scientific reports

OPEN



MBLSTM is a contextual interaction refined method for time series prediction

Weihan Qiu^{1,5}, Fangjie Zhu², Tianyong Hao¹, Maojie Wang^{2,3,4} & Runyue Huang^{2,3,4}

Time series prediction has been widely used in the medical field to predict patient recurrence or physiological fluctuations. However, the adequacy of the existing methods for contextual information interaction is still insufficient when dealing with a longer memory need in clinical data modelling. In order to enhance the utilization of memory interaction, this paper introduces a new contextual interaction refinement method MB-LSTM by incorporating a Hidden Layer Information Interaction Intensifier. The MB-LSTM method allows for simultaneous interaction of input and hidden layer states at each time step to enhance capability of capturing complex temporal relationships. Besides, more features of time series data are learned utilizing contrastive learning and a data augmentation scheme based on Kernel Density Estimation is designed to identify more accurate features from time series data. The method is evaluated on a real clinical dataset including 1053 records of patient with Gouty arthritis from the Guangdong Provincial Traditional Chinese Medicine Hospital by predicting the subsequent status of patients. The results show the proposed method achieves state-of-the-art performance by 0.5–7.2% using four different evaluation metrics compared with baseline methods.

Keywords Gouty arthritis, Time series prediction, Data augmentation, Contrastive learning

Gouty arthritis is a common form of inflammatory disease that is aroused by the sedimentation of Monosodium Urate (MSU) in joints or soft tissues, directly related to hyperuricemia caused by the disorders of purine metabolism¹. Gouty arthritis patients frequently experience repeated redness, pain, heat, joint malformations and tophus causing by elevated levels of uric acid in the blood, and have a high likelihood of acquiring various comorbidities due to hyperuricemia, leading to joint destruction, kidney damage and cardiovascular system injury over time². The current treatment of gout disease is unsatisfactory, as only 34.9% of patients meet the goal of Treat-to-Target (T2 T) of Serum Urate (SUA)³. Despite advancements in understanding the pathophysiology of gout, managing its episodic flare-ups remains a challenge for clinicians and patients themselves. Therefore, more therapeutic interventions need to be developed to meet different requests of individuals. In order to assist clinicians to make medical decisions for patient treatment, understanding and predicting the patterns of gout recurrence status has become a critical area of gouty arthritis research, giving a psychological expectation of the disease for the patients. An accurate prediction model for the disease recurrence based on the time series patient data would be advantageous for clinicians to proactively devise treatment plans and expedite patient recovery as swiftly as possible⁴.

Time series prediction has been a pivotal method for analyzing ordered data and making predictions based on the information hidden in the former historical data^{5,6}. These methods are particularly adept at identifying patterns in complex, non-linear data, which is often the case with medical records and clinical index^{7,8}. In recent years, machine learning has been applied to predict the patient states of physiology for these clinical indexes in the time series manner^{9,10}. With the development of artificial intelligence, an increasing number of neural networks, such as convolutional neural networks (CNN) and recurrent neural networks (RNN), are being utilized to learn complex data representations and show great potential in predictive tasks^{11,12}. Many neural network models have demonstrated remarkable performance for applications in the medical and healthcare research, improving the performance of recurrent neural networks (RNN) in the representation and processing

¹School of Computer Science, South China Normal University, Guangzhou, China. ²Second Clinical Medical College, Guangzhou University of Chinese Medicine, Guangzhou, China. ³Guangdong Provincial Key Laboratory of Clinical Research on Traditional Chinese Medicine Syndrome, Guangzhou, China. ⁴State Key Laboratory of Dampness Syndrome of Chinese Medicine, The Second Affiliated Hospital of Guangzhou University of Chinese Medicine, Guangzhou, China. ⁵Clinical Research and Big Data Laboratory, South China Research Center for Acupuncture and Moxibustion, Guangzhou University of Chinese Medicine, Guangzhou, China. [⊠]email: haoty@m.scnu.edu.cn; maojiewang@gzucm.edu.cn in time series manner¹³⁻¹⁷. The application of these sophisticated methods to gout recurrence prediction holds the potential to revolutionize patient care by enabling preemptive treatment strategies and personalized management $plans^{18-21}$.

The Long Short-Term Memory (LSTM) networks²² has achieved advanced results in many time series prediction tasks, including speech recognition and myoelectric motion decoding²³. More recently, Bidirectional Long Short-Term Memory (Bi-LSTM) networks²⁴ have gained attention in the medical domain due to their ability to capture context from both past and future time points within a sequence²⁵. This dual-directional method allows Bi-LSTM to provide a more comprehensive understanding of the temporal dynamics present in medical data. The ability of Bi-LSTM to handle the intricacies of sequential medical data, along with the incorporation of advanced techniques, paves the way for developing robust models in early diagnosis, treatment planning, clinical decision-making, and personalized medicine assistant. Despite the success, LSTM and Bi-LSTM methods still have some structural flaws in context interaction and effective feature extraction when dealing with highly contextually relevant data that require stronger information interaction, which is prevalent in medical time series data. These methods only update the next unit based on the single-gated interaction information between the hidden layer and the input layer when dealing with the contextual interaction of the time series data.

To that end, this paper proposes the Memory Interaction Boosting Bidirectional Long Short-Term Memory (MB-LSTM), a contextual interaction refined neural network model for time series prediction, based on the idea of paying equal attention to past and future information, and strengthening the interaction of context information. The enhanced representational capacity of MB-LSTM improves medical time series data modeling, where capturing intricate temporal patterns can be vital for patient care. Different from the previous models, this paper proposes a hidden layer information interaction intensifier (Hii) as a core component that allows for simultaneous interaction of the input, the hidden layer state and the recurrent connections at each time step, thereby enhancing ability in capturing complex temporal relations and improving interaction degree. The intensifier is embedded in each unit of the Bi-LSTM model. After embedding, the input and output sequences of the Bi-LSTM model is adjusted according to the intensifier. In addition, this paper designed a data augmentation process according to the characteristics of experiment data, using Kernel Density Estimation (KDE) to determine the data distribution of each patient, and partially enhance the data according to the distribution.

In summary, the main contributions of this paper lie in the three aspects:

1) A new contextual interaction refined method named MB-LSTM featured with a hidden layer information interaction intensifier is proposed to improve the performance of clinical time series data prediction.

2) A data augmentation scheme is designed by applying KDE to determine the distribution of indicator data and to enhance data to uniform step size for all data samples.

3) A contrastive learning method is used to enlarge the distance between positive and negative samples while reduce the distance between samples of the same label.

Related work

Existing work on time series prediction can be broadly classified into three types: (1) traditional statistical -based methods, (2) machine learning -based methods, and (3) deep learning -based methods.

The Auto Regressive Integrated Moving Average (ARIMA) model²⁶ remains a cornerstone in the field of time series predicting due to its flexibility and robustness. In medical research, ARIMA models have been employed to predict the incidence of diseases, hospital admissions, and demand for medical services, aiding in resource allocation and public health planning. During COVID-19 pandemic, ARIMA models have been instrumental in predicting the spread of the virus, assessing the impact of interventions, and planning healthcare resources²⁷. Moreover, hybrid models incorporating ARIMA and machine learning methods have been proposed to tackle non-linear patterns and enhance the predicting accuracy in medical time series data²⁸, outperforming ARIMA when dealing with complex datasets that exhibit irregular trends or seasonal effects. Traditional methods have proven to be valuable for time series predicting in the medical field, but these methods often fall short when dealing with the complexity and non-linearity of medical data.

A variety of machine learning methods are included in the field of time series predicting. Support Vector Machines (SVM)²⁹ has been widely recognized for the robustness in high-dimensional spaces and the capacity to model nonlinear relationships through kernel functions. In medical time series predicting, SVM has been used to predict patient outcomes and disease progression³⁰. Random Forest (RF) is another machine learning method for time series prediction³¹, known for its ability to handle large datasets with numerous input variables without overfitting. RF has been successfully applied to forecast the incidence of infectious diseases, outperforming traditional time series models³². Gradient Boosting Machines (GBM)³³ has gained popularity due to predictive accuracy and flexibility. The mistakes of weak learners are iteratively corrected by GBM, converted to be suitable for medical time series data. GBM has been used to predict hospital readmission rates with high accuracy³⁴. Ongoing research continues to explore the synergies between these machine learning methods to develop more accurate and reliable predicting models, aiming at enhancing patient care and healthcare management.

The LSTM architecture addresses the vanishing gradient problem commonly encountered with traditional Recurrent Neural Networks (RNN), making it a robust choice for modeling complex time series data. LSTM is capable of learning long-term dependencies, and has been applied to predict patient diagnoses, treatment outcomes, electronic health records (EHR), and disease progression with promising results³⁵. The correlation of past events has been required taking into account among the task of making predictions in the field of medicine³⁶. Studies on medication adherence confirmed a strong relationship between time series predictions and patients' clinical history events^{37,38}. Traditional LSTM has a limitation in only learning from previous time steps, known as forward states. Bi-LSTM networks extend the capabilities of traditional LSTM by processing the data in both forward and backward directions, allowing it to access both states simultaneously. Bi-LSTM is allowed

by this dual-direction processing to capture information from both the past and the future, making Bi-LSTM particularly well-suited for tasks where understanding the entire sequence is important, such as natural language processing, speech recognition, and time series prediction. This bidirectional structure has shown to be useful in applications where the context in both directions is important for prediction, such as in the case of medical event detection in time series data³⁹. Many deep learning models are improved upon the LSTM and Bi-LSTM, and Mogrifier LSTM is one of them⁴⁰. The Mogrifier LSTM applies a series of transformations that "mogrify" the input and hidden states by iteratively mixing information between them before the standard LSTM computations. Zhang et al.⁴¹ has demonstrated the potential of advanced LSTM variants, including the Mogrifier LSTM and attention network in predicting patient outcomes. A study by Che et al.⁴² applied the Mogrifier LSTM to predict the onset of sepsis in ICU patients, showing that the model outperformed traditional LSTM and GRU models in terms of predictive accuracy and timeliness. The new time series prediction methods alleviates the problem of long sequence dependence of LSTM methods, but increases the runtime overhead, and some of the methods are improved in this way. Simply Attend and Diagnose (SAnD)⁴³ proposed a masked self-attention mechanism, using positional coding and dense interpolation strategies to incorporate the time sequence and improve the model effect. The AdaCare⁴⁴ model is one of the best-performing methods in recent years to maintain state-ofthe-art performance in prediction accuracy by capturing long and short-term variations of biomarkers as clinical features and establishing a correlation model. In conclusion, most of the current improvement goals of time series prediction methods focus on improving prediction accuracy and reducing operational overhead, with less work on optimizing the degree of contextual interaction. Therefore, our method will make improvements targeting information interaction.

Methods The MB-LSTM model

The main structure of MB-LSTM model consists of two parts: The Bi-LSTM and the Hidden layer information interaction intensifier (Hii). The Bi-LSTM is served as the fundamental model that complete the task of model training and learning of MB-LSTM. The Hii is embedded as an enhancement for extracted features and information, as the core module of MB-LSTM. The architecture of MB-LSTM is shown in Fig. 1.

MB-LSTM processes the input of two hidden layers in two standard LSTM layers at the same time. Since there is only one input, the interaction between the input and the hidden layers is no longer one-to-one or alternating. The cross-information interaction structure proposed in the Mogrifier LSTM cannot be used, but the interaction information in both directions is equal through reasonable design⁴⁵. An effective way to maintain information equivalence in both directions is to process the information in both directions while executing the







Fig. 2. The structure of a single unit of MB-LSTM.



interactive components at the same time. In this study, a method that refined to function in parallel with each LSTM unit is proposed. Combining with a single unit of Hii and two LSTM units in the meantime, the structure of the interaction module between two LSTM units at the same time of the training is shown in Fig. 2.

 hf_{t-1} is the hidden layer state of the previous unit in the forward LSTM layer, and hb_{t+1} is the hidden layer state of the previous unit in the backward LSTM layer. Different from Bi-LSTM, the current input and two hidden layer states are sent to the Hii first. A sufficient information exchange of the current input with the hidden layer states are interacted in the Hii. Then the fully interacted information of the same dimension is imported into the forward LSTM layer and backward LSTM layer. Finally, the output and the hidden layer state are formed respectively after two standard LSTM units. hf_t and hb_t are served as the output of the current unit as well as the input to the next MB-LSTM unit. The number of the MB-LSTM units is also part of the hyper parameters of the model.

(1) The Bi-LSTM: A Bi-LSTM consists of two standard LSTM layers that are applied in parallel: a forward LSTM layer that processes the sequence from start to end, and a backward LSTM layer that processes the sequence from end to start. The outputs of both layers are typically concatenated at each time step, providing a combined representation that contains information from both directions. This concatenated output can be used to make predictions. The architecture of a Bi-LSTM is inherently more complex than a unidirectional LSTM due to the double LSTM units. This complexity comes with the advantage of a richer representation of the time series data, supporting the Bi-LSTM making a more accurate prediction. MB-LSTM follows the bi-directional architecture, but appropriately changes the direction of the information interaction. The inputs and outputs no longer pass directly through the LSTM cells, thus increasing the degree of information interaction. The brief structure of Bi-LSTM is show in Fig. 3.

The Hidden layer information interaction intensifier: The main objective of Hii is to further extract and utilize the information of the hidden layer, aiming at solving the long-term dependence problem and making full use of the features of each moment. Hii does not disturb the normal operation of the Bi-LSTM. The location of this intensifier between the LSTM units ensures that receiving equal attention of the contextual information is unaffected. The embedded intensifier consists of several interaction units laid end to end. The information of the input and the hidden layer states are fully exchanged through each interaction unit.

The brief structure of a single interaction unit in the Hii is shown in Fig. 4. The interaction unit consists of four steps. The Input X_0 first is interacted with the hidden layer information of the forward LSTM layer HF_0 to generate a new X_1 state. The dimension of state X_1 is served as a transition state and is unified with the input dimension and the hidden layer dimension. The newly formed state X_1 then interacts with the hidden layer information of the backward LSTM layer HB_0 to generate a new hidden layer state HB_2 , which is then served as the interacted hidden layer state of the backward LSTM layer. Similarly, the transition state X_1 then interacts with HF_0 to generate the forward LSTM layer. Finally, after the updated hidden layer state HB_2 interacts with state X_1 again, updating the input to X_2 . Afterwards, the status update of a single interaction unit is completed. The key equation of the interaction unit in Hii can be summarized as Eq. (1) - Eq. (4). σ is the Sigmoid function, W and b are parameters of the fully connected layer.

$$X_1 = \sigma(W(HF_0 + X_0) + b)$$
(1)

$$HB_2 = \sigma(W(HB_0 + X_1) + b) \tag{2}$$

$$HF_2 = \sigma(W(HF_0 + X_1) + b)$$
(3)

$$X_2 = \sigma(W(HB_2 + X_1) + b) \tag{4}$$

After the end of the current interaction unit, X_2 , HF_2 and HB_2 join the next interaction unit for continuous interaction and updating. The number of the iteration of the interaction units is also a hyperparameter. With countable number of these interaction units sequentially, Hii accomplish the task of strengthening the interaction of context information.

Contrastive learning

Contrastive learning is a self-supervised learning technique with significant traction in the field of machine learning for its ability to learn powerful representations and to contrast positive pairs against negative pairs. The



Fig. 4. A single interaction unit in the Hidden layer information interaction intensifier.

Scientific Reports | (2025) 15:18563

contrastive loss function⁴⁶, is central to contrastive learning method. The contrast loss function is embedded in the training process of the model and performs as a component used to optimize the model. It measures the difference between the predictions of the model and the actual results, and is used to adjust the weight of the model to minimize this difference. The selection of the loss function in contrast learning is based on the current training task and data characteristics. Since the prediction with classified labels requires the distinguish of positive and negative samples of patient records, the Triplet Loss⁴⁷ is selected as loss function. The basic idea of the Triplet Loss is to structure the embedding space in such a manner that the distance between similar data points is minimized while simultaneously maximizing the distance between dissimilar points, thereby ensuring a margin of separation. The formal definition of the Triplet Loss is given by the following Eq. (5):

$$L = \max \left\{ D(a, p) - D(a, n) + margin, 0 \right\}$$
(5)

As shown in Eq. (5), *a* denotes the anchor sample, *p* denotes the positive sample, and *n* denotes the negative sample. D(x, y) denotes the distance between the embedding of samples *x* and *y*, and *margin* is a hyper parameter that specifies the desired separation between the positive and negative pairs. In this study the *margin* value is standardized at 0.5. The function ensures that the loss is non-negative, and the loss is zero when the negative example is farther from the anchor than the positive example. This loss function encourages positive pairs to be similar and encourages negative pairs to be dissimilar simultaneously, which is essential for learning a robust metric and leading to a more structured and discriminative feature space.

The KDE-based data augmentation

The KDE-based data augmentation (KDEda) method is proposed for the time series prediction with imbalanced data. The number of follow-up visits of each patient is not the same, which cannot be used for unified training and learning. In order to maintain the same sequence length for each patient during training, it is necessary to enhance the time series patient data without a certain length. To solve this problem, Kernel Density Estimation (KDE)⁴⁸ is used to infer the distribution of data based on limited samples⁴⁹.

The utilization of KDE is formularized as follows: *n* observations are taken from a population *X* to form the series $\{x_i, x_2, ..., x_n\}$. The series are independent and identically distributed. The probability distribution function f(x) remains unknown. $K(x_i, t)$ is called a kernel function with $0 \le K(x_i, t) < \infty$ and it is non-negative and bounded for all real *x* and *t*. The Kernel estimate $\hat{f}(x)$ assigns each *i*-th sample data point x_i a kernel function:

$$\hat{f}(t) = \frac{1}{n} \sum_{i=1}^{n} K(x_i, t)$$
(6)

The most frequently used form of the kernel function is allowed under the symmetry property in most common practical applications, in which parameter *h* called bandwidth:

$$K_{sym}(x,t) = \frac{1}{h} K(\frac{x-t}{h})$$
(7)

The choice of kernel function and bandwidth plays an important role in the estimation. The expression of kernel density curve cannot be too smooth or too delicate, which may cause the ignorance of the important features or revealing unnecessary details.

After comparing the combination of four different kernel functions and two bandwidths for each column of data, the probability density function curve image under the Expanechnikov kernel function⁵⁰ with the Silverman bandwidth⁵¹ is the most consistent combination. This combination is therefore chosen as the enhancement method during data augmentation.

The Epanechnikov kernel is a quadratic kernel function commonly used in kernel density estimation due to its optimal balance between bias and variance. This parabolic shaped kernel is bounded and has a compact support, which means it only affects points within a unit distance, reducing computational cost. The favorable properties of the Expanechnikov kernel include being one of the least-variance unbiased estimators within the class of kernels with finite support, leading to its widespread adoption in non-parametric density estimation. The definition of the Expanechnikov kernel function is shown in Eq. (8). *u* denotes the normalized relative distance between the target point and the sample point.

$$K(u) = \begin{cases} \frac{3}{4}(1-u^2), |u| \le 1\\ 0, otherwise \end{cases}$$
(8)

The Silverman bandwidth, also known as Silverman's rule of thumb, is a heuristic for selecting the bandwidth in kernel density estimation, providing an analytical method to determine the smoothing parameter. The Silverman bandwidth is derived under the assumption that the data follows a normal distribution and aims to minimize the mean integrated squared error (MISE), offering a practical and widely-used initial estimate for the bandwidth in non-parametric density estimation. The formula of the Silverman bandwidth *h* is shown in Eq. (9), where $\hat{\sigma}$ represents the standard deviation of the sample and *n* is the number of observations.

$$h = \left(\frac{4\hat{\sigma}^5}{3n}\right)^{1/5} \approx 1.06\hat{\sigma}n^{-1/5} \tag{9}$$

During the data augmentation operation, each column of data for each patient is enhanced according to the distribution that the selected kernel function and bandwidth after comparison. The selected distribution of data enhancement for each column is Epanechnikov kernel and Silverman bandwidth. The data augmentation operation in KDEda aims to unify the jagged follow-up records of patients into a new data set. The vacant data are enhanced based on patient follow-up data. Each patient's data set was processed and analyzed independently of the distribution to ensure that data from individual patients did not affect each other. To align with the original 36-week follow-up data collection plan requirements, we augmented each patient with fewer than 9 follow-up data to 9, augmenting the data with the missing portion of the 9 follow-up records for that patient. After the data augmentation operation, each patient has the same number of the follow-up record in the new data set suitable for model training.

The overall architecture

The proposed method is based on patient data with multiple follow-up records and predicts the subsequent recurrence status of patients. The method consists of three components including data pre-processing, KDE-based data augmentation (KDEda) and MB-LSTM model. The input data is firstly grouped according to labels based on the original classification, and is further pre-processed through the KDEda to unify the record length for each patient. The enhanced time series data is fed into the MB-LSTM, where the training and the learning procedure are proceeded for the time series prediction. The main target of MB-LSTM is solving the long-term dependence problem while making full use of the characteristics of the information in the hidden layer of each moment. Finally, Contrastive learning is also embedded to learn more accurate features and improve the performance of the model. The overall architecture of the time series prediction based on MB-LSTM is show in Fig. 5.

Experiment setups Dataset

Since there is no publicly available real dataset for the time series-based gouty prediction task, we collect data from Guangdong Provincial Hospital of Chinese Medicine from 2020 to 2023. The dataset contains real patients diagnosed with gouty arthritis and has a relatively complete time series relationship. The studies involving humans were approved by Ethics Committee of Guangdong Provincial Hospital of Chinese Medicine No: BF2020-193-01. The studies were conducted in accordance with the local legislation and institutional requirements. The participants provided their written informed consent to participate in this study. After pre-processing, the dataset contains 1053 valid data of serum biomarkers from 160 patients. Data Masking (hiding the name and



Fig. 5. The overall architecture of the time series prediction based on MB-LSTM.

Scientific Reports | (2025) 15:18563

medical card number of the patients) has been performed when the data is obtained, and the acquisition and processing of the data does not violate any ethical principles. