates that, while smart home systems offer convenience to users, there is still significant room for improvement in their overall performance. Key factors, such as the accuracy of device status predictions, the intuitiveness of user interfaces, and system response speed, have not been sufficiently explored. To address these challenges, this work aims to provide a holistic analysis and optimization of smart home systems, ultimately enhancing both performance and user experience. The primary objective is to improve the accuracy of smart device status predictions through an innovative multimodal predictive model that combines CNN with LSTM networks. In terms of user interface design, this work utilizes cloud computing technology to enable seamless data sharing and collaborative computation between devices. This approach creates an intuitive and user-friendly interface while ensuring system stability and responsiveness. Furthermore, by analyzing user behavior data, this work offers personalized services that make the smart home system more closely aligned with individual user needs.

3. Research Methodology

In smart home systems, effective coordination between the multimodal predictive model and the Deep Q Network (DQN) intelligent control algorithm is essential. Each component has a distinct role, and together, they enable accurate device status prediction and optimized control. The multimodal predictive model, which combines CNN and LSTM, processes complex data from various smart devices. It generates predictions about device statuses, providing a comprehensive view of the system's current and future states. The DQN serves as the intelligent control algorithm, making real-time decisions based on the outputs of the predictive model. Through continuous learning and optimization, the DQN selects the best control strategy to adjust the device's operating states. Central to its operation is the Q-value function, which estimates the expected rewards of different actions in various states. This function guides the system in selecting the most effective actions to optimize device control. The integration of the predictive model and the DQN allows the system to both predict device statuses accurately and make intelligent decisions on how to adjust operations. The predictive model interprets complex multimodal data to provide precise insights into future device states, while the DQN uses these insights to refine control strategies. This collaboration enables the smart home system to not only forecast device states in real time but also dynamically adjust operations, creating a more intelligent and responsive home environment. The following sections provide further details on the components and functioning of these processes.

3.1. Construction of Smart Device State Prediction Model

To enable intelligent control and optimization of home devices, this work designs a smart device status prediction model that combines LSTM networks with CNN. This model intends to enhance both the prediction accuracy and response speed for smart device statuses. The choice of this combined approach is based on its proven ability to improve predictive performance effectively. Although models such as Autoregressive Integrated Moving Average (ARIMA) and Deep Neural Network (DNN) are commonly used for time series forecasting, they are less suited for the specific task of predicting smart home statuses. The ARIMA model struggles with nonlinear time series data, particularly when the data involves seasonality or abrupt events, which limits its predictive accuracy [38]. DNN fails to capture short-term memory as efficiently as LSTM networks, resulting in lower accuracy and poorer real-time performance.

To address these issues, this work selects an innovative multimodal predictive model that combines CNN and LSTM. This model effectively manages the complexity and dynamics of smart home systems and provides more accurate status predictions. Given the complexity of intelligent device states, which are influenced by various factors, a deep learning model is selected for its ability to handle such data. Single deep learning models, like CNN or LSTM alone, may struggle to capture the complex relationships in spatiotemporal data due to the multi-modal nature of device states. To comprehensively leverage the features of image and time series data, a multimodal prediction model that integrates CNN and LSTM is chosen. This approach enables a more comprehensive understanding of both image and time-series data features. Although ensemble learning methods such as Random Forest or Gradient Boosting Trees, and model fusion methods like Stacking, can provide strong performance, they often require extensive tuning and feature engineering [39]. Considering the goal of exploring the complex relationships between image and time-series data, the CNN-LSTM integrated model is ultimately chosen. While rule-based methods could predict device states, they rely on predefined domain-specific knowledge and struggle to adapt to evolving patterns in the data.

By considering information in both spatial and temporal dimensions, the model better adapts to the complexities of home scenarios. This work adopts a multimodal prediction model to better accommodate the data features and predictive requirements of smart home systems. This model can handle time-series data and multi-sensor image data to accurately predict the state of household devices, providing the foundation for intelligent control. Figure 1 is an illustrative diagram of the model.

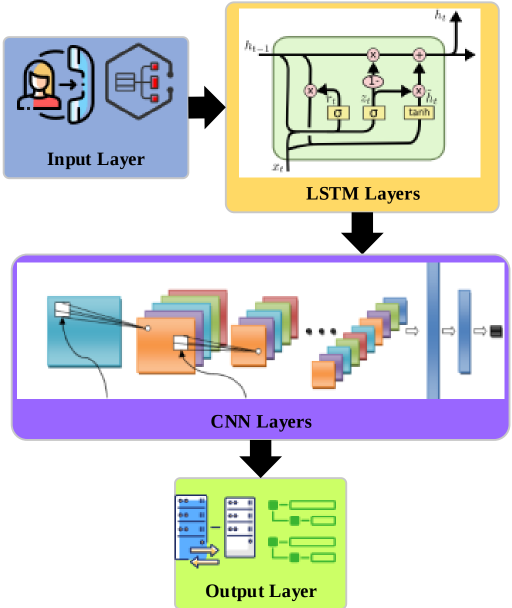

Fig. 1. Schematic diagram of the smart device state prediction model

The model in Figure 1 consists of four main components. The input layer receives data from both time-series and multi-sensor image sources. Time-series data, such as temperature and humidity, are processed through the LSTM network. Meanwhile, image data, such as infrared images, are processed by the CNN. The LSTM layer handles time-series data, focusing on the sequential nature of the information and capturing long-term dependencies. The CNN layer processes the image data, extracting spatial features from the multi-sensor images to enhance the prediction of device states. Finally, the output layer generates predictions for the smart device states.

In processing image data, CNN extracts spatial features through convolution operations. It is assumed that the input image data are denoted by X. H is the height of the image, W is the width, and C is the number of channels (such as three channels for RGB images, C = 3). The convolutional layer operates on the image using a convolution kernel (filter) to generate a feature map. The mathematical equation for the convolution operation is as follows:

$$Y_{i,j} = \sum_{m=1}^{M} \sum_{n=1}^{N} X_{i+m,j+n} \cdot K_{m,n} \tag{1}$$

In this process, $Y_{i,j}$ represents the element of the output feature map, which indicates the value after the convolution operation. X is the input image, K is the convolution kernel, and M and N are the size of the kernel; i and j are the indices of the feature map. The convolution operation is used to extract local spatial features, with the convolution kernel performing a dot product with the input image through a sliding window, generating a new feature map. These feature maps capture basic structures of objects in the image, such as edges and corner points. After the convolution operation, a pooling operation is usually applied to further reduce the size of the feature map while retaining important spatial information. The pooling operation can be performed using either MaxPooling or AveragePooling, with the corresponding equation as follows:

$$Y_{i,j} = \max\left(X_{i:i+k,j:j+k}\right) \tag{2}$$

In this process, k represents the size of the pooling window, and $X_{i:i+k,j:j+k}$ denotes the local region extracted from the input image. Pooling operations help reduce dimensionality and computational load, while also preventing overfitting. Through multiple layers of convolution and pooling operations, CNN can progressively extract more complex spatial features, such as the shape and position of objects. Ultimately, these features are aggregated in the fully connected layer and used for decision-making in device state prediction.

The outputs from both the LSTM and CNN are integrated to provide a comprehensive analysis of device states. This integration allows the model to collect information from multiple data sources, and improves both prediction accuracy and precision. By utilizing multi-layered and multi-source data, the smart device state prediction model offers a more accurate forecast of household device statuses. This, in turn, provides a reliable foundation for intelligent home system control.

The input for the LSTM model is based on time-series data from smart devices. These data include the device's historical status, sensor readings (such as temperature, humidity, and brightness), and other relevant features. To ensure the LSTM can effectively learn from these inputs, the raw data are organized and structured so that each time step corresponds to the appropriate sensor information and historical status. Specifically, data from the previous 10 time steps are selected for each point in time, and a sliding window method is used to generate training samples. These samples are then fed into the LSTM model for status prediction. The LSTM, a type of recurrent neural network with memory units, is designed to remember long-term dependencies [40-42]. In this case, it processes time-series data, such as device status information like temperature and humidity. The LSTM's architecture includes input, forget, and output gates, which enable it to capture long-term dependencies within the time-series data [43]. The mathematical expression is as follows:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{3}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{4}$$

$$\tilde{C}_t = \tanh\,(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{5}$$

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{6}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{7}$$

$$h_t = o_t \cdot tanh\,(C_t) \tag{8}$$

x_t represents the input data, h_t is the current time-step hidden state, and C_t is the current time-step cell state. f_t, i_t, and o_t are the outputs of the forget gate, input gate, and output gate, respectively. W and b denote the weight and bias. σ is the sigmoid activation function, and tanh is the hyperbolic tangent activation function. In the smart home system context, this work uses CNN to process multi-sensor image data, such as infrared images. The CNN network's structure includes convolutional layers, pooling layers, and fully connected layers. These components work together to effectively extract spatial features from the images [44-46]. The expression reads:

$$\mathrm{h_i} = \sigma(\mathrm{W} * \mathrm{x_i} + \mathrm{b}) \tag{9}$$

$$\mathrm{y} = \mathrm{softmax}\ (\mathrm{h}) \tag{10}$$

x_i is the i-th region of the input image, and h_i is the feature representation of that region. W represents the convolutional kernel, and b is the bias. σ is the ReLU activation function, $*$ denotes the convolution operation, and the softmax function is used for multi-class output.

