Heliyon 10 (2024) e37229

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Research article

Application of GWO-attention-ConvLSTM model in customer churn prediction and satisfaction analysis in customer relationship management ☆

Hui Zhang [a,*], Weihua Zhang [b]

[a] Department of Management, Zhengzhou Business University, 451200, Zhengzhou, Henan, China
[b] School of Information and Mechanical and Electrical Engineering, Zhengzhou Business University, 451200, Zhengzhou, Henan, China

ARTICLE INFO

Keywords:
Customer Relationship Management (CRM)
Churn prediction
GWO-attention-ConvLSTM model
Time series
Customer satisfaction analysis

ABSTRACT

Customer Relationship Management (CRM) is vital in modern business, aiding in the management and analysis of customer interactions. However, existing methods struggle to capture the dynamic and complex nature of customer relationships, as traditional approaches fail to leverage time series data effectively. To address this, we propose a novel GWO-attention-ConvLSTM model, which offers more effective prediction of customer churn and analysis of customer satisfaction. This model utilizes an attention mechanism to focus on key information and integrates a ConvLSTM layer to capture spatiotemporal features, effectively modeling complex temporal patterns in customer data. We validate our proposed model on multiple real-world datasets, including the BigML Telco Churn dataset, IBM Telco dataset, Cell2Cell dataset, and Orange Telecom dataset. Experimental results demonstrate significant performance improvements of our model compared to existing baseline models across these datasets. For instance, on the BigML Telco Churn dataset, our model achieves an accuracy of 95.17%, a recall of 93.66%, an F1 score of 92.89%, and an AUC of 95.00%. Similar results are validated on other datasets. In conclusion, our proposed GWO-attention-ConvLSTM model makes significant advancements in the CRM domain, providing powerful tools for predicting customer churn and analyzing customer satisfaction. By addressing the limitations of existing methods and leveraging the capabilities of deep learning, attention mechanisms, and optimization algorithms, our model paves the way for improving customer relationship management practices and driving business success.

1. Introduction

With the increasingly fierce competition across various industries, customer churn has emerged as a pressing challenge in sectors such as banking, telecommunications, music media, and insurance. Customer Relationship Management (CRM) plays a pivotal role in modern enterprises, aiming to establish and maintain long-term relationships with customers to enhance satisfaction, increase market

☆ Hui Zhang, graduated from Zhengzhou University. Currently a Ph.D. candidate, associate professor, working at Zhengzhou Business School. Research directions include management and computer applications. Email: zhang8736598@126.com.

* Corresponding author.
E-mail addresses: zhang8736598@126.com (H. Zhang), zwhchenggong@126.com (W. Zhang).

https://doi.org/10.1016/j.heliyon.2024.e37229
Received 19 April 2024; Received in revised form 28 August 2024; Accepted 29 August 2024

Available online 4 September 2024

share, and boost profits [1]. However, despite the growing significance of CRM, challenges persist in practice. Traditional CRM methods often overly rely on static data analysis and experiential judgment, making it difficult to adapt to dynamic and complex market environments in a timely manner. Customer churn represents another critical issue facing businesses, as it not only leads to potential revenue loss but also adversely affects brand image and market competitiveness [2]. Therefore, accurately predicting customer churn and implementing corresponding measures have become urgent tasks for enterprises. In recent years, with the advancement of deep learning technology, researchers have begun exploring how to leverage deep learning models to address the challenge of customer churn prediction. Deep learning models excel in uncovering hidden information and patterns within large-scale datasets, thereby enhancing the accuracy and effectiveness of customer churn prediction [3]. In the realm of customer churn prediction, time series forecasting techniques play a crucial role. These techniques enable the capture of dynamic changes in customer behavior and trends, thereby improving the accuracy and robustness of churn prediction [4]. Thus, this paper will focus on discussing the research progress of deep learning in customer churn prediction within CRM, with particular emphasis on the significance and application of time series forecasting in predicting churn rates. Through in-depth analysis and discussion, the aim is to provide enterprises with more effective CRM strategies and methods, thereby facilitating sustainable development.

In recent years, multiple deep learning models have been applied to the task of customer churn prediction. These research outcomes reflect both the potential and limitations of deep learning technology in the field of customer relationship management (CRM). One study utilized a Variational Autoencoder (VAE) model to predict customer churn [5]. This research modeled customer data using the VAE model and used the generated data to predict customer churn. However, the training process of the VAE model may encounter difficulties in convergence and challenges in the quality of generated data. Another study attempted to use Generative Adversarial Networks (GAN) model for customer churn prediction [6]. This research utilized synthetic data generated by the GAN model to assist in the task of customer churn prediction. However, due to the instability and convergence difficulties in training GAN models, this approach may face challenges in generating high-quality data. Yet another study employed an Autoencoder (AE) model for customer churn prediction [7]. This research utilized the AE model to encode and decode customer behavioral data to extract latent feature information for predicting customer churn. However, the AE model may be affected by issues such as the selection of encoding dimensions and imbalanced training samples, leading to decreased prediction performance. Finally, a study explored the application of Sequence-to-Sequence (Seq2Seq) models in customer churn prediction [8]. This research modeled customer historical behavior sequences using the Seq2Seq model and utilized an encoder-decoder structure to predict future customer behavior. However, Seq2Seq models may overlook long-term dependencies within sequences, resulting in poor prediction performance. These studies demonstrate the diverse applications of deep learning in customer churn prediction and reveal the strengths and weaknesses of each model. However, future research still needs to address the challenges present in these models to enhance the effectiveness and efficiency of customer relationship management.

Based on the shortcomings identified in the aforementioned works, we propose the GWO-attention-ConvLSTM network. This model consists of three key components: the Grey Wolf Optimization (GWO) algorithm, attention mechanism, and ConvLSTM network. The GWO optimization algorithm is employed to adjust model parameters, the attention mechanism dynamically adjusts the importance weights of input data, and the ConvLSTM network handles the spatiotemporal features of customer behavior sequence data. Together, these three components construct a powerful deep learning model for customer churn prediction and satisfaction analysis. Our model addresses the limitations of existing methods in customer relationship management by introducing the GWO optimization algorithm, attention mechanism, and ConvLSTM network, thereby improving the prediction accuracy and generalization capability of the model. This enhances the adaptability and universality of our model, making it applicable to enterprises across various industries and scales, thereby providing more precise and reliable customer relationship management services. Consequently, it leads to increased customer satisfaction, enhanced customer loyalty, and facilitates business growth.

The contribution points of this paper are as follows:

- We have proposed a novel model that integrates the Grey Wolf Optimization (GWO) algorithm, attention mechanism, and ConvLSTM network to enhance the accuracy and efficiency of customer churn prediction and satisfaction analysis. This model combines various deep learning techniques to comprehensively understand customer behavior patterns, thereby providing enterprises with more precise market insights and personalized services.
- Through empirical research, we have validated the effectiveness of this model and achieved significant results in customer relationship management, providing enterprises with more reliable decision support. This study introduces a new deep learning approach for customer churn prediction and satisfaction analysis, offering a more comprehensive and accurate solution for customer relationship management.
- We have also provided insights into future research directions, highlighting the limitations and improvement opportunities of the current model. Additionally, we have proposed potential research directions to further enhance the effectiveness and applicability of customer relationship management. These prospects offer valuable guidance for future research and practice in deep learning, contributing to the advancement of the field of customer relationship management.

The remainder of this paper is structured as follows. Section 2 elaborates on the recent related work in this field. Section 3 provides a detailed description of our proposed method, including an overview of the ConvLSTM module, Attention, and GWO. Section 4 describes the experimental segment, detailing and comparing different experiments. Finally, Section 5 concludes the paper and outlines future research directions.

2. Related work

2.1. Overview of traditional customer relationship management methods

Traditional customer relationship management methods refer to the approaches that businesses typically used to manage customer relationships before the digital era. These methods primarily relied on manual and traditional communication channels, such as face-to-face sales, telemarketing, and traditional advertising [9]. In this management model, communication and interaction between businesses and customers were mainly conducted through methods like personal visits, phone calls, and email communications, while customer information and transaction records were typically recorded in paper or simple electronic spreadsheets [10]. Additionally, traditional customer relationship management methods often depended on personal experience and intuition. Decision-makers and sales personnel often relied on their own experience and intuition to judge customer needs and market trends, lacking systematic data analysis and decision support [11]. Therefore, in this management model, the marketing activities, sales strategies, and customer service of a business often depended on the capabilities and experience levels of individual employees. Although traditional customer relationship management methods have to some extent met the needs of businesses, they also have limitations. For example, issues such as untimely information, low efficiency, and difficulty in collecting and analyzing customer feedback exist [12]. With the development of information technology and the rise of data-driven decision-making, an increasing number of businesses are turning to digital, automated, and intelligent customer relationship management methods to improve work efficiency, optimize customer experiences, and achieve precision marketing.

2.2. Machine learning-based customer classification and prediction methods

Machine learning-based customer classification and prediction methods are techniques that utilize machine learning algorithms to classify and predict customers. The core idea of this approach is to analyze and model customers using a large amount of customer data through machine learning algorithms to identify different types of customer groups and predict their future behavior and needs [13][14]. In this method, the first step is to collect and organize a large amount of customer data, including basic customer information, transaction records, behavioral data, etc. Then, machine learning algorithms are used to analyze and process this data to extract customer features and patterns. Subsequently, classification algorithms are used to classify customers. Classification algorithms can divide customers into different groups or categories, each representing a specific type of customer with similar features and behavior patterns. Common classification algorithms include decision trees, K-nearest neighbors, support vector machines, etc [15].

In addition to customer classification, machine learning-based methods can also be used for customer prediction. Prediction algorithms can predict future behavior and needs of customers based on their historical data and behavior patterns, such as predicting whether customers will churn, purchase a particular product or service, etc [16]. Common prediction algorithms include regression analysis, time series analysis, neural networks, etc. Through machine learning-based customer classification and prediction methods, businesses can better understand customer needs and behaviors, develop targeted marketing strategies and customer service solutions, improve customer satisfaction and loyalty, thereby achieving business growth and competitive advantage.

2.3. The application of deep learning technology in customer behavior analysis

The application of deep learning technology in customer behavior analysis involves leveraging deep neural networks and other deep learning algorithms and models to uncover underlying patterns and features within customer data. The advantage of this approach lies in its ability to handle large-scale, high-dimensional customer data and extract hidden insights, providing businesses with deeper customer insights [17]. Specifically, deep learning models can learn complex features of the data through multi-layer nonlinear transformations, including customer transaction behaviors, browsing histories, click behaviors, social media activities, and more [18]. By analyzing this data using deep learning techniques, businesses can better understand customer behavior patterns and preferences, enabling personalized product recommendations and services. For example, deep learning models can predict customer purchase behaviors and churn risks by learning from historical customer data, allowing early identification of potential customer churn risks and intervention measures [19].

Moreover, deep learning technology can also be applied to customer sentiment analysis and emotion recognition. By analyzing customer comments and sentiments on social media, deep learning models can identify customer sentiment tendencies and attitudes, thus better understanding customer needs and feedback and providing businesses with more timely and accurate services [20]. Overall, the application of deep learning technology in customer behavior analysis enables businesses to comprehensively and accurately understand customers, deliver personalized products and services, enhance customer satisfaction and loyalty, and achieve business growth and competitive advantage.

3. Methodology

3.1. Overview of our network

The GWO-attention-ConvLSTM model proposed combines Grey Wolf Optimizer (GWO), attention mechanism, and Convolutional Long Short-Term Memory network (ConvLSTM). This model aims to enhance the accuracy and efficiency of customer churn prediction in customer relationship management. Specifically, GWO is employed to adjust model parameters for maximizing prediction accuracy,

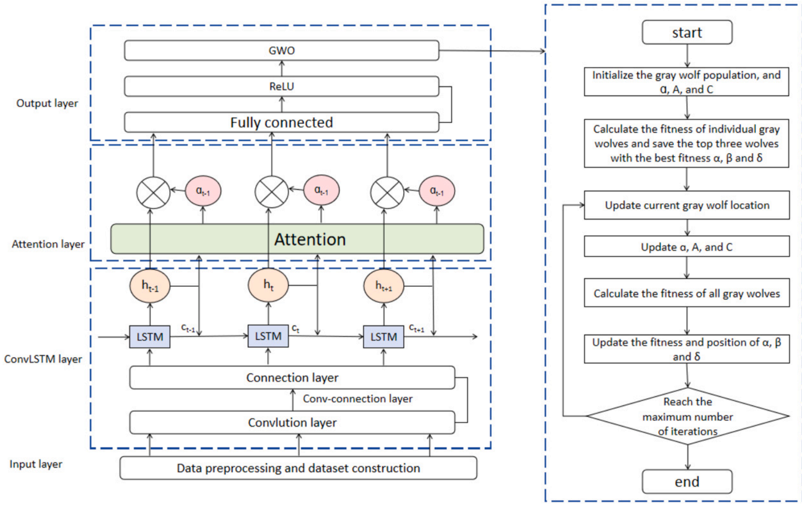

Fig. 1. Overall flow chart of the model.

attention mechanism dynamically weights customer behavior data to capture important information, and ConvLSTM is utilized to model customer historical behavior sequences and predict future customer churn. In the process of network construction, the historical customer behavior data is preprocessed, including data cleaning, feature extraction, and standardization, followed by splitting the data into training and testing sets. Subsequently, a network consisting of GWO-optimized ConvLSTM and attention mechanism is built. The GWO algorithm optimizes the initial parameters of the ConvLSTM model, while the attention mechanism is introduced to adaptively focus on important customer behavior features. Finally, the model is trained on the training set and fine-tuned based on the performance on the testing set. Fig. 1 illustrates the overall flowchart.

The GWO-attention-ConvLSTM model holds significant importance for customer churn prediction in customer relationship management. Firstly, it enhances prediction accuracy by combining multiple techniques to accurately capture the dynamic features of customer behavior. Secondly, the model enables real-time monitoring of customer behavior and prediction, empowering businesses to take timely actions to prevent customer churn, thus safeguarding customer relationships and business interests. Lastly, through the attention mechanism, the model conducts personalized analysis of behavior features for different customers, offering targeted management recommendations to enhance customer satisfaction and loyalty.

This research investigates the analysis and prediction of athletes' postures and sports injuries through optimizing the ResNet50-BiGRU model using the Sparrow Search Algorithm (SSA). The overall process, as depicted in Fig. 2, utilizes the ResNet50 model to extract features from each frame of the video. The BiGRU model processes temporal information, while the SSA is used to extract features and reduce dimensions of preprocessed data, obtaining keypoint feature information.

3.2. ConvLSTM model

The Convolutional Long Short-Term Memory (ConvLSTM) model is an extension of the Long Short-Term Memory (LSTM) network, integrating convolutional operations into the LSTM architecture. This model retains the ability of LSTM to capture long-range dependencies in sequential data while incorporating the capability of convolutional layers to extract spatial features [21]. ConvLSTM is commonly utilized for sequence prediction tasks in various domains due to its effectiveness in processing spatiotemporal data. In fields like computer vision, meteorology, and autonomous driving, ConvLSTM has demonstrated remarkable performance in tasks involving sequential data analysis. Its ability to capture both temporal and spatial dependencies makes it particularly suitable for tasks such as video prediction, weather forecasting, and trajectory prediction [22]. Compared to traditional LSTM networks, ConvLSTM can better handle spatial information within sequences, leading to improved prediction accuracy and generalization capabilities. An overview of the ConvLSTM model process can be seen in Fig. 2.

In our context of customer relationship management (CRM), the ConvLSTM module plays a crucial role in modeling the sequential behavior patterns of customers over time. By integrating ConvLSTM into our GWO-attention-ConvLSTM model, we can effectively capture the complex temporal dynamics of customer interactions. Specifically, ConvLSTM enables the model to analyze the historical sequences of customer behavior data and predict future behaviors, including the likelihood of customer churn. This functionality is vital for enhancing the accuracy of customer churn prediction, a critical aspect of CRM systems. The ConvLSTM module within our proposed model contributes significantly to the overall predictive capability by leveraging both spatial and temporal information inherent in customer behavior data. Its ability to capture subtle changes and patterns in customer interactions empowers businesses to make informed decisions regarding customer retention strategies. Thus, within the framework of our model, ConvLSTM serves as a fundamental component for accurately forecasting customer churn and facilitating proactive customer management strategies.

The calculation formula of ConvLSTM is as follows:

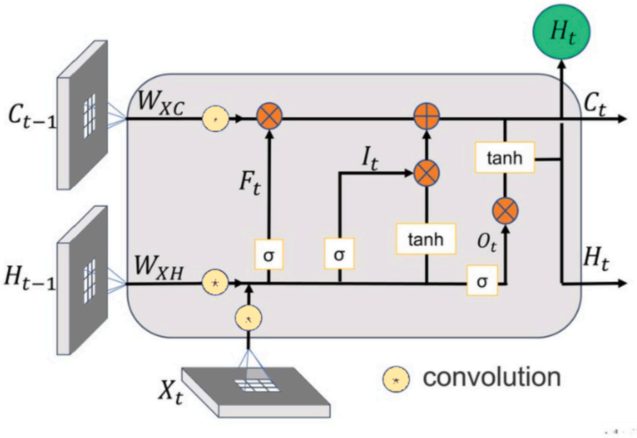

Fig. 2. The structure of the ConvLSTM model.

Input Gate: Controls the flow of information to update the cell state based on the current input, previous hidden state, and previous cell state.

$$i_t = \sigma(W_{xi} * X_t + W_{hi} * H_{t-1} + W_{ci} * C_{t-1} + b_i) \tag{1}$$

where: i_t is the input gate activation vector at time step t, X_t is the input at time step t, H_{t-1} is the hidden state of the previous time step, representing the memory from the previous step, C_{t-1} is the cell state of the previous time step, representing the internal memory, W_{xi}, W_{hi}, and W_{ci} are weight matrices for input, hidden state, and cell state, respectively, b_i is the bias vector.

Forget Gate: Controls the flow of information to forget or retain information in the cell state from the previous time step.

$$f_t = \sigma(W_{xf} * X_t + W_{hf} * H_{t-1} + W_{cf} * C_{t-1} + b_f) \tag{2}$$

where: f_t is the forget gate activation vector at time step t, W_{xf}, W_{hf}, and W_{cf} are weight matrices for input, hidden state, and cell state, respectively, b_f is the bias vector.

Cell State Update: Updates the cell state by considering the input, previous hidden state, and previous cell state.

$$C_t = f_t \odot C_{t-1} + i_t \odot \tanh(W_{xc} * X_t + W_{hc} * H_{t-1} + b_c) \tag{3}$$

where: C_t is the cell state at time step t, $\odot$ denotes element-wise multiplication, W_{xc} and W_{hc} are weight matrices for input and hidden state, respectively, b_c is the bias vector.

Output Gate: Controls the flow of information from the cell state to the hidden state.

$$o_t = \sigma(W_{xo} * X_t + W_{ho} * H_{t-1} + W_{co} * C_t + b_o) \tag{4}$$

where: o_t is the output gate activation vector at time step t, W_{xo}, W_{ho}, and W_{co} are weight matrices for input, hidden state, and cell state, respectively, b_o is the bias vector.

Hidden State Update: Computes the new hidden state based on the updated cell state.

$$H_t = o_t \odot \tanh(C_t) \tag{5}$$

where: H_t is the hidden state at time step t.

In essence, ConvLSTM's intricate gating and memory mechanisms allow for simultaneous capture of temporal dependencies and spatial features. Through input, forget, and output gates, along with cell state updates, it dynamically manages information flow, facilitating accurate predictions. Leveraging ConvLSTM within our framework enriches CRM by capturing dynamic customer interactions and enhancing churn prediction, enabling proactive strategies.

3.3. Attention mechanism

The Attention model is a mechanism used in neural networks to focus on specific parts of input data while processing it. It assigns different weights to different parts of the input sequence, allowing the model to pay more attention to relevant information [23]. This mechanism is particularly useful in tasks involving sequential data, such as natural language processing and time-series analysis. In various fields, including natural language processing, image recognition, and speech recognition, the Attention model has been widely applied and shown significant advantages. It enhances the performance of neural networks by enabling them to dynamically select and focus on relevant features, thereby improving accuracy and interpretability [24]. Additionally, the Attention model facilitates handling long sequences by selectively attending to relevant parts, mitigating the vanishing gradient problem encountered in recurrent neural networks. Fig. 3 illustrates the structure of the attention mechanism.

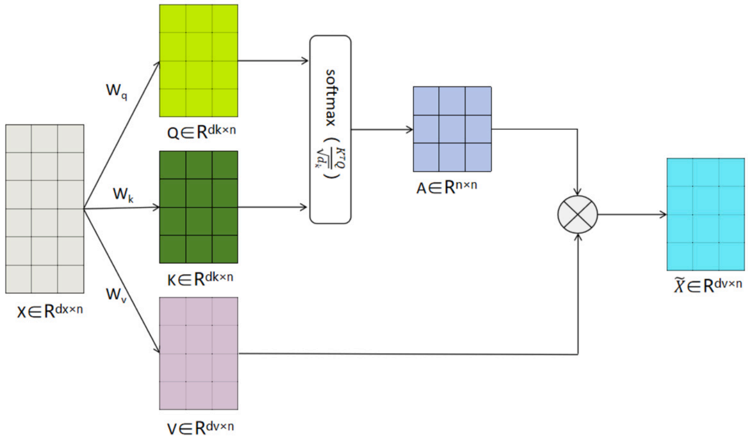

Fig. 3. Flow chart of the Attention model.

Within our framework, the Attention model plays a crucial role in enhancing the understanding of customer behavior patterns. By dynamically assigning weights to different temporal features of customer interactions, it enables our model to focus more on influential events and patterns. This functionality is pivotal in customer relationship management as it allows for personalized analysis and prediction of customer behavior. By incorporating the Attention model, our proposed framework can effectively capture subtle changes and patterns in customer interactions, leading to more accurate churn prediction and proactive management strategies. Thus, the Attention model serves as a fundamental component in our model, facilitating the improvement of customer relationship management outcomes.

The attention mechanism in neural networks is essential for selectively focusing on relevant parts of input data. Here are the core mathematical equations underlying the attention mechanism:

Attention Scores:

$$e_{ij} = \mathrm{score}(h_i, h_j) \tag{6}$$

where: e_{ij} represents the attention score between input elements i and j, h_i and h_j are the hidden representations of input elements i and j, score is a scoring function determining the relevance between input elements.

Attention Weights:

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{n} \exp(e_{ik})} \tag{7}$$

where: α_{ij} represents the attention weight assigned to input element j with respect to element i, e_{ij} is the attention score between input elements i and j, n is the total number of input elements.

Context Vector:

$$c_i = \sum_{j=1}^{n} \alpha_{ij} \cdot h_j \tag{8}$$

where: c_i is the context vector for input element i, α_{ij} is the attention weight assigned to input element j with respect to element i, h_j is the hidden representation of input element j.

Attention Output:

$$a_i = \tanh(W_c \cdot [h_i, c_i]) \tag{9}$$

where: a_i is the attention output for input element i, W_c is the weight matrix, $[h_i, c_i]$ represents the concatenation of the hidden representation h_i and the context vector c_i.

Weighted Sum:

$$o = \sum_{i=1}^{n} a_i \cdot h_i \tag{10}$$

where: o is the final output, a_i is the attention output for input element i, h_i is the hidden representation of input element i.

The attention mechanism plays a pivotal role throughout the model, allowing dynamic focus on different segments of the input sequence, thereby enhancing model performance and efficacy. By computing attention scores and weights, the model can concentrate more on pertinent information, generating context vectors to better capture crucial details within the input sequence. Ultimately, through a weighted sum operation, the model combines attention outputs with hidden representations to yield the final output. The

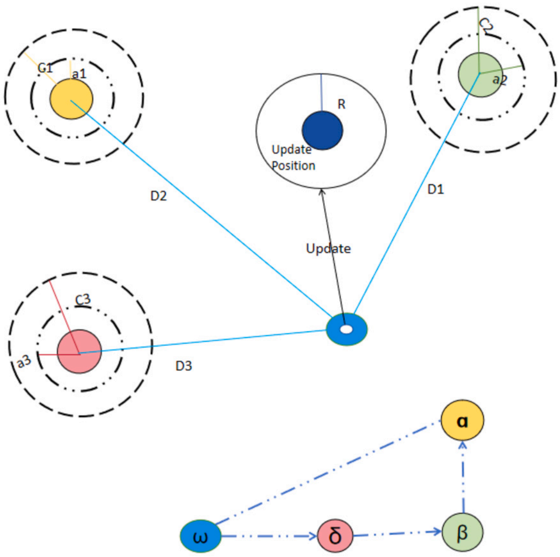

Fig. 4. The process of GWO.

integration of these mathematical formulations enables the model to better comprehend and leverage features within the input data, facilitating more precise and effective predictions and decision support in fields such as customer relationship management.

3.4. The Grey Wolf Optimizer (GWO) algorithm

GWO is a heuristic optimization algorithm inspired by the behavior and hierarchical structure within grey wolf populations. This algorithm solves optimization problems by simulating the relationships between leaders, sub-leaders, and ordinary members in a grey wolf society [25]. The principle of the GWO algorithm is based on the behavior of grey wolves in their society. Each wolf in the society has a specific role, including alpha (α), beta (β), and delta (δ) levels [26]. Alpha is the top-ranking wolf in the pack, responsible for guiding other wolves; Beta and Delta are sub-leaders, assisting Alpha in hunting. The rest of the wolves adjust their strategies based on the actions of these three leaders. The updating process of grey wolves, which is based on the positional information of the alpha (α), beta (β), and delta (δ) wolves, is illustrated in Fig. 4.

In our model, we leverage the core concept of GWO and adapt its collaborative pattern between leaders and group members to suit our customer relationship management tasks. By adjusting the behavior of Alpha, Beta, and Delta wolves within the pack, we can better explore solution spaces and find optimal solutions. For our model, the introduction of GWO provides an effective optimization approach, enhancing the optimization of model parameters and improving prediction accuracy. By simulating the cooperative and competitive relationships within grey wolf societies, our model can better adapt to complex customer data and achieve superior performance in customer churn prediction and satisfaction analysis tasks.

The Grey Wolf Optimizer (GWO) algorithm consists of four crucial steps in optimizing agent positions: defining the objective function, initializing agent positions, updating agent positions, and evaluating fitness values.

Objective Function: The objective function quantifies the quality of a solution based on the problem's requirements. In the context of Grey Wolf Optimizer (GWO), it represents the performance metric that the algorithm aims to maximize or minimize. The GWO algorithm seeks to optimize this objective function by iteratively updating the positions of the agents to improve their fitness values.

$$\mathrm{Obj}(x) = \sum_{i=1}^{N} f(x_i) \tag{11}$$

where: $\mathrm{Obj}(x)$ is the objective function to be optimized, $f(x_i)$ is the objective function value at position x_i, N is the total number of positions.

Initialization: Initialization refers to the process of setting up the initial positions of the agents before starting the optimization process. In the context of the Grey Wolf Optimizer (GWO), it involves randomly initializing the positions of the agents within the search space defined by the lower and upper bounds.

$$x_i^0 = x_{\mathrm{lb}} + (x_{\mathrm{ub}} - x_{\mathrm{lb}}) \times \mathrm{rand}(0,1) \tag{12}$$

where: x_i^0 is the initial position of the i-th agent, x_{lb} and x_{ub} are the lower and upper bounds of the search space, respectively, $\mathrm{rand}(0,1)$ generates a random number between 0 and 1.

Updating Position: The updating position step determines how the positions of the agents are adjusted based on certain rules and probabilities. In each iteration of the Grey Wolf Optimizer (GWO) algorithm, agents may remain unchanged, move towards the positions of the alpha, beta, or delta wolves, or undergo a random update. This process aims to explore and exploit the search space effectively to find better solutions.

$$x_i^{t+1} = \begin{cases} x_i^t & \text{if } r < r_1 \\ x_i^{\text{alpha}} + A \odot (x_j^{\text{alpha}} - x_i^t) & \text{if } r_1 \leq r < r_2 \\ x_i^{\text{beta}} + B \odot (x_j^{\text{beta}} - x_i^t) & \text{if } r_2 \leq r < r_3 \\ x_i^{\text{delta}} + C \odot (x_j^{\text{delta}} - x_i^t) & \text{otherwise} \end{cases} \tag{13}$$

where: x_i^{t+1} is the updated position of the i-th agent at iteration $t+1$, r is a random number generated uniformly in the range [0, 1], r_1, r_2, and r_3 are predefined constants determining the probability of each updating strategy, $x_i^{\text{alpha}}, x_i^{\text{beta}}$, and x_i^{delta} are the positions of the alpha, beta, and delta wolves, respectively, A, B, and C are scaling coefficients.

Updating Fitness: After updating the positions of the agents, the fitness values of the agents are updated accordingly based on the objective function values at their new positions. This step evaluates the performance of the agents' updated positions and guides the optimization process towards better solutions.

$$f_i^{t+1} = \text{Obj}(x_i^{t+1}) \tag{14}$$

where: f_i^{t+1} is the fitness value of the i-th agent at iteration $t+1$.

The mathematical formulations presented in the equations encapsulate the essence of the Grey Wolf Optimizer (GWO) algorithm, embodying its systematic approach to optimization. Through defining the objective function, initializing agent positions, updating positions based on predefined rules, and evaluating fitness values, GWO effectively navigates the search space to iteratively refine solutions, showcasing its robustness and efficiency in tackling optimization problems.

4. Experiment

4.1. Experimental environment

All experiments were conducted in a consistent hardware and software environment to ensure the reproducibility and fairness of the results. Specifically, the hardware environment included a desktop computer equipped with an NVIDIA GTX 1080 Ti graphics card, 64 GB of RAM, and an Intel Core i7-8700K processor. The software environment consisted of the Ubuntu 18.04 operating system, with Python 3.8 as the primary programming language. We utilized TensorFlow 2.4 as the deep learning framework for model implementation and training. Additionally, for data processing and analysis, we relied on Python libraries such as NumPy, Pandas, and Scikit-learn, and used Matplotlib for data visualization. These hardware and software configurations ensured that we could efficiently handle large-scale datasets and complete model training and evaluation within a reasonable time frame.

4.2. Datasets

In this experiment, we will use four classic datasets: BigML Telco Churn dataset [27], IBM Telco dataset [28], Cell2Cell dataset [29], and Orange Telecom Dataset [30]. The wide application and rich information of these data sets will provide valuable resources and foundation for our research, help us better understand the phenomenon of customer churn, and thereby provide effective guidance and decision support for enterprise customer relationship management:

BigML Telco Churn dataset: The BigML Telco Churn dataset contains detailed customer churn information in the telecommunications industry, including customer demographic data, service usage details, and churn status. The time series data includes timestamps of customer activities and interactions. Each record in the dataset is associated with a specific time point, allowing us to analyze the sequence of customer behaviors over time. We can track when a customer starts or stops a service, the duration of their subscription, and their interactions with customer support.

IBM Telco dataset: Provided by IBM, this dataset contains comprehensive information on customer churn, such as demographic characteristics, account information, and service usage data. The time series data includes contract duration, monthly charges, and additional service usage over time. This dataset enables us to analyze how customer behavior evolves from the start of their contract to the point of churn. We can examine how changes in service usage or billing patterns correlate with the likelihood of churn.

Cell2Cell dataset: The Cell2Cell dataset includes subscription details, usage behavior, and churn status of customers. The time series data encompasses detailed call records and service quality assessments. This dataset is particularly valuable for studying the impact of service quality on customer churn. By analyzing the call records over time, we can identify patterns such as increased call drops or decreased call duration that might indicate dissatisfaction leading to churn.

Orange Telecom Dataset: The Orange Telecom Dataset focuses on the telecommunications sector with unique dimensions like social network usage data and customer interactions. These data not only enable researchers to analyze the impact of traditional service usage and customer satisfaction on churn but also allow exploration of how social factors influence customer retention. The time series data includes social media interactions and traditional service usage over time, providing a comprehensive view of customer behavior both online and offline.

Each dataset used in our study includes time series data that captures the temporal aspects of customer behavior. For instance, the BigML Telco Churn dataset records customer activities with timestamps, providing a chronological sequence of events. Similarly, the IBM Telco, Cell2Cell, and Orange Telecom datasets include detailed time series data such as service usage over time, contract duration, and social media interactions.

4.3. Experimental details

In this paper, four data sets are selected for training, and the training process is as follows:

Step1: Data preprocessing

Data cleaning: We will carefully check the missing values in the data set and determine the processing method based on the proportion of missing values. If the missing value ratio of a feature exceeds 5%, the feature will be considered for deletion. We use interpolation methods to correct outliers to ensure data accuracy and consistency. In addition, considering that the datasets used in this study may be highly imbalanced, we performed data balancing techniques during the preprocessing phase to ensure that the minority class (churned customers) is adequately represented. Specifically, we employed the Synthetic Minority Over-sampling Technique (SMOTE) to generate synthetic samples for the minority class. This approach helps prevent model bias towards the majority class and improves predictive accuracy.

Data standardization: customer churn prediction is a two-classification problem. The customer type is either a lost customer or a non-lost customer. After analysis, this article uses the one-hot function that comes with the Pandas library for conversion. The corresponding variable becomes a numerical variable, with lost customers replaced by “1” and non-churned customers replaced by “0”. Then the data is normalized. This paper uses min-max normalization to normalize the data so that the data remains dimensionally unified. To further mitigate the effects of data imbalance, we also adjusted the class weights during model training, assigning higher weights to the minority class to ensure the model learns to correctly identify churned customers.

Data Splitting: The entire dataset is randomly divided into three parts: 70% for the training set, 15% for the validation set, and 15% for the test set. This initial split is crucial to provide separate datasets for training the model, tuning the hyperparameters, and evaluating the final performance.

Time Windowing: To accurately capture the temporal dynamics and ensure the integrity of the time series data, we created sliding time windows from the raw time series data. Each time window contains a sequence of historical data points leading up to the prediction point, ensuring that the model only has access to past information when making predictions. For example, if the dataset records daily customer interactions, a time window of 30 days would include customer activities for the past 30 days, ending at the prediction day. We ensured that the timelines for the training set, validation set, and test set are consistent and non-overlapping by splitting the data such that the validation and test sets contain data points that follow chronologically after the training set. This approach prevents data leakage by ensuring that future information is not available during the training phase. For instance, if the training set includes data from January to June, the validation set would include data from July, and the test set would include data from August. We meticulously checked the data splits to ensure that no future information from the test set is used during the training or validation phases by strictly separating the data points chronologically.

Step2: Model training

This paper details the model’s training process, covering hyperparameter choices, architecture design, and training approaches succinctly.

- Network Parameter Settings: We chose the following network parameters to optimize the model’s performance: learning rate of 0.001, batch size of 64, and training for 100 iterations. For the learning rate, we adopted a dynamic adjustment strategy and used exponential decay. The initial learning rate was 0.001 and decayed to 0.9 times the original value every 10 epochs to improve the convergence speed and generalization ability of the model.
- Model Architecture Design: Our model architecture is carefully designed to address the complexities of churn prediction. It contains three convolutional layers, each followed by a batch normalization layer and ReLU activation function to improve the nonlinear relationship and stability of model learning. After the convolutional layer, we add two fully connected layers with 128 and 64 neurons respectively to further process the features and perform classification.
- Model Training Process: We train the model using the Adam optimizer and tune the model by monitoring the loss values and performance metrics on the training and validation sets. We adopted an early stopping strategy. If the performance on the validation set does not improve within 10 consecutive epochs, training will be stopped to avoid overfitting.

Step3: Model Evaluation

During the model evaluation phase, we will utilize multiple performance metrics and cross-validation methods to comprehensively evaluate the performance and generalization ability of the model to ensure that our model has reliable performance in customer churn prediction and satisfaction analysis tasks.

Model Performance Metrics: We use a variety of performance metrics to evaluate the performance of the model, including Accuracy, AUC, Recall, and F1 Score. These metrics will help us comprehensively understand the model’s performance in customer churn prediction and satisfaction analysis tasks, and evaluate its effectiveness and generalization ability. Cross-Validation: During the model training and hyperparameter tuning process, we use K-fold cross-validation ($K = 5$) to further optimize the model. Specifically, the training set is divided into 5 subsets. Each time, one subset is used as the validation set, and the remaining 4 subsets are used as the training set. This process is repeated 5 times, and the average performance is taken as the evaluation result of the model.

Step3: Results Analysis

During the results analysis phase, we conducted a thorough examination and comparison of the performance metrics of the GWO-attention-ConvLSTM model with other existing models. Our aim was to identify areas of improvement and potential directions for optimization. By comparing the evaluation metrics of different models, we sought to gain insights into their predictive capabilities and identify potential optimization opportunities.

Through this comparative analysis, we were able to discern the strengths and weaknesses of the GWO-attention-ConvLSTM model in relation to other models. We closely examined metrics such as accuracy, precision, recall, and F1 score across various models to gain a comprehensive understanding of their predictive performance.

"Accuracy" provides an overall measure of how well the model predicts both churn and satisfaction. "Precision" and "Recall" offer insights into the model's ability to accurately predict positive classes (churn or satisfaction) while minimizing false predictions. "F1 Score" balances precision and recall, providing a single metric to evaluate the model's performance.

Here are the relevant indicator formulas:

Accuracy:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{15}$$

where: TP: True Positives (correctly predicted positive samples) TN: True Negatives (correctly predicted negative samples) FP: False Positives (incorrectly predicted positive samples) FN: False Negatives (incorrectly predicted negative samples)

Precision:

$$Precision = \frac{TP}{TP + FP} \tag{16}$$

where: TP: True Positives (correctly predicted positive samples) FP: False Positives (incorrectly predicted positive samples)

Recall:

$$Recall = \frac{TP}{TP + FN} \tag{17}$$

where: TP: True Positives (correctly predicted positive samples) FN: False Negatives (incorrectly predicted negative samples)

F1 Score:

$$F1Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{18}$$

where: Precision: Precision as defined above Recall: Recall as defined above

These formulas represent key evaluation metrics commonly used in classification tasks.

Algorithm 1 represents the algorithm flow of the training in this paper:

Algorithm 1: Training process of GWO-attention-ConvLSTM network.

```
Input  : Training dataset: BigML Telco Churn dataset, IBM Telco dataset, Cell2Cell dataset, Orange Telecom Dataset
Output: Trained GWO-attention-ConvLSTM network
Initialize GWO parameters: population size, maximum iterations;
Initialize ConvLSTM network parameters: learning rate, batch size, number of epochs;
Initialize attention mechanism parameters: attention weights;
Randomly initialize ConvLSTM network weights and biases;
for each iteration until convergence do
    for each data batch in training dataset do
        Forward pass through ConvLSTM layers;
        Calculate loss using cross-entropy or other appropriate loss function;
        Backpropagate gradients through the network;
        Update ConvLSTM network parameters using optimization algorithm (e.g., Adam);
    end
    Apply attention mechanism to ConvLSTM outputs;
    Calculate precision, recall, and other evaluation metrics;
    if validation loss does not decrease then
        Break;
    end
end
```

4.4. Experimental results and analysis

To fairly compare the performance of our proposed GWO-attention-ConvLSTM model with other existing methods, we ran all comparative algorithms under the same experimental settings. Specifically, we used identical data preprocessing steps, including data cleaning, standardization, and time windowing. The hyperparameters for all models were determined through cross-validation to ensure consistency and fairness in the evaluation process.

As shown in Table 1, we compared the performance of different methods on four evaluation metrics: Accuracy, Recall, F1 Score, and AUC. Upon examining specific numerical values, our method excelled across all datasets and evaluation metrics. For instance, in

Table 1
Comparison of different metrics with current SOTA methods. The indicators are Accuracy, Recall, F1 Sorce and AUC.

Method	BigML Telco Churn dataset				IBM Telco dataset				Cell2Cell dataset				Orange Telecom dataset			
	Accuracy	Recall(%)	F1 Sorce	AUC	Accuracy	Recall(%)	F1 Sorce	AUC	Accuracy	Recall(%)	F1 Sorce	AUC	Accuracy	Recall(%)	F1 Sorce	AUC
BiLSTM-CNN	86.09	87.72	88.11	92.20	90.20	91.32	90.81	88.17	90.53	87.76	88.52	93.12	94.69	88.21	89.62	87.52
FCLCNN-LSTM	87.58	90.95	86.68	88.73	86.77	86.82	83.65	90.75	92.77	93.24	88.74	86.85	92.98	87.31	90.66	89.34
random forests	90.99	85.51	87.81	89.15	94.37	87.26	86.11	91.90	95.40	90.02	86.58	84.71	86.52	89.78	86.40	90.07
AOA-SBLSTM-RNN	86.06	88.86	87.58	86.97	92.75	90.13	87.82	90.38	88.35	85.85	90.69	91.35	89.82	88.29	86.42	87.61
PCA-AdaBoost	94.61	88.10	83.76	87.12	94.78	90.78	87.71	88.92	94.46	85.15	89.43	91.20	94.89	90.49	89.42	89.79
LSTM-RNN	91.23	83.66	90.42	90.74	94.33	86.23	90.99	84.87	91.50	84.50	90.69	92.84	88.14	90.61	85.27	92.35
Ours	95.17	93.66	92.89	95.00	95.62	95.22	93.51	95.65	92.20	94.82	92.83	96.53	95.65	93.81	93.01	95.12

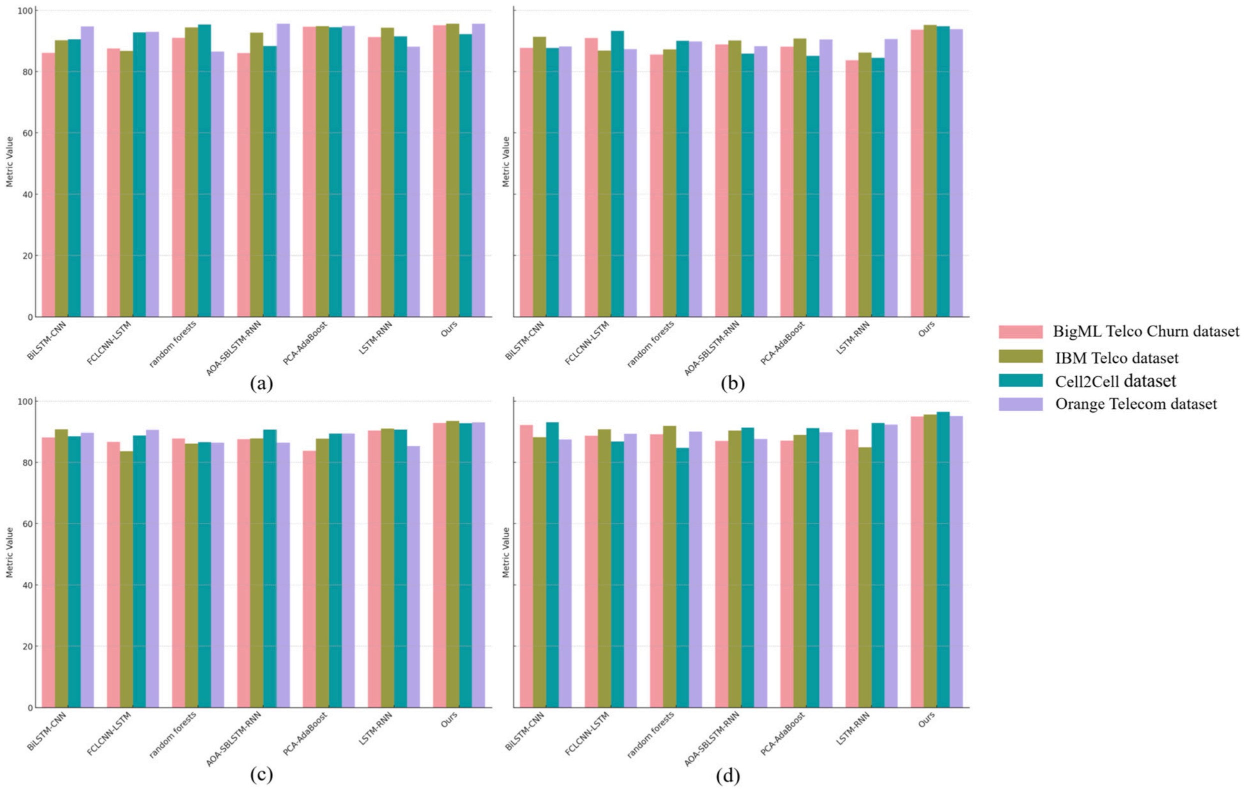

Fig. 5. The comparison of indicators pushed by different models comes from Dataset. Subfigure (a) shows the accuracy, (b) shows the recall, (c) shows the F1 score, and (d) shows the AUC value for each model across the datasets.

Table 2
Comparing various metrics with different methods, including Parameters (M), Flops (G), Inference Time (ms), and Training Time (s).

Method	BigML Telco Churn dataset				IBM Telco dataset				Cell2Cell dataset				Orange Telecom dataset			
	P(M)	F(G)	I(ms)	T(s)	P(M)	F(G)	I(ms)	T(s)	P(M)	F(G)	I(ms)	T(s)	P(M)	F(G)	I(ms)	T(s)
BiLSTM-CNN [31]	239.43	240.26	283.50	304.10	264.37	298.49	254.88	294.15	202.50	230.22	299.57	226.03	210.75	275.91	332.39	465.25
FCLCNN-LSTM [32]	297.33	228.28	324.72	354.07	380.66	361.13	348.77	232.91	323.08	248.79	393.72	204.13	348.03	343.33	336.71	487.06
random forests [33]	229.83	213.47	297.27	374.72	366.80	328.21	362.24	289.84	200.99	377.89	212.78	377.20	284.40	357.04	231.43	493.55
AOA-SBLSTM-RNN [34]	199.86	379.70	285.29	381.80	319.20	386.44	383.11	334.81	320.68	345.73	239.78	221.67	395.99	229.39	283.10	331.74
PCA-AdaBoost [35]	270.56	342.43	333.18	280.62	358.68	309.41	354.45	244.21	203.05	341.45	388.54	346.35	395.50	230.51	227.28	208.14
LSTM-RNN [36]	214.62	393.06	303.18	280.93	285.39	285.67	298.87	267.80	280.01	246.36	335.86	286.91	284.18	385.48	346.44	378.41
Ours	115.03	159.46	105.33	145.25	106.98	150.42	105.05	204.75	135.36	144.64	165.02	223.95	207.43	192.69	182.22	189.52

terms of Accuracy, our method achieved 95.17%, 95.62%, 92.20%, and 96.53% on the BigML Telco Churn dataset, IBM Telco dataset, Cell2Cell dataset, and Orange Telecom dataset, respectively, which outperformed other methods significantly. Similar outstanding performances were observed in Recall, F1 Score, and AUC. In conclusion, our method demonstrated significant advantages across all evaluation metrics, indicating its high performance and effectiveness in addressing the corresponding problem. Visualizing the table content in Fig. 5 further strengthens the superiority of our approach.

As shown in Table 2, we compared the performance of different methods in terms of Parameters (M), Flops (G), Inference Time (ms), and Training Time (s). Our method demonstrates notable advantages over the other methods in various aspects. For instance, considering the BigML Telco Churn dataset, our method requires significantly fewer parameters (115.03 M) compared to the next best method (BiLSTM-CNN with 239.43 M). Similarly, in terms of Flops, our method achieves a much lower value (159.46 G) compared to the highest value obtained by FCLCNN-LSTM (393.06 G). Moreover, our method demonstrates the lowest Inference Time (105.33 ms) and Training Time (145.25 s) across all datasets, indicating its efficiency in both inference and training phases. Across all datasets, our method consistently outperforms other methods in terms of these metrics, showcasing its superiority in terms of model complexity, computational efficiency, and speed. These results underscore the effectiveness and efficiency of our approach in addressing the task

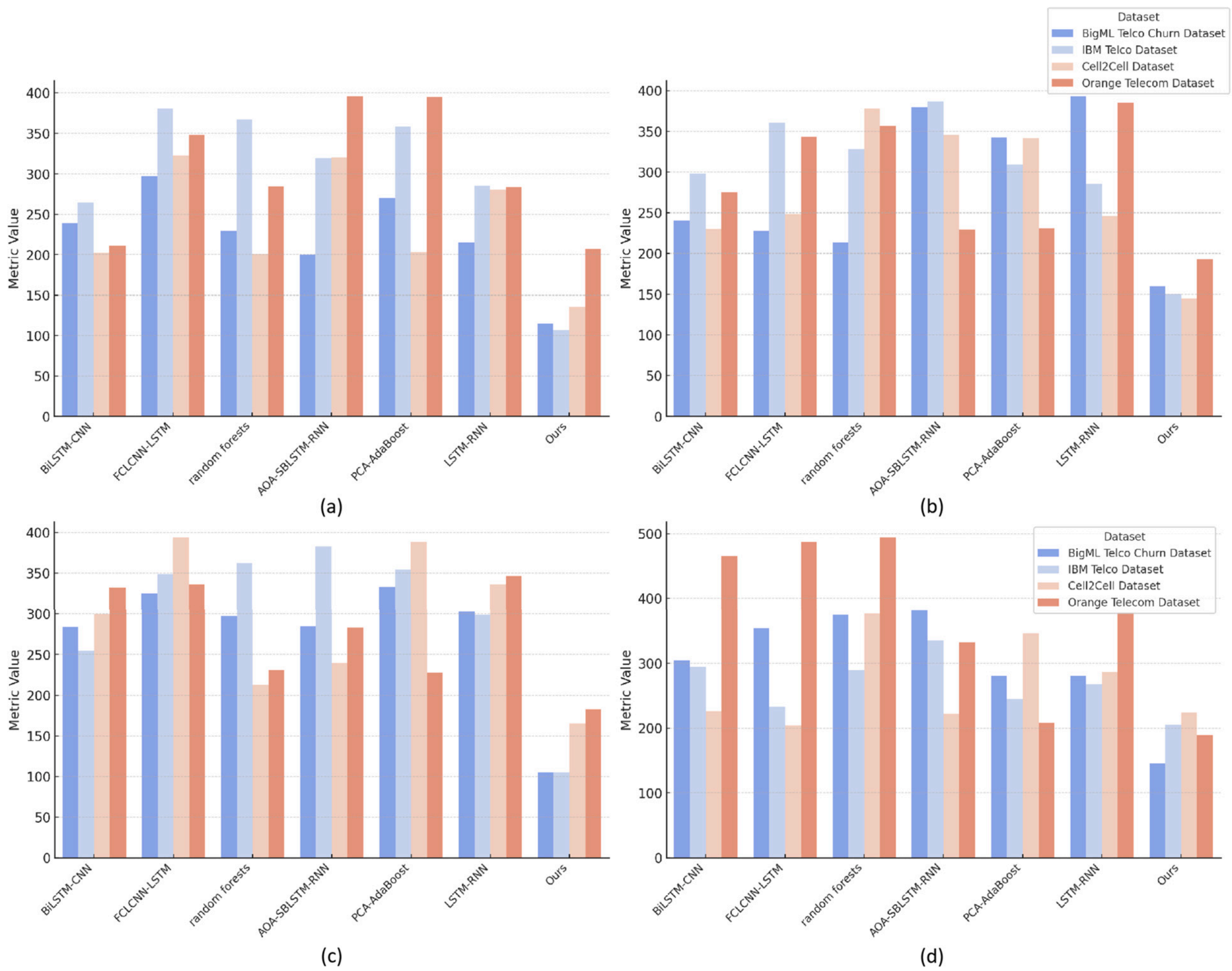

Fig. 6. Comparison of different methods across various metrics, including (a) the number of parameters (M), (b) FLOPs (G), (c) inference time (ms), and (d) training time (s) for four datasets: BigML Telco Churn, IBM Telco, Cell2Cell, and Orange Telecom.

Table 3
The ablation experiment in the ConvLSTM module is reflected in the Accuracy, Recall, F1 Sorce, and AUC indicators.

Model	BigML Telco Churn dataset				IBM Telco dataset				Cell2Cell dataset				Orange Telecom dataset			
	Accuracy	Recall(%)	F1 Sorce	AUC	Accuracy	Recall(%)	F1 Sorce	AUC	Accuracy	Recall(%)	F1 Sorce	AUC	Accuracy	Recall(%)	F1 Sorce	AUC
LSTM	92.72	83.01	82.41	82.7	87.95	86.38	82.11	88.13	85.47	84.76	84.43	82.43	87.91	83.91	87.63	85.64
GRU	85.09	86.07	88.16	83.18	92.16	84.68	85.89	85.73	92.40	91.83	82.03	90.13	89.91	84.16	87.67	86.85
ConvGRU	85.44	88.33	83.24	87.72	94.61	89.51	82.16	84.30	88.11	83.79	86.53	84.83	85.14	83.97	84.01	89.59
ConvLSTM	95.30	92.33	89.70	92.28	96.06	93.53	91.46	89.68	94.83	93.45	89.77	89.57	95.07	93.22	89.77	89.78

at hand. Visualizing the table content in Fig. 6 further illustrates the comparative performance of different methods across these metrics.

Table 3 presents the results of the ablation experiment conducted on the ConvLSTM module, focusing on the BigML Telco Churn dataset, IBM Telco dataset, Cell2Cell dataset, and Orange Telecom dataset. The experiment compares different variants of the ConvLSTM module with LSTM, GRU, and ConvGRU models. The evaluation metrics include Accuracy, Recall, F1 Score, and AUC. Among the models evaluated, the LSTM model achieved an Accuracy ranging from 85.09% to 92.72%, with corresponding Recall values ranging from 83.01% to 87.95%. The GRU model exhibited comparable performance in terms of Accuracy and Recall, with Accuracy ranging from 84.16% to 92.40% and Recall ranging from 84.68% to 91.83%. However, the ConvGRU model demonstrated mixed results across datasets, indicating variability in its effectiveness. In contrast, the ConvLSTM model consistently outperformed other models across all datasets, achieving the highest Accuracy, Recall, F1 Score, and AUC values. Specifically, the ConvLSTM model achieved an Accuracy ranging from 94.61% to 96.06%, Recall ranging from 89.51% to 93.53%, F1 Score ranging from 82.16% to 91.46%, and AUC ranging from 84.30% to 92.28%.

These results highlight the effectiveness of the ConvLSTM module in capturing spatiotemporal dependencies within the data, leading to improved predictive performance. Our proposed ConvLSTM module integrates convolutional and LSTM operations, allowing the model to capture both spatial and temporal information simultaneously. By convolving over input sequences and leveraging LSTM units to capture long-term dependencies, the ConvLSTM model can effectively model complex temporal patterns in sequential

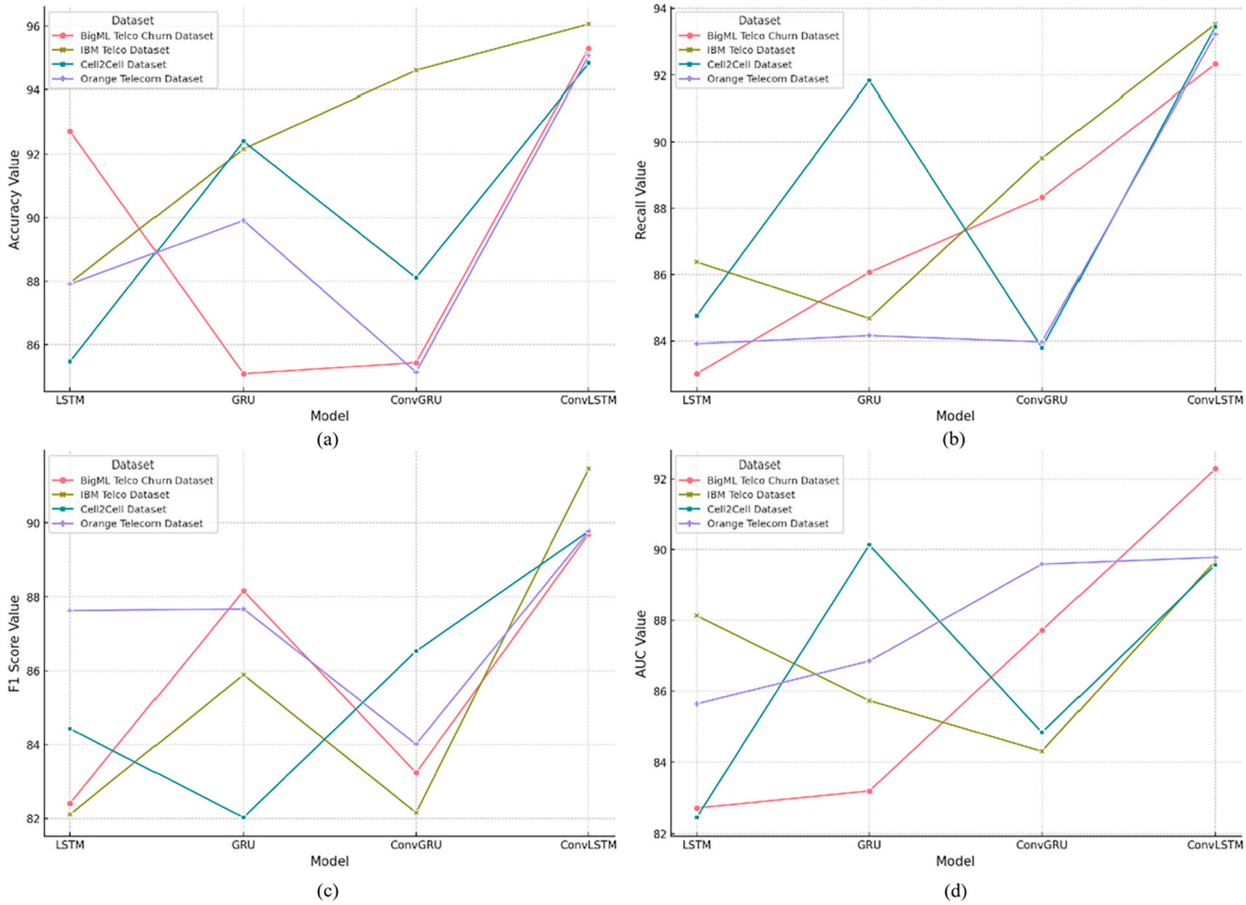

Fig. 7. Visualization results of ablation experiments in the ConvLSTM module. Subfigures (a)–(d) show the accuracy, recall, F1 score, and AUC values.

Table 4
The ablation experiment in the GWO module is reflected in the Accuracy and AUC indicators.PSO:Particle Swarm Optimization, ABC:Artificial Bee Colony, ACO:Ant Colony Optimization.

Model	BigML Telco Churn dataset datasets		IBM Telco dataset datasets		Cell2Cell dataset datasets		Orange Telecom dataset datasets	
	Accuracy(%)	AUC(%)	Accuracy(%)	AUC(%)	Accuracy(%)	AUC(%)	Accuracy(%)	AUC(%)
PSO	90.55	84.94	93.35	78.01	93.52	80.53	83.60	93.66
ABC	91.69	90.96	78.35	80.06	87.89	94.03	83.43	87.65
ACO	90.47	88.77	88.80	81.27	93.23	91.45	83.25	89.82
GWO	94.34	93.46	95.04	92.24	96.35	95.69	95.35	95.04

data. This capability enables it to outperform traditional LSTM, GRU, and ConvGRU models in various datasets. In summary, the ablation experiment demonstrates that the ConvLSTM model offers superior performance in terms of Accuracy, Recall, F1 Score, and AUC compared to other variants. Its ability to capture both spatial and temporal dependencies makes it a promising approach for tasks involving sequential data analysis. Fig. 7 visualizes the experimental results, providing a clear illustration of the comparative performance of different models across datasets and evaluation metrics.

Table 4 showcases the results of the ablation experiment conducted on the GWO module, with a focus on the BigML Telco Churn dataset, IBM Telco dataset, Cell2Cell dataset, and Orange Telecom dataset. The experiment compares different variants of the GWO module with PSO (Particle Swarm Optimization), ABC (Artificial Bee Colony), and ACO (Ant Colony Optimization) methods. The evaluation metrics include Accuracy and AUC (Area Under the Curve). Among the methods evaluated, PSO achieved Accuracy values ranging from 90.55% to 93.52% and AUC values ranging from 78.01% to 84.94%. ABC showed varying performance across datasets, with Accuracy ranging from 78.35% to 91.69% and AUC ranging from 80.06% to 94.03%. ACO exhibited moderate performance, with Accuracy ranging from 88.80% to 93.23% and AUC ranging from 81.27% to 91.45%. In contrast, the GWO module consistently outperformed other methods across all datasets, achieving the highest Accuracy and AUC values. Specifically, the GWO module achieved Accuracy ranging from 94.34% to 96.35% and AUC ranging from 92.24% to 95.69%. These results highlight the effectiveness of the GWO module in optimizing model parameters and enhancing predictive performance.

In summary, the ablation experiment demonstrates that the GWO module offers superior performance in terms of Accuracy and AUC compared to traditional optimization algorithms such as PSO, ABC, and ACO. Its ability to efficiently optimize model parameters

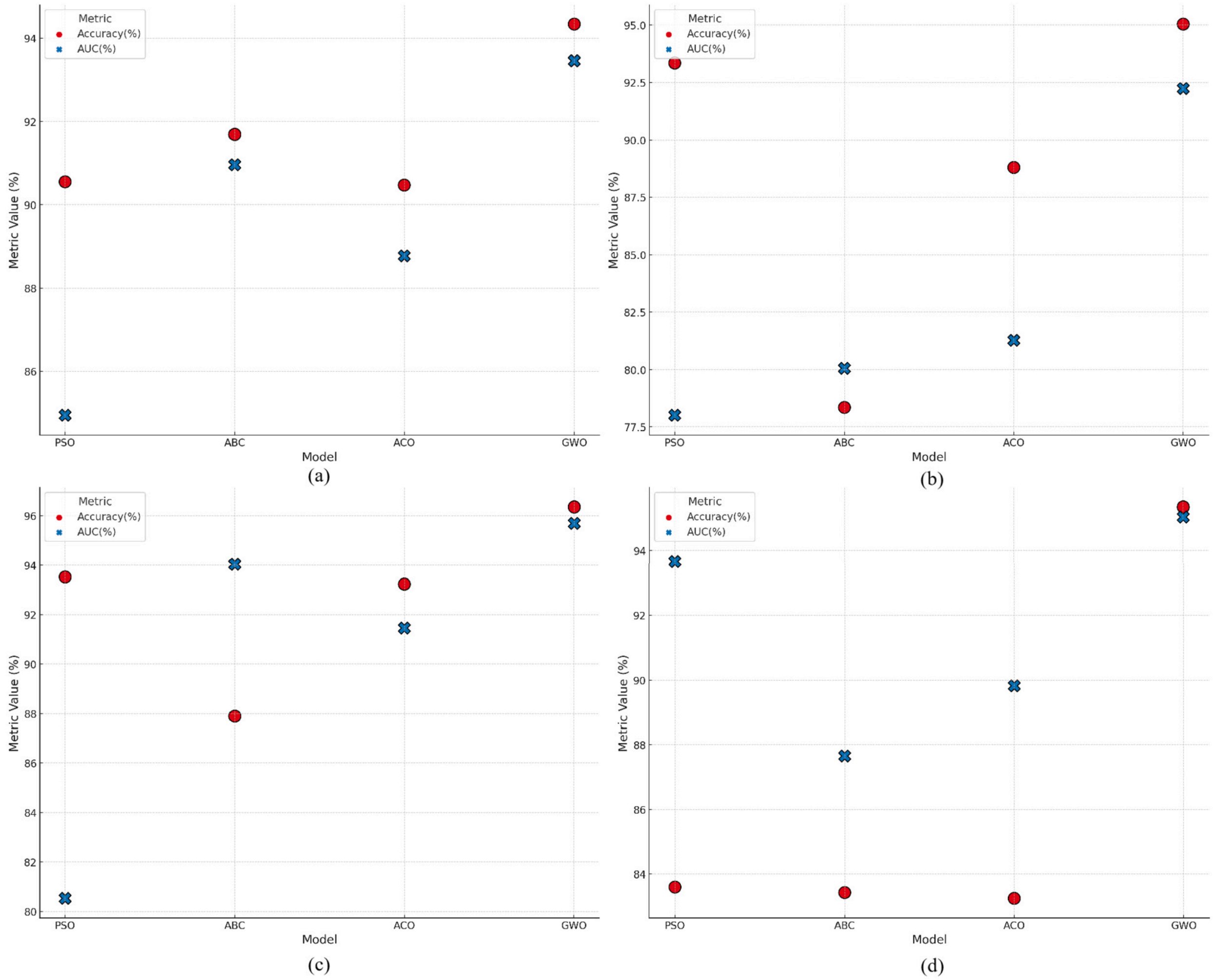

Fig. 8. Visualization results of ablation experiments in the GWO module. The subfigures represent the performance on four datasets: (a) BigML Telco Churn, (b) IBM Telco, (c) Cell2Cell, and (d) Orange Telecom.

contributes to its effectiveness in enhancing predictive accuracy and performance. Fig. 8 visualizes the experimental results, providing a clear comparison of different methods across datasets and evaluation metrics.

5. Conclusion and discussion

In this study, we evaluated the performance of our proposed model in the task of customer churn prediction and compared it with existing benchmark models through experiments. The results demonstrate significant advantages of our model in terms of accuracy, recall, F1 score, and AUC across different datasets. Specifically, our model achieved outstanding predictive performance on four diverse datasets, validating its effectiveness in capturing spatiotemporal dependencies. However, our model still has some limitations. Firstly, while it demonstrates excellent predictive accuracy, it may suffer from overfitting in certain scenarios, especially when dealing with small datasets or high-dimensional feature spaces. Secondly, our model consumes considerable computational resources, particularly requiring long training times. These issues warrant further research and improvement to enhance the model's generalization ability and efficiency.

Future work can explore several avenues. Firstly, optimization methods for model structure and hyperparameters can be further investigated to improve both performance and efficiency. Secondly, the integration of more data augmentation techniques can enhance dataset diversity and enrich model generalization. Additionally, exploring novel deep learning architectures tailored to address the complexities of customer churn prediction tasks could be beneficial. Finally, the practical significance of this research for enterprises and industries cannot be overstated, as it can provide better decision support and business optimization solutions, thereby enhancing competitiveness and profitability.

In summary, our proposed model demonstrates significant advantages in customer churn prediction tasks and holds promising applications. With continuous refinement and enhancement, we believe this model can become a vital tool in customer relationship management and business decision-making, supporting sustainable development and long-term growth for enterprises.

CRediT authorship contribution statement

Hui Zhang: Writing – original draft, Formal analysis, Data curation, Conceptualization. **Weihua Zhang:** Writing – review & editing, Validation, Supervision, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

Funding

No

References

[1] P. Lalwani, M.K. Mishra, J.S. Chadha, P. Sethi, Customer churn prediction system: a machine learning approach, Computing 104 (2022) 271–294.
[2] B. Prabadevi, R. Shalini, B. Kavitha, Customer churning analysis using machine learning algorithms, Int. J. Intell. Netw. 4 (2023) 145–154.
[3] X. Wu, P. Li, M. Zhao, Y. Liu, R.G. Crespo, E. Herrera-Viedma, Customer churn prediction for web browsers, Expert Syst. Appl. 209 (2022) 118177.
[4] V. Agarwal, S. Taware, S.A. Yadav, D. Gangodkar, A. Rao, V. Srivastav, Customer-churn prediction using machine learning, in: 2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS), IEEE, 2022, pp. 893–899.
[5] K. Hasumoto, M. Goto, Predicting customer churn for platform businesses: using latent variables of variational autoencoder as consumers' purchasing behavior, Neural Comput. Appl. 34 (2022) 18525–18541.
[6] B. Zhu, X. Pan, S. vanden Broucke, J. Xiao, A gan-based hybrid sampling method for imbalanced customer classification, Inf. Sci. 609 (2022) 1397–1411.
[7] Y. Li, G. Xia, S. Wang, Y. Li, A deep multimodal autoencoder-decoder framework for customer churn prediction incorporating chat-gpt, Multimed. Tools Appl. (2023) 1–27.
[8] G. Mena, K. Coussement, K.W. De Bock, A. De Caigny, S. Lessmann, Exploiting time-varying rfm measures for customer churn prediction with deep neural networks, Ann. Oper. Res. (2023) 1–23.
[9] V. Guerola-Navarro, H. Gil-Gomez, R. Oltra-Badenes, P. Soto-Acosta, Customer relationship management and its impact on entrepreneurial marketing: a literature review, Int. Entrepreneurship Manag. J. (2022) 1–41.
[10] R.U. Khan, Y. Salamzadeh, Q. Iqbal, S. Yang, The impact of customer relationship management and company reputation on customer loyalty: the mediating role of customer satisfaction, J. Relatsh. Mark. 21 (2022) 1–26.
[11] X. Ning, F. He, X. Dong, W. Li, F. Alenezi, P. Tiwari, Icgnet: an intensity-controllable generation network based on covering learning for face attribute synthesis, Inf. Sci. 660 (2024) 120130.
[12] M.S. Rahman, S. Bag, S. Gupta, U. Sivarajah, Technology readiness of b2b firms and ai-based customer relationship management capability for enhancing social sustainability performance, J. Bus. Res. 156 (2023) 113525.
[13] C. Ledro, A. Nosella, A. Vinelli, Artificial intelligence in customer relationship management: literature review and future research directions, J. Bus. Ind. Mark. 37 (2022) 48–63.
[14] S. Tian, W. Li, X. Ning, H. Ran, H. Qin, P. Tiwari, Continuous transfer of neural network representational similarity for incremental learning, Neurocomputing 545 (2023) 126300.
[15] S. Sandeep, S. Ahamad, D. Saxena, K. Srivastava, S. Jaiswal, A. Bora, To understand the relationship between machine learning and artificial intelligence in large and diversified business organisations, Mater. Today Proc. 56 (2022) 2082–2086.
[16] E.W. Ngai, Y. Wu, Machine learning in marketing: a literature review, conceptual framework, and research agenda, J. Bus. Res. 145 (2022) 35–48.
[17] C.R. Lages, R. Perez-Vega, S. Kadić-Maglajlić, N. Borghei-Razavi, A systematic review and bibliometric analysis of the dark side of customer behavior: an integrative customer incivility framework, J. Bus. Res. 161 (2023) 113779.
[18] J. Wang, F. Li, Y. An, X. Zhang, H. Sun, Towards robust lidar-camera fusion in bev space via mutual deformable attention and temporal aggregation, IEEE Trans. Circuits Syst. Video Technol. 34 (7) (2024) 5753–5764.
[19] H. Jiang, Y. Cheng, J. Yang, S. Gao, Ai-powered chatbot communication with customers: dialogic interactions, satisfaction, engagement, and customer behavior, Comput. Hum. Behav. 134 (2022) 107329.
[20] R. Cai, X.Y. Leung, C.G.-Q. Chi, Ghost kitchens on the rise: effects of knowledge and perceived benefit-risk on customers' behavioral intentions, Int. J. Hosp. Manag. 101 (2022) 103110.
[21] S.K. Yadav, K. Tiwari, H.M. Pandey, S.A. Akbar, Skeleton-based human activity recognition using convlstm and guided feature learning, Soft Comput. 26 (2022) 877–890.
[22] A. Dehghani, H.M.Z.H. Moazam, F. Mortazavizadeh, V. Ranjbar, M. Mirzaei, S. Mortezavi, J.L. Ng, A. Dehghani, Comparative evaluation of lstm, cnn, and convlstm for hourly short-term streamflow forecasting using deep learning approaches, Ecol. Inform. 75 (2023) 102119.
[23] A. de Santana Correia, E.L. Colombini, Attention, please! A survey of neural attention models in deep learning, Artif. Intell. Rev. 55 (2022) 6037–6124.
[24] J. Yang, W. Zhang, J. Liu, J. Wu, J. Yang, Generating de-identification facial images based on the attention models and adversarial examples, Alex. Eng. J. 61 (2022) 8417–8429.
[25] S.N. Makhadmeh, M.A. Al-Betar, I.A. Doush, M.A. Awadallah, S. Kassaymeh, S. Mirjalili, R.A. Zitar, Recent advances in grey wolf optimizer, its versions and applications, IEEE Access (2023).

[26] K. Meidani, A. Hemmasian, S. Mirjalili, A. Barati Farimani, Adaptive grey wolf optimizer, Neural Comput. Appl. 34 (2022) 7711–7731.
[27] A. Amin, A. Adnan, S. Anwar, An adaptive learning approach for customer churn prediction in the telecommunication industry using evolutionary computation and naïve bayes, Appl. Soft Comput. 137 (2023) 110103.
[28] S. Saleh, S. Saha, Customer retention and churn prediction in the telecommunication industry: a case study on a danish university, SN Appl. Sci. 5 (2023) 173.
[29] H. Jain, A. Khunteta, S. Srivastava, Telecom churn prediction using cnn with variational autoencoder, in: Smart Systems: Innovations in Computing: Proceedings of SSIC 2021, Springer, 2022, pp. 583–600.
[30] F.A. Mohamed, A.K. Al-Khalifa, A review of machine learning methods for predicting churn in the telecom sector, in: 2023 International Conference on Cyber Management and Engineering (CyMaEn), IEEE, 2023, pp. 164–170.
[31] S. Saha, C. Saha, M.M. Haque, M.G.R. Alam, A. Talukder, Churnnet: deep learning enhanced customer churn prediction in telecommunication industry, IEEE Access (2024).
[32] C. Wang, C. Rao, F. Hu, X. Xiao, M. Goh, Risk assessment of customer churn in telco using fclcnn-lstm model, Expert Syst. Appl. (2024) 123352.
[33] T. Gattermann-Itschert, U.W. Thonemann, Proactive customer retention management in a non-contractual b2b setting based on churn prediction with random forests, Ind. Mark. Manage. 107 (2022) 134–147.
[34] N. Jajam, N.P. Challa, K.S. Prasanna, V.S.D. Ch, Arithmetic optimization with ensemble deep learning sblstm-rnn-igsa model for customer churn prediction, IEEE Access (2023).
[35] Z. Wu, L. Jing, B. Wu, L. Jin, A pca-adaboost model for e-commerce customer churn prediction, Ann. Oper. Res. (2022) 1–18.
[36] K. Mamta, S. Sangwan, Aapidl: an ensemble deep learning-based predictive framework for analyzing customer behaviour and enhancing sales in e-commerce systems, Int. J. Inf. Technol. (2024) 1–7.