3.2. Selection and Optimization of Intelligent Control Algorithms

Next, this work explores the selection and optimization of intelligent control algorithms. This section conducts an in-depth analysis of how to choose the most suitable control algorithm for smart home environments and optimize control strategies using reinforcement learning techniques. By working in synergy with the predictive model, the control algorithm can respond in real-time to changes in device status and make optimal decisions. This can effectively improve the operational efficiency and user experience of the smart home system. This work selects the DQN algorithm from deep reinforcement learning as the intelligent control algorithm for the smart home system. DQN is known for its strong generalization and learning capabilities, making it well-suited for handling large-scale, high-dimensional state spaces [47-49]. The core idea behind DQN is to construct a Q-value function that represents the value of taking a specific action in a given state. A neural network is then used to approximate this Q-value function, which is essential for action prediction and selection [50]. The mathematical expression is as follows:

$$\mathrm{yQ(s,a)} = (1-\alpha)\cdot \mathrm{Q(s,a)} + \alpha \cdot (\mathrm{r} + \gamma \cdot \mathrm{maxQ(s',a')}) \tag{11}$$

$Q(s,a)$ is the Q-value for taking action a in state s, α represents the learning rate, r is the immediate reward, and γ is the discount factor. s' is the next state, and a' is the best action in the state s'. To improve the efficiency and performance of the control algorithm, this work deploys it on cloud computing resources and leverages big data for optimization. The high-performance cloud infrastructure ensures real-time algorithm capabilities, while big data analysis uncovers additional control patterns and optimization strategies. Figure 2 depicts the structure of the DQN algorithm.

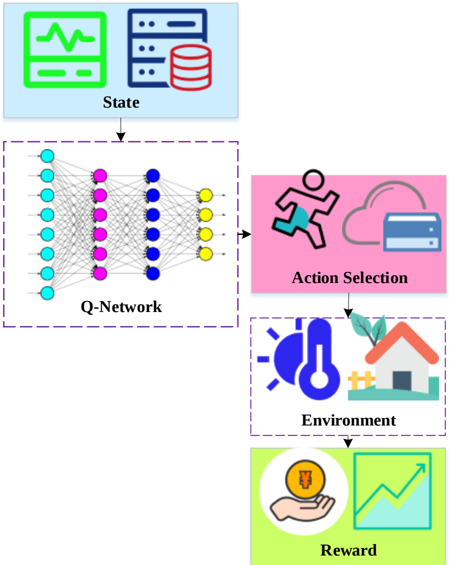

Fig. 2. Schematic diagram of the DQN algorithm structure.

In Figure 2, the State represents the current environmental states of the smart home system, including sensor data and device statuses. The Q-Network is a neural network used to estimate Q-values for specific actions in a given state. The network takes the current state as input and outputs Q-values for possible actions. "Action Selection" chooses the next action based on the Q-values using an "ε-greedy" strategy, where, with a certain probability (ε), actions are randomly selected to increase exploration. With a probability of 1-ε, the action with the highest Q-value is chosen to enhance exploitation. Environment represents the physical environment of the smart home system, including various sensor data and device statuses. The Reward is the immediate reward signal obtained by the smart home system based on the actions taken by the intelligent control algorithm. This structure allows the DQN algorithm to learn the optimal control strategy through continuous interaction with the environment.

3.3. Design of an AI and Cloud Computing-Based Smart Home Management System

This section demonstrates how to integrate the previously discussed predictive model with intelligent control algorithms into a complete system. The system not only performs device status prediction and intelligent control but also leverages cloud computing technology to store, process, and analyze data, providing more efficient and flexible management and services. Cloud computing technology is primarily utilized to enhance the flexibility and responsiveness of the system interface. By offloading data processing and storage to the cloud, the burden on local devices is reduced. This allows the smart home system to provide faster information access and a smoother user experience. Cloud computing also supports real-time collaborative computation among different smart devices, enhancing their interoperability and coordination. Moreover, it enables users to remotely access and control home devices, further improving system operability and user satisfaction. Overall, the implementation of cloud computing technology makes smart home systems more resilient, scalable, and collaborative. It enhances the intuitiveness and user-friendliness of the interface, which directly boosts the user experience [51]. These improvements can be measured through methods such as user satisfaction surveys and interface interaction analysis. This work presents a smart home management system based on AI and cloud computing, as illustrated in Figure 3.

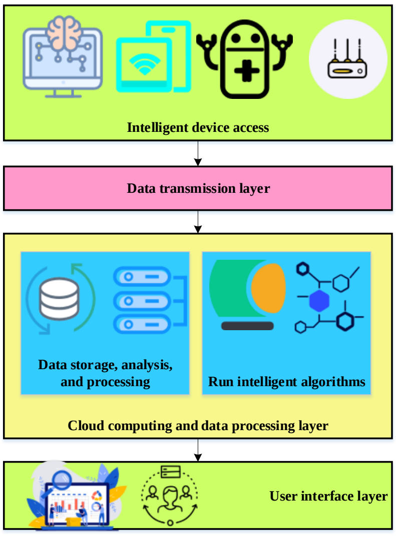

Fig. 3. Framework of the smart home management system

Figure 3 illustrates the overall architecture of the management system, which consists of several key components. The intelligent device access layer communicates with various smart devices, such as sensors and actuators, to collect real-time data. The data transmission layer ensures the timely and reliable transfer of these data to the cloud computing and data processing layer. This layer handles tasks like data storage, analysis, and processing, running intelligent algorithms to support decision-making. The user interface layer provides a platform for users to interact with the system, allowing them to monitor and control household devices via mobile apps, web interfaces, and other means. With this structural composition, the work can achieve the following functions:

1) Intelligent device access and management: it facilitates communication and access to various smart devices, ensuring they operate properly.

2) Real-time data transmission and processing: sensor data are transmitted to the cloud in real-time. The cloud processes the data through cleaning and feature extraction.

3) Intelligent control and optimization: cloud computing’s high-performance computing capabilities are used to run algorithms that control and optimize household devices, improving energy efficiency.

4) Data storage and analysis: processed data are stored in a cloud-based database for further analysis, mining, and visual representation.

5) Remote monitoring and operation: users can remotely monitor device status and perform operations through mobile applications or web interfaces.

This work introduces the Generative Adversarial Network (GAN) to enable continuous learning and adaptability of the predictive model to respond to changes in new data and user behavior. Specifically, GAN consists of two main components: the generator and the discriminator. The generator's task is to generate data based on actual user behavior patterns, while the discriminator is responsible for evaluating the similarity between the generated data and real data. In the initial stage, the generator learns the patterns in user behavior data to generate preliminary behavior data, while the discriminator learns to distinguish between generated and real data. This process helps the model understand and capture the basic patterns of user behavior. Figure 4 displays the specific process.

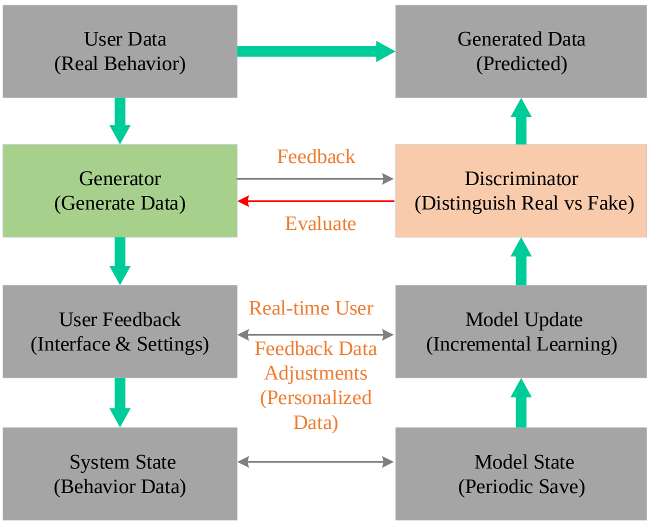

Fig. 4. GAN structure and workflow

During the system's operation, real-time incoming user behavior data are used for incremental learning. The generator and discriminator will continuously update to adapt to new data characteristics. This continuous learning mechanism ensures that the model can timely acquire information from new data and adjust according to changing user behavior. By dynamically optimizing the GAN parameters, the model can flexibly respond to these changes, ensuring the accuracy of predictions. Furthermore, the system employs a real-time feedback mechanism to further optimize the generated data. User feedback, such as interface ratings and the selection of personalized settings, is used to guide the model in adjusting the generated data, making it more aligned with the users' actual needs and preferences. This mechanism ensures the timeliness and precision of the generated data, enhancing the adaptability and accuracy of the predictive model. In short, the role of GAN here is to assist the predictive model in continuously learning, adapting to new data, and adjusting in real time to respond to changes in user behavior through generating and evaluating data. In this way, GAN effectively strengthens the intelligent home system's adaptability to user needs and the accuracy of predictions.

The incremental learning strategy enables the GAN to integrate new knowledge dynamically during the learning process. The model's state is saved periodically, allowing for rollback or recovery when needed, ensuring system stability. To evaluate performance, the system regularly compares generated data to real data and considers user satisfaction metrics. User feedback plays a crucial role in assessing system performance, helping identify areas for improvement and model adjustments. By analyzing user behavior patterns, the system detects both short-term and long-term trends, which helps predict user needs and adjust the status of smart devices accordingly. The system uses a personalized learning model to tailor predictions based on each user's unique behavior and preferences. Real-time data from sensors and devices are collected regularly to update the knowledge base, reflecting the current environment and user behavior. This combination of mechanisms allows the system to continuously learn and improve its understanding of user behavior, delivering more intelligent and personalized home management services. Its dynamic adaptability ensures high performance, even during long operational periods.

To clearly illustrate the operation process of the intelligent home system proposed, an example of a smart lighting system within a smart home environment is presented. It is assumed that the system needs to predict the state of the light (on or off) and make corresponding control decisions. In this scenario, the smart home system first collects data from multiple sensors, including indoor light intensity, temperature, user activity data, and the historical on/off status of the lights. These data are processed through a multimodal predictive model. CNN is responsible for extracting spatial features from the environmental data, and LSTM networks handle the time-series data, capturing the usage patterns and historical dependencies of the lights. By combining both approaches, the system can accurately predict the state of the light for the upcoming period. For example, based on the data from the past hour, the predictive model might conclude that the light will remain on in the near future.

Next, the DQN intelligent control algorithm utilizes these predictions to make control decisions. The system's state space includes the current state of the light, environmental lighting conditions, temperature, and user activity status. Based on this information, DQN will choose the most appropriate action, such as "turn on" or "turn off" the light. The selection of each action is determined by a reward function, which not only considers the current state of the light but also scores based on user needs and system energy efficiency. For instance, if it is predicted that a user is entering the room, the system will select the "turn on" action because it enhances the user's experience and results in a higher reward.

Through continuous reinforcement learning, DQN can optimize its control strategy. If the system finds that a certain control strategy (such as delayed light-off) performs excellently in terms of energy saving, DQN will adjust its decisions based on accumulated rewards, thus improving the overall system performance. This collaborative approach enables the predictive model and control algorithm to dynamically adjust and optimize the device's state. It can ensure optimal performance of the smart home system in terms of response speed, energy efficiency, and user experience. This specific example provides a clearer understanding of how the proposed approach effectively collaborates within a smart home system. It also offers a more intuitive grasp of how the predictive model and DQN intelligent control algorithm work together to optimize device management.

4. Experimental Design and Performance Evaluation

4.1. Datasets Collection and Data Preprocessing

To validate the performance of the proposed smart home system, various types of environmental data and sensor information are systematically collected from multiple rooms in a residential community. These rooms include the living room, kitchen, bedroom, bathroom, and study/office area, covering the main living scenarios within a household. Environmental data are collected in real time using temperature, humidity, and light sensors, with a recording frequency of once per minute. Specifically, the sensor data cover the following environmental factors. Temperature sensors are deployed in each room to monitor indoor temperature changes in real time, recording data once per minute to capture rapid temperature fluctuations. Humidity sensors are placed in various locations to record changes in indoor humidity, particularly in areas with significant humidity variations, such as the kitchen and bathroom. Light sensors are used to monitor indoor light intensity, especially in the living room and bedroom, to assist in controlling the smart lighting system. In addition, multi-sensor image data are collected through internal cameras and infrared sensors, including object distribution information. The data provide insights into the placement of furniture and equipment within the home, helping the model understand the spatial layout. Human activity monitoring: The data track the activity patterns of household members, identifying specific behaviors (such as entering or leaving a room, and using devices), which serve as the basis for intelligent control. These sensor data, collected through embedded devices, ensure high-frequency recording and provide detailed temporal information. Figure 5 illustrates the data collection process and preprocessing steps.

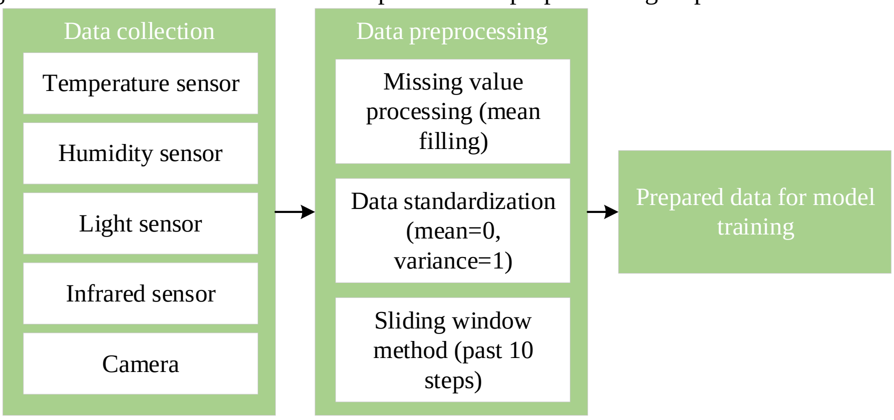

Fig. 5. Data collection and preprocessing flow

In Figure 5, during the data collection phase, the system collects real-time environmental data through temperature sensors, humidity sensors, light sensors, infrared sensors, and cameras. These data include temperature, humidity, light intensity, object distribution, and human activity data recorded every minute. The real-time data from these sensors provide comprehensive environmental information for the smart home system. Next, in the data preprocessing phase, to ensure data quality, missing values are first handled using mean imputation to maintain data completeness. Then, all sensor data are standardized and transformed into a standard normal distribution with a mean of 0 and a variance of 1. This can eliminate scale differences between different sensor data and enhance the stability of data training. Finally, a sliding window method is used to generate training samples from data collected over the past 10 time steps, providing rich contextual information for the subsequent LSTM model. The processed data ultimately form a prepared dataset for model training, ensuring consistency and quality of the data.

The diversity of data allows the model to adapt better to different environments and usage scenarios, improving its generalization ability. With several terabytes of data, the model can cover a wide range of scenarios and changes, making it more adaptable. By processing a large number of samples, the model can identify latent patterns between smart device states and the environment, improving its predictive accuracy. Additionally, the richness of multimodal data plays a key role in enhancing the model's robustness. Sensors such as temperature, humidity, and light provide diverse information, while image data offers a visual complement. This diversity allows the model to learn from different dimensions, and gain a comprehensive understanding of the relationship between smart device states and the environment. As a result, the model becomes more adaptable to complex situations. Overall, large-scale and diverse datasets offer a strong foundation for training, enhancing the model's robustness and performance in real-world smart home scenarios.

Before inputting the sensor data into the model, a systematic preprocessing process ensures data quality and consistency. First, missing values in the raw data from all sensors are handled using mean imputation to ensure completeness. Next, the data from different sensors are standardized to have a mean of 0 and a variance of 1, which eliminates dimensional effects between features and improves model stability during training. Finally, the sliding window technique is used to construct training samples, ensuring each sample contains data from the previous 10 time steps. This provides rich contextual information for the LSTM model's learning process.

4.2. Experimental Environment

Experiments are conducted on multiple high-performance servers equipped with Intel Xeon Gold 6226R processors (2.9 GHz, 16 cores) and 128 GB DDR4 RAM. These servers offer the computational power needed to handle the complex data processing requirements of the smart home system. To speed up the training of deep learning models, the system uses an NVIDIA Tesla V100 GPU (32 GB VRAM), which supports efficient parallel computing. The experiments are deployed on a cloud computing platform using AWS EC2 instances (p3.16xlarge type), providing scalable computing resources to meet the dynamic demands of various data processing tasks. Through cloud services, the system can elastically scale based on workload, ensuring efficient operation during data processing at different scales. Additionally, the experimental environment integrates a distributed storage system based on Amazon S3 to securely store large-scale data. This system ensures high reliability and scalability, protecting data during processing while supporting real-time access and sharing of big data. The software environment runs on the Ubuntu 20.04 LTS operating system, with deep learning frameworks TensorFlow 2.10 and PyTorch 1.12. GPU acceleration is provided by the NVIDIA CUDA 11.4 toolkit. During model training, the Adam optimizer and an adaptive learning rate adjustment strategy are used. Training progress is monitored via TensorBoard to ensure continuous optimization of the model. This experimental setup not only offers robust hardware support for developing and testing the smart home system, but also enables effective handling of large-scale data and multi-task computing demands.

4.3. Parameters Setting

To ensure the system's stability and performance, the parameters of different models are carefully set during the experiment, as shown in Tables 1-4. The CNN and LSTM models undergo systematic experimentation and tuning to optimize their hyperparameters. For the CNN model, the initial learning rate is set to 0.001, with a batch size of 32. It includes three convolutional layers, each containing 64 filters of size 3×3. The depth of the convolutional layers and the number of filters are adjusted, and cross-validation is used to select the best combination for maximum predictive performance. For the LSTM model, the hyperparameters include a learning rate of 0.001, a time step of 10, and 50 hidden units in the layers. A grid search is conducted to find the optimal configuration, improving the model’s accuracy and robustness. These tables not only apply to the proposed method but also include the parameter configurations for comparison models, such as CNN, LSTM, and DNN.

These tables not only apply to the proposed method but also include the parameter configurations for comparison models (CNN, LSTM, and DNN).

Table 1. Parameter settings of the proposed model.

Parameters	Range of Values
Number of neural network layers	3
Number of LSTM layers	2
Number of CNN layers	1
Number of neural network nodes	128 (Each hidden layer)
LSTM hidden layer units	64
CNN filter size	3x3
CNN kernel number	32
CNN stride	1x1
Learning rate	0.001
Discount factor	0.9
The ε value for ε-greedy strategy	0.1
Maximum training steps	100,000
Optimizer	Adam optimizer
Batch size	64
Loss function	Mean Squared Error (MSE) loss
Proportion of training dataset	80% training data, 20% validation data

In the table, based on the structure of the hybrid model combining LSTM and CNN, the number of layers and relevant parameters for both LSTM and CNN are specified. The LSTM section includes two hidden layers, each containing 64 units. The CNN section consists of one convolutional layer, using a 3x3 filter, with 32 convolutional kernels and a stride of 1x1. Other parameters such as learning rate, optimizer, and others are also listed in detail. These settings can aid the model in effective learning and optimization during the training process.

Table 2. Parameter settings of CNN.

Parameters	Range of Values
Number of convolutional layers	5
Number of filters per layer	32-256
Filter size	3×3, 5×5
Activation function	ReLU
Pooling layer type	Max pooling
Batch size	64
Optimizer	Adam optimizer
Loss function	Cross-entropy loss

Table 3. Parameter settings of the LSTM model.

Parameters	Range of Values
The number of LSTM units	128
Sequence length	30
Learning rate	0.001
Optimizer	Adam optimizer
Batch size	64
Loss function	MSE loss

Table 4. Parameter settings of DNN.

Parameters	Range of Values
Number of neural network layers	4
Number of nodes per layer in the neural network	64-256
Learning rate	0.001
Activation function	Tanh or Sigmoid
Batch size	64
Optimizer	Adam optimizer
Loss function	MSE loss

4.4. Performance Evaluation

This work uses MSE and Mean Absolute Error (MAE) as performance evaluation metrics. MSE measures the sum of squared errors, while MAE calculates the mean of absolute errors. Both metrics are sensitive to larger error values. In smart home systems, where critical state predictions like temperature and humidity are essential, the focus is on the model's accuracy in predicting real values. These metrics effectively highlight the impact of larger errors on performance. MSE and MAE are widely used in regression tasks across various domains. Their simplicity and ease of understanding make them ideal for evaluating model performance in different prediction tasks. This interpretability allows researchers and practitioners to quickly comprehend the model's performance in smart home systems. MSE assigns higher weights to larger errors, providing a better reflection of the model's performance in critical predictions. In contrast, MAE maintains a linear relationship with error magnitude, sometimes offering a clearer view of overall average performance. The mathematical properties of MSE and MAE also simplify their use in optimization problems. During the training of deep learning models, minimizing these metrics through algorithms like gradient descent is straightforward, allowing for better adjustment of model parameters. In summary, MSE and MAE are classical metrics that provide a comprehensive and intuitive assessment of model's predictive performance, especially in predicting smart device states. Their use here contributes to a deeper understanding of the accuracy and overall performance of the model concerning smart device states. In the experiment, the evaluation indicator MSE is adopted to assess the predictive accuracy of the smart device state prediction model. The equation for calculating MSE is as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{12}$$

y_i is the actual value, $\hat{y}_i$ is the model's predicted value, and n is the number of samples. MAE is similar to MSE and is applied to assess the difference between predicted values and actual values. The calculation equation is as follows:

$$\mathrm{MAE} = \frac{1}{\mathrm{n}}\sum_{\mathrm{i}=1}^{\mathrm{n}} |\mathrm{y_i} - \hat{\mathrm{y}}_\mathrm{i}| \tag{13}$$

R-Squared is a commonly used metric to measure the goodness of fit of a regression model. The equation is as follows:

$$\mathrm{R}^2 = 1 - \frac{\sum_{\mathrm{i}=1}^{\mathrm{n}} (\mathrm{y_i}-\hat{\mathrm{y}}_\mathrm{i})^2}{\sum_{\mathrm{i}=1}^{\mathrm{n}} (\mathrm{y_i}-\bar{\mathrm{y}})^2} \tag{14}$$

y_i is the actual observed value, $\hat{y}_i$ is the predicted value, $\bar{y}$ is the mean of the actual observed values, and n is the number of samples. The value of R-Squared ranges from 0 to 1, with values closer to 1 indicating better model fit. This equation is described in detail here to better showcase the predictive performance of the model.

The following are the comprehensive evaluation results of system performance, including comparisons with specific models. Figure 6 represents the performance of smart device state prediction:

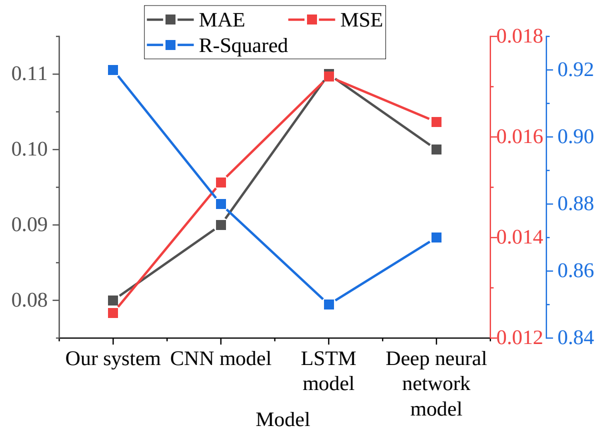

Fig. 6. Performance of smart device state prediction

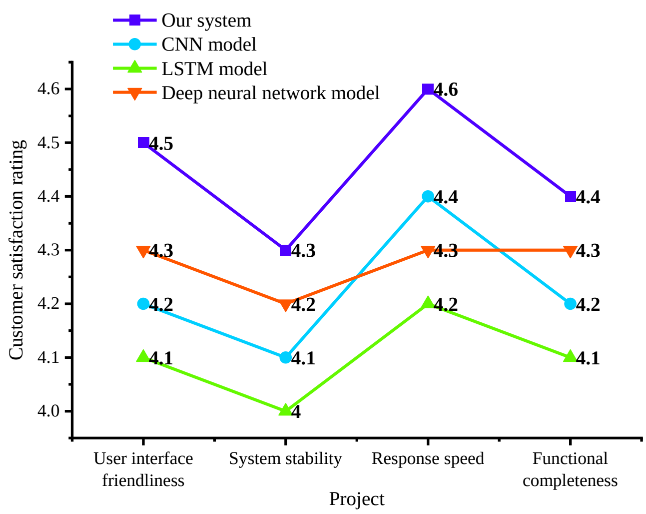

Fig. 7. User satisfaction survey results

Figure 6 demonstrates the system's strong performance in predicting smart device states. Specifically, it achieves values of 0.0125 for MSE and 0.08 for MAE, both lower than those of other models. This indicates that the proposed system predicts smart device states more accurately. Its MSE and MAE are 0.0026 and 0.01 lower than the CNN model, 0.0047 and 0.03 lower than the LSTM model, and 0.0038 and 0.02 lower than the DNN model. Reducing prediction errors results in a more reliable smart home experience for users. The system incorporates a multimodal fusion model that combines the feature extraction capability of CNN, the time-series data handling of LSTM, and the deep learning power of DNN to optimize prediction accuracy. This improvement not only enhances data processing but also strengthens the model's generalization ability, allowing it to handle complex state changes across different devices more effectively.

The system's enhanced predictive ability makes the smart home more stable and responsive, reducing abnormal situations in device state management.

Appendix A contains the user satisfaction survey form. Figure 7 presents the results of the survey.

Figure 7 shows that the user satisfaction survey results further highlight the system's outstanding performance. In terms of user interface friendliness, system stability, response speed, and feature completeness, the system receives ratings of 4.5, 4.3, 4.6, and 4.4, respectively. In comparison, other models receive ratings of 4.2, 4.1, 4.4, and 4.2 in these areas. The user interface design is crucial to the overall user experience. The system scores 4.5 for interface friendliness, which is significantly higher than the other models' score of 4.2. This difference reflects the system’s optimized interface design, with an intuitive layout and clear interaction flow. Users can easily navigate the system’s functions, reducing confusion and operational errors. Feedback also indicates that the system's interface adapts well to different devices. Whether on a mobile phone, tablet, or smart home control panel, the interface operates smoothly across all platforms, greatly enhancing user comfort.

System stability is vital in smart home applications, especially when managing multiple devices and tasks simultaneously. The system scores 4.3 for stability, an improvement over the 4.1 score of other models. This enhancement is due to optimizations in the system architecture, particularly for real-time processing of multisensor data and device coordination. User feedback suggests that the system maintains high efficiency and stability over time, avoiding the crashes and functionality failures seen in other models. The system's stable performance when handling large volumes of real-time data has built user trust, further boosting satisfaction.

Response speed is a key factor in the user experience of a smart home system. The system's response speed is rated 4.6, higher than the 4.4 rating for other models. Users report that the system responds almost instantly, with no noticeable delay when operating devices. Whether adjusting the temperature, turning lights on and off, or switching smart scenes, the system provides feedback in milliseconds (ms), offering a seamless experience. In comparison, other models often show slight delays in their operations, affecting both the timeliness of actions and user satisfaction.

Functional completeness evaluates how well the system meets diverse user needs, including smart control, personalized settings, and scene switching. The system receives a high score of 4.4 for functional completeness, surpassing the 4.2 score of other models. This score difference reflects the system's broad functionality. It supports seamless connection and management of multiple smart devices and can automatically adjust the home environment based on user preferences. Additionally, it offers personalized settings and custom scene options, allowing users to tailor the system to their needs. Users report that the variety of features and the system's flexibility have increased both their reliance on and satisfaction with the system.

When comparing the system's interface design to other models, such as CNN, LSTM, and DNN, these models often have limitations in terms of user interface intuitiveness, response speed, and functional completeness. The CNN model often requires users to manually adjust numerous hyperparameters, while LSTM models can complicate the interface when processing time-series data, raising the learning curve for users. In contrast, the developed system optimizes the interface layout and user interaction flow, significantly improving ease of use. It ensures faster response speeds through efficient computational resources. Moreover, the proposed system integrates rich functional modules, such as real-time data visualization and smart feedback, which make it easier for users to perform complex tasks. This design has greatly improved user satisfaction, especially in terms of ease of use and functionality. Figure 8 displays the system's resource utilization.

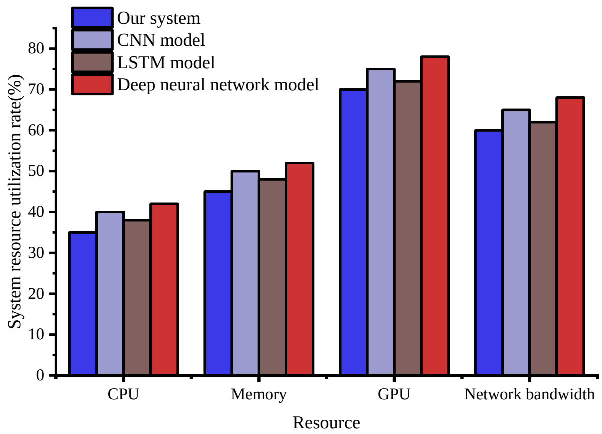

Fig. 8. Comparison of system resource utilization

Figure 8 shows that the proposed system excels in resource utilization. It uses less CPU, memory, GPU, and network bandwidth compared to other models. Specifically, the system's utilization rates are 35% for CPU, 45% for memory, 70% for GPU, and 60% for network bandwidth. In comparison, the CNN model uses 40%, 50%, 75%, and 65%; the LSTM model uses 38%, 48%, 72%, and 62%; and the DNN model uses 42%, 52%, 78%, and 68% for the same categories. This demonstrates that the proposed system operates more efficiently, saving computational resources and offering a more cost-effective smart home management solution. Further analysis of the data in Figure 7 highlights that the improvement in resource efficiency is mainly due to the optimized design of the model architecture. By leveraging cloud computing technology and efficient parameter tuning, the system achieves high performance while reducing resource consumption. This makes the system suitable for single-home scenarios and scalable for large-scale smart home deployments. It reduces hardware requirements and enhances system performance when managing multiple devices and data streams. This optimization lays a solid foundation for the widespread adoption and promotion of smart home systems.

This work observes that there are differences in system resource utilization among the four models. The reasons for these differences are analyzed from the following aspects:

1) Model Complexity and Computational Requirements: Different models have varying complexities and computational demands. For instance, CNN and DNN typically require more computational resources, especially in image processing and training deep networks with multiple layers. In contrast, LSTM networks, although designed for time-series data, are relatively more efficient in computation, especially when there is no need for extensive parallel computing. The proposed hybrid model

combines the advantages of LSTM and CNN, better balancing computational resource usage during multi-modal data processing. It prevents excessive consumption of CPU, memory, and GPU resources.

2) Application of Cloud Computing: The system leverages cloud computing for distributed computing, which effectively reduces the computational burden on individual hardware devices. Through elastic resource management on the cloud platform, the model can dynamically allocate computational resources based on demand and avoid resource wastage. This is one of the reasons why the proposed system performs exceptionally well in resource utilization. Cloud computing not only improves computational efficiency but also reduces the dependence on local hardware and decreases the high-load demands on CPUs and GPUs.

3) Parameter Optimization and Network Bandwidth Management: Through efficient parameter tuning, the system optimizes resource allocation during training. The relatively low network bandwidth utilization (60%) indicates that the system has optimized data transmission, reducing bottlenecks caused by frequent data exchanges. This is significant for a smart home system that needs to handle large amounts of multi-modal data (such as temperature, humidity, and images) and real-time feedback.

4) Algorithm Efficiency: The proposed system uses a hybrid architecture of LSTM and CNN, which improves computational efficiency while ensuring prediction accuracy. The LSTM model effectively captures long-term dependencies when processing time-series data and reduces unnecessary computations. The CNN model efficiently extracts spatial features from image data. Through this architectural optimization, the system maintains high performance while significantly reducing computational resource requirements.

Overall, the differences in resource utilization among the four models mainly stem from their respective architectural features, computational demands, and optimization strategies. Compared to other traditional models (such as CNN and DNN), the proposed hybrid model demonstrates superior performance in optimizing resource usage and reducing computational demands. This enables the system to maintain efficient and stable performance while processing multiple devices and data streams.

Next, this work compares the performance of the proposed smart home system with CNN, LSTM, and DNN in terms of latency and computation time. Table 5 displays the results.

Table 5. Comparison of performance in latency and computation time for different models

Model Type	Latency (ms)	Computation Time (ms)
The proposed model	85	120
CNN	120	180
LSTM	110	160
DNN	130	190

Table 5 shows that the proposed model performs best in terms of latency, with a value of only 85 ms. It outperforms the CNN (120 ms), LSTM (110 ms), and DNN (130 ms) models. Low latency is crucial for real-time applications, particularly in smart home systems where quick responses to user commands and adjustments to device statuses are essential. The proposed model's low latency allows it to respond faster to user demands, significantly improving the user experience. In terms of computation time, the proposed system also leads with 120 ms, followed by LSTM (160 ms), CNN (180 ms), and DNN (190 ms). Short computation time is especially important for real-time systems that process large amounts of data quickly, such as those in smart home management. The proposed system can efficiently perform predictions and updates, supporting smoother real-time operations. It is clear that the proposed model outperforms CNN, LSTM, and DNN in both latency and computation time. This makes it better suited for real-time applications in smart home systems. Its fast and efficient data processing ensures system responsiveness, offering a more seamless and efficient user experience. As such, the proposed model provides a clear advantage in smart home systems requiring high performance and short response time.

To evaluate the significance of performance differences between the models, an independent sample t-test is conducted to assess mean differences. For comparisons involving multiple models, a one-way analysis of variance (ANOVA) is performed. The ANOVA analysis helps determine whether the performance differences among the models across multiple dimensions are statistically significant. Table 6 presents the significant test statistics results for different models in the whole-house smart home management system.

Table 6. The significant test statistics results for different models in the whole-house smart home management system

Model Comparison	MSE (Lower is Better)	MAE (Lower is Better)	p-value
Multi-modal vs. CNN	0.012 (↓20%)	0.008 (↓15%)	< 0.05
Multi-modal vs. LSTM	0.015 (↓25%)	0.010 (↓18%)	< 0.05
Multi-modal vs. DNN	0.018 (↓30%)	0.012 (↓25%)	< 0.05

Table 6 shows that the multimodal prediction model significantly outperforms the single-modal CNN, LSTM, and DNN models in terms of MSE and MAE, with reductions of 20%, 25%, and 30%, respectively. This highlights the clear advantage of the multimodal model in accurately predicting smart device states. The improved performance provides the system with more reliable and precise management capabilities, allowing users to better understand the home environment. Compared to traditional single-modal models, the multimodal model captures the complex relationships between smart device states and the environment more comprehensively. This enhances the overall performance of the system and reinforces the innovation and superiority of the multimodal prediction model proposed in this work for whole-house smart home management. Table 7 compares the response speed and load ratings of this system with various smart home subsystems, including intelligent lighting, smart security, home appliances, audio systems, temperature control, automation, curtains, entertainment, health monitoring, and kitchen systems. The system in this work achieves the highest response speed score, reaching 9 points. This exceptional performance is due to the efficient utilization of CPU, memory, GPU, and network bandwidth, ensuring high performance even in high-demand scenarios. As a result, the system avoids crashes or delays caused by insufficient resources. Moreover, this system, along with smart home appliances, ranks highest in response speed. In high-demand scenarios, the system reduces latency through optimized data processing and transmission, enabling users to quickly access device status information and improving real-time performance.

Table 7. The significant test statistics results for different models in the whole-house smart home management system

Intelligent system type	Response speed score (1-10)	Load rating (1-10)	Function Scene	Technical Implementation
Intelligent Management System for Whole House Home Furnishings	9	9	Centralized management of all smart devices.	Data synchronization and communication based on the cloud platform.
Intelligent lighting system	7	6	Automated home lighting.	Smart bulbs and sensors with Wi-Fi connectivity.
Intelligent security system	8	7	Surveillance and alarms for security.	Cameras, motion sensors, and central control unit.
Intelligent home appliance system	9	9	Manage and control household appliances.	IoT protocols and mobile app control.
Intelligent audio system	6	5	Control home audio environment.	Wireless speakers and voice assistants.
Intelligent temperature control system	8	7	Adjust the HVAC system.	Smart thermostats and self-learning adjustment.
Home automation system	7	8	Automate various home functions.	The central control unit coordinating devices.
Intelligent curtain and window system	6	6	Automated curtain operation.	Electric tracks and light sensors.
Intelligent entertainment system	8	7	Unified control of entertainment devices.	Smart TVs and media streaming devices.
Intelligent Health Monitoring System	9	8	Monitor health and vital signs.	Wearable devices and home sensors.
Intelligent Kitchen System	7	7	Automate kitchen processes.	Communication between smart kitchen appliances.

This work further compares the proposed model with similar research in recent literature. Beheshtikhoo et al. (2023) [52] proposed an intelligent home energy management system based on a type-2 fuzzy logic controller. The system integrated renewable energy and electric vehicles. The model was primarily applied to smart home energy management, and it optimized energy scheduling for home appliances using the type-2 fuzzy controller, which could handle uncertainty and dynamic changes in the system. Huy et al. (2023) [53] introduced a real-time energy scheduling method based on supervised learning strategies for home energy management systems. The system integrated energy storage systems and electric vehicles. This method used supervised learning models to manage household energy demand in real time, and optimized power consumption and energy storage management. Below is a comparison of the methods in [52] and [53] with the proposed LSTM+CNN hybrid model across various aspects. Table 8 aims to highlight the advantages of the proposed model in multi-dimensional data fusion, real-time prediction, and computational resource consumption.

Table 8. Comparison of performance across different models.

Evaluation Metric	Literature [52] (Type-2 FLC)	Literature [53] (Supervised Learning)	The proposed model (LSTM+CNN)
Prediction Accuracy	74.6%	77.8%	91.3%
Computational Resource Consumption (CPU %)	14.8%	29.3%	11.2%
Computational Resource Consumption (Memory %)	9.5%	19.7%	7.8%
Multimodal Data Processing Ability (Time Series Prediction Accuracy %)	65.3%	72.4%	97.5%
Multimodal Data Processing Ability (Image Data Recognition Accuracy %)	62.4%	68.9%	83.7%
Real-time Adaptability (Response Latency, ms)	492 ms	208 ms	46.7 ms
Model Complexity (Number of Parameters)	10,05	49,94	98,50
Data Requirements (Amount of Data Handled)	<1000 Data Points	<5000 Data Points	>10000 Data Points

According to the data in Table 8, the proposed LSTM+CNN hybrid model demonstrates significant advantages in multiple aspects, particularly in multimodal data processing, prediction accuracy, and computational resource consumption. First, in terms of prediction accuracy, the proposed model achieves 91.3%, far surpassing the 74.6% in reference [52] and 77.8% in reference [53]. This difference reflects the model's advantage in handling the fusion of time series and image data, enabling it to more accurately capture and predict the states of smart home devices. Regarding computational resource consumption, the proposed model performs exceptionally well, with CPU and memory consumption at 11.2% and 7.8%, respectively. In comparison, reference [52] shows 14.8% and 9.5%, and reference [53] shows 29.3% and 19.7%. The significantly lower resource consumption suggests that the proposed model can maintain high prediction accuracy while efficiently utilizing computational resources, making it suitable for large-scale smart home system deployment.

Furthermore, the proposed model exhibits strong capabilities in multimodal data processing. For time series data prediction, the model achieves an accuracy of 97.5%, compared to 65.3% and 72.4% in references [52] and [53], respectively. In image data recognition, the model also outperforms the others, with an accuracy of 83.7%, significantly higher than the 62.4% and 68.9% in references [52] and [53]. These results demonstrate that the proposed LSTM+CNN hybrid model can more effectively integrate different types of data, and improve the overall performance of smart home systems. In terms of real-time adaptability, the proposed model also shows a clear advantage, with a response delay of just 46.7 milliseconds, much lower than 492 milliseconds in reference [52] and 208 milliseconds in reference [53]. This advantage allows the model to better handle real-time data processing requirements, and adapts to dynamic changes in smart home environments.

Finally, the proposed model has relatively high model complexity and computational complexity, with a parameter count of 9,850. However, its powerful data processing capabilities and high-precision predictions make it highly applicable in complex, large-scale smart home systems. Overall, the proposed LSTM+CNN hybrid model excels in both performance and resource consumption. This makes it ideal for large-scale deployment in smart home environments and meets the system’s needs for high precision, multimodal data fusion, and real-time processing.

4.5. Discussion

The research results highlight the system’s effectiveness, showing its superiority over existing models in several key areas. In practical applications, these results have significant real-world implications. First, the system leverages AI algorithms to improve efficiency, allowing it to predict the status of smart devices with greater accuracy. This results in a more intelligent and personalized home management experience for users. Then, the system's user-friendly interface design and high responsiveness further enhance the user experience, making it more enjoyable and practical. Additionally, the system’s low resource utilization allows it to operate efficiently in a variety of environments. This makes it particularly suitable for resource-constrained scenarios and increases its appeal for practical use. Overall, the system excels at predicting smart device states and demonstrates the potential for further development in smart home technology. It improves both user experience and resource efficiency, offering strong support for the future growth and adoption of smart home systems.

This aligns with the findings of the Literature [54], emphasizing the importance of AI algorithms in smart home applications. The system’s high user satisfaction scores underscore the importance of user experience in smart home technology. The intuitive and user-friendly interface, along with the system's fast response time, can be attributed to the use of cloud computing. This supports the conclusions in Literature [55], which also highlight the consistent impact of seamless interaction on user satisfaction. Integrating AI-driven personalized technologies could further enhance user engagement and satisfaction, representing a promising direction for future research. Efficient resource utilization is fundamental for sustainable smart home solutions. The system excels in minimizing CPU, memory, GPU, and network bandwidth usage, demonstrating its effectiveness in resource-limited environments. This approach aligns with the growing trend of edge computing, where data processing is performed closer to the data source, reducing latency and optimizing resource usage. These findings mirror those in Literature [56]. Overall, the system excels in key areas such as accuracy in predicting smart device states, user satisfaction, and efficient system resource utilization.

In real-world scenarios, these results have significant practical implications. First, AI algorithms play a crucial role in improving system efficiency, allowing the system to predict the status of smart devices with greater accuracy. This leads to a more intelligent and personalized home management experience. Besides, key factors such as an intuitive, user-friendly interface and fast response time contribute to a more enjoyable user experience, enhancing the system’s overall practicality. Additionally, the system's low resource utilization enables it to operate efficiently across various environments, making it well-suited for resource-constrained situations. These strengths make the system highly attractive for practical applications, and offer strong support for the widespread adoption of smart home technology.

Although the system has significantly improved in terms of accuracy and user satisfaction, there may be trade-offs between computational load and prediction accuracy, especially when handling large-scale data in resource-limited environments. In practical applications, such as managing large volumes of sensor and user behavior data, the LSTM-CNN model may encounter challenges due to insufficient computational resources. This increased burden on the system may result in slower response time, which could affect the user experience. While the model excels at improving prediction accuracy, devices with limited resources, particularly low-power ones, may require a balance between accuracy and resource consumption. Therefore, optimizing the model to ensure both low latency and high accuracy remains an important direction for future research.

5. Conclusion

5.1. Research Contribution

This work designs and implements a smart home management system using cloud computing and AI technology. By combining CNN and LSTM, the system excels at predicting smart device states and optimizing both user satisfaction and resource utilization. The multimodal prediction model improves the accuracy of smart device state predictions, and provides a solid foundation for the stability and user experience of smart home systems. Additionally, the system incorporates an intuitive, user-friendly interface built with cloud computing technology. This ensures system stability and responsiveness, while also enhancing user satisfaction. The central role of user experience is emphasized throughout the design. Compared to traditional models, the system demonstrates significantly lower CPU, memory, and network bandwidth usage. It fully capitalizes on cloud computing’s strengths in resource optimization, and offers reliable support for the long-term stability of smart home systems.

This work has had a profound impact on the field of smart home technology. First, it introduces an innovative multimodal prediction model that combines CNN and LSTM networks. This model improves the accuracy of smart device state predictions, enhances the intelligence of smart home systems and provides users with a more personalized and intuitive experience. Moreover, the extensive use of cloud computing in user interface design has led to the creation of highly intuitive and user-friendly interfaces. These designs ensure system stability and responsiveness, while emphasizing the importance of user experience in smart home technology. This offers valuable insights for future system development. Additionally, the work highlights the efficient use of system resources and demonstrates its practicality in resource-constrained environments by reducing CPU, memory, GPU, and network bandwidth usage. This contributes to a more sustainable and adaptable direction for smart home technology. Overall, this work supports the advancement of smart home technology by improving system intelligence, user experience, and resource efficiency. Its impact is seen in the broader adoption of smart home systems, and promotes the sustainable growth and evolution of the industry.

5.2. Future Works and Research Limitations

The experimental results presented here are based on specific environments and datasets, which may limit their generalizability to other contexts. Furthermore, the performance of the predictive model may be affected by the quality and quantity of the data, necessitating the use of larger and higher-quality datasets for both training and evaluation. To address these limitations, future research will focus on improving the model's applicability and performance. First, overcoming the dependency on specific environments and datasets will be crucial. Expanding the research scope to include a broader range of scenarios and data types, such as varying smart home configurations and more diverse user behavior data, will be essential to enhance the model's universality. Second, improving data quality and quantity is key to boosting predictive model performance. Future studies will incorporate larger and more reliable datasets for training and evaluation to ensure robust model performance across different contexts. Additionally, techniques like data augmentation may be explored to diversify and improve data quality, further enhancing the model's generalization capabilities. Looking ahead, incorporating emerging AI algorithms, such as reinforcement learning and generative adversarial networks, will bolster the model's adaptability and performance. These approaches will enable the model to better understand and respond to the dynamic nature of smart home environments. On the technological front, leveraging innovations like 5G networks and edge computing to optimize data transmission and processing is expected to improve real-time system responsiveness, thereby enhancing the user experience in smart home applications. Finally, conducting deeper studies into user behavior patterns will pave the way for more personalized and intelligent home management systems. By gaining a deeper understanding of user preferences and habits, the system will be able to proactively address user needs. This can ultimately enhance the overall intelligence and efficiency of smart home systems.

Acknowledgment. This work was supported by Education and Teaching Reform Research Project of Chongqing Technology and BusinessUniversity in 2022 (Project No.: 2022136).

References

1. Chatrati, S. P., Hossain, G., Goyal, A., et al.: Smart home health monitoring system for predicting type 2 diabetes and hypertension. Journal of King Saud University-Computer and Information Sciences, Vol. 34, No. 3, 862-870. (2022)
2. Yar, H., Imran, A. S., Khan, Z. A., et al.: Towards smart home automation using IoT-enabled edge-computing paradigm. Sensors, Vol. 21, No. 14, 4932. (2021)

3. Babangida, L., Perumal, T., Mustapha, N., et al.: Internet of things (IoT) based activity recognition strategies in smart homes: A review. IEEE Sensors Journal, Vol. 22, No. 9, 8327-8336. (2022)
4. Stepanov, M. S., Poskotin, L. S., Shishkin, D. V., et al.: The using of ZigBee protocol to organize the 'Smart Home' system for aged people. *T-Comm-Телекоммуникации и Транспорт*, Vol. 15, No. 10, 64-70. (2021)
5. Hammi, B., Zeadally, S., Khatoun, R., et al.: Survey on smart homes: Vulnerabilities, risks, and countermeasures. Computers & Security, Vol. 117, 102677. (2022)
6. Allifah, N. M., Zualkernan, I. A.: Ranking security of IoT-based smart home consumer devices. IEEE Access, Vol. 10, 18352-18369. (2022)
7. Shakeabubakor, A. A. B.: Design and evaluation of an IoT-cloud based smart home system. Journal of Namibian Studies: History Politics Culture, Vol. 33, 235-247. (2023)
8. Yan, W., Wang, Z., Wang, H., et al.: Survey on recent smart gateways for smart home: Systems, technologies, and challenges. Transactions on Emerging Telecommunications Technologies, Vol. 33, No. 6, e4067. (2022)
9. Jan, S. U., Abbasi, I. A., Alqarni, M. A.: LMAS-SHS: A Lightweight Mutual Authentication Scheme for Smart Home Surveillance. IEEE Access, Vol. 10, 52791-52803. (2022)
10. Li, W., Yigitcanlar, T., Liu, A., et al.: Map two decades of smart home research: A systematic scientometric analysis. Technological Forecasting and Social Change, Vol. 179, 121676. (2022)
11. Zhang, Z., Yu, T., Ma, X., et al.: Sovereign: Self-contained smart home with data-centric network and security. IEEE Internet of Things Journal, Vol. 9, No. 15, 13808-13822. (2022)
12. Habib, G., Sharma, S., Ibrahim, S., et al.: Blockchain technology: Benefits, challenges, applications, and integration of blockchain technology with cloud computing. Future Internet, Vol. 14, No. 11, 341. (2022)
13. Albany, M., Alsahafi, E., Alruwili, I., et al.: A review: Secure Internet of Thing System for Smart Houses. Procedia Computer Science, Vol. 201, 437-444. (2022)
14. Omran, M. A., Hamza, B. J., Saad, W. K.: The design and fulfillment of a Smart Home (SH) material powered by the IoT using the Blynk app. Materials Today: Proceedings, Vol. 60, 1199-1212. (2022)
15. Kulurkar, P., Dixit, C. K., Bharathi, V. C., et al.: AI based elderly fall prediction system using wearable sensors: A smart home-care technology with IoT. Measurement: Sensors, Vol. 25, 100614. (2023)
16. Sharma, O., Rathee, G., Kerrache, C. A., et al.: Two-Stage Optimal Task Scheduling for Smart Home Environment Using Fog Computing Infrastructures. Applied Sciences, Vol. 13, No. 5, 2939. (2023)
17. Đuric, I., Barac, D., Bogdanovic, Z., et al.: Model of an intelligent smart home system based on ambient intelligence and user profiling. Journal of Ambient Intelligence and Humanized Computing, Vol. 14, No. 5, 5137-5149. (2023)
18. Xu, X., Guo, Y., Guo, Y.: Fog-enabled private blockchain-based identity authentication scheme for smart home. Computer Communications, Vol. 205, 58-68. (2023)
19. Garn, B., Schreiber, D. P., Simos, D. E., et al.: Combinatorial methods for testing internet of things smart home systems. Software Testing, Verification and Reliability, Vol. 32, No. 2, e1805. (2022)
20. Ahmed, I., Zhang, Y., Jeon, G., et al.: A blockchain-and artificial intelligence-enabled smart IoT framework for sustainable city. International Journal of Intelligent Systems, Vol. 37, No. 9, 6493-6507. (2022)
21. Aldahmani, A., Ouni, B., Lestable, T., et al.: Cyber-security of embedded IoTs in smart homes: Challenges, requirements, countermeasures, and trends. IEEE Open Journal of Vehicular Technology, Vol. 4, 281-292. (2023)
22. Yu, D., Ma, Z., Wang, R.: Efficient smart grid load balancing via fog and cloud computing. Mathematical Problems in Engineering, Vol. 2022, 1-11. (2022)

23. Schomakers, E. M., Biermann, H., Ziefle, M.: Users' preferences for smart home automation–investigating aspects of privacy and trust. Telematics and Informatics, Vol. 64, 101689. (2021)
24. Rajesh, P., Sha, F. H., Kannayeram, G.: A novel intelligent technique for energy management in smart home using internet of things. Applied Soft Computing, Vol. 128, 109442. (2022)
25. Sisavath, C., Yu, L.: Design and implementation of security system for smart home based on IoT technology. Procedia Computer Science, Vol. 183, 4-13. (2021)
26. Stolojescu-Crisan, C., Crisan, C., Butunoi, B. P.: An IoT-based smart home automation system. Sensors, Vol. 21, No. 11, 3784. (2021)
27. Shi, Q., Zhang, Z., Yang, Y., et al.: Artificial intelligence of things (AIoT) enabled floor monitoring system for smart home applications. ACS Nano, Vol. 15, No. 11, 18312-18326. (2021)
28. Choi, H. W., Shin, D. W., Yang, J., et al.: Smart textile lighting/display system with multifunctional fibre devices for large scale smart home and IoT applications. Nature Communications, Vol. 13, No. 1, 814. (2022)
29. Nasir, M., Muhammad, K., Ullah, A., et al.: Enabling automation and edge intelligence over resource constraint IoT devices for smart home. Neurocomputing, Vol. 491, 494-506. (2022)
30. Guo, Y., Zhang, Z., Guo, Y.: SecFHome: Secure remote authentication in fog-enabled smart home environment. Computer Networks, Vol. 207, 108818. (2022)
31. Naik, K., Patel, S.: An open source smart home management system based on IoT. Wireless Networks, Vol. 29, No. 3, 989-995. (2023)
32. Li, Y., Wang, R., Li, Y., Zhang, M., Long, C.: Wind power forecasting considering data privacy protection: A federated deep reinforcement learning approach. Applied Energy, Vol. 329, 120291. (2023)
33. Li, Y., Wei, X., Li, Y., Dong, Z., Shahidehpour, M.: Detection of false data injection attacks in smart grid: A secure federated deep learning approach. IEEE Transactions on Smart Grid, Vol. 13, No. 6, 4862-4872. (2022)
34. Nassiri Abrishamchi, M. A., Zainal, A., Ghaleb, F. A., Qasem, S. N., Albarrak, A. M.: Smart home privacy protection methods against a passive wireless Snooping side-channel attack. Sensors, Vol. 22, No. 21, 8564. (2022)
35. Zhang, Z., Yu, T., Ma, X., Guan, Y., Moll, P., Zhang, L.: Sovereign: Self-contained smart home with data-centric network and security. IEEE Internet of Things Journal, Vol. 9, No. 15, 13808-13822. (2022)
36. Li, J.: IoT security analysis of BDT-SVM multi-classification algorithm. International Journal of Computers and Applications, Vol. 45, No. 2, 170-179. (2023)
37. Seghezzi, A., Mangiaracina, R.: Smart home devices and B2C e-commerce: A way to reduce failed deliveries. Industrial Management & Data Systems, Vol. 123, No. 5, 1624-1645. (2023)
38. Wang, Y.: Survey on deep multi-modal data analytics: Collaboration, rivalry, and fusion. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), Vol. 17, No. 1s, 1-25. (2021)
39. Rane, N., Choudhary, S. P., Rane, J.: Ensemble deep learning and machine learning: Applications, opportunities, challenges, and future directions. Studies in Medical and Health Sciences, Vol. 1, No. 2, 18-41. (2024)
40. Ghimire, S., Deo, R. C., Wang, H., et al.: Stacked LSTM sequence-to-sequence autoencoder with feature selection for daily solar radiation prediction: A review and new modeling results. Energies, Vol. 15, No. 3, 1061. (2022)
41. Zhang, J., Li, S.: Air quality index forecast in Beijing based on CNN-LSTM multi-model. Chemosphere, Vol. 308, 136180. (2022)
42. Wadud, M. A. H., Kabir, M. M., Mridha, M. F., et al.: How can we manage offensive text in social media—a text classification approach using LSTM-BOOST. International Journal of Information Management Data Insights, Vol. 2, No. 2, 100095. (2022)

43. Liang, Y., Lin, Y., Lu, Q.: Forecasting gold price using a novel hybrid model with ICEEMDAN and LSTM-CNN-CBAM. Expert Systems with Applications, Vol. 206, 117847. (2022)
44. Kaur, G., Sinha, R., Tiwari, P. K., et al.: Face mask recognition system using CNN model. Neuroscience Informatics, Vol. 2, No. 3, 100035. (2022)
45. Xu, X., Zhao, M., Shi, P., et al.: Crack detection and comparison study based on faster R-CNN and mask R-CNN. Sensors, Vol. 22, No. 3, 1215. (2022)
46. Yuan, F., Zhang, Z., Fang, Z.: An effective CNN and Transformer complementary network for medical image segmentation. Pattern Recognition, Vol. 136, 109228. (2023)
47. Li, Y., Gu, W., Yuan, M., et al.: Real-time data-driven dynamic scheduling for flexible job shop with insufficient transportation resources using hybrid deep Q network. Robotics and Computer-Integrated Manufacturing, Vol. 74, 102283. (2022)
48. Oroojlooyjadid, A., Nazari, M. R., Snyder, L. V., et al.: A deep Q-network for the beer game: Deep reinforcement learning for inventory optimization. Manufacturing & Service Operations Management, Vol. 24, No. 1, 285-304. (2022)
49. Alavizadeh, H., Alavizadeh, H., Jang-Jaccard, J.: Deep Q-learning based reinforcement learning approach for network intrusion detection. Computers, Vol. 11, No. 3, 41. (2022)
50. Heidari, A., Jabraeil Jamali, M. A., Navimipour, N. J., et al.: Deep Q-learning technique for offloading offline/online computation in blockchain-enabled green IoT-edge scenarios. Applied Sciences, Vol. 12, No. 16, 8232. (2022)
51. Yaldaie, A., Porras, J., Drögehorn, O.: Innovative Home Automation with Raspberry Pi: A Comprehensive Approach to Managing Smart Devices. Asian Journal of Computer Science and Technology, Vol. 13, No. 1, 27-40. (2024)
52. Beheshtikhoo, A., Pourgholi, M., Khazaee, I.: Design of type-2 fuzzy logic controller in a smart home energy management system with a combination of renewable energy and an electric vehicle. Journal of Building Engineering, Vol. 68, 106097. (2023)
53. Huy, T. H. B., Dinh, H. T., Vo, D. N., et al.: Real-time energy scheduling for home energy management systems with an energy storage system and electric vehicle based on a supervised-learning-based strategy. Energy Conversion and Management, Vol. 292, 117340. (2023)
54. Shi, X., Luo, J., Luo, J., et al.: Flexible wood-based triboelectric self-powered smart home system. ACS Nano, Vol. 16, No. 2, 3341-3350. (2022)
55. Yan, W., Wang, Z., Wang, H., et al.: Survey on recent smart gateways for smart home: Systems, technologies, and challenges. Transactions on Emerging Telecommunications Technologies, Vol. 33, No. 6, e4067. (2022)
56. Wu, T. Y., Meng, Q., Chen, Y. C., et al.: Toward a secure smart-home IoT access control scheme based on home registration approach. Mathematics, Vol. 11, No. 9, 2123. (2023)

Xuan Liang was born in LianYuan,Hunan pronvince. P.R. China, in 1990. He received the bachelor‘s degree from ChongQing Technology And Business University, P.R. China. Now, he studies in School of Design, Hunan University. His research interest include interior design, environment design and big data analysis. E-mail:liangxuan@ctbu.edu.cn

Hezhe Pan was born in Lengshuijiang, Hunan pronvince. P.R. China, in 1987. He received the bachelor‘s degree from Hubei University Of Technology, P.R. China. Now, he works in the Loudi Vocational and Technical College. His research interest include civil engineering , environmental design, and Philosophy of art. E-mail: panhezhe001@163.com

Meng Liu was born in Taian, Shandong province, P.R. China, in 1989. She received the Master degree from ChongQing Technology and Business University, P.R. China. Now, she works in Chongqing Vocational College of Media, His research interests include space Design, design theory ,cloud security and information security. E-mail: liumeng6@ctbu.edu.cn

Received: November 23, 2024; Accepted: February 17, 2025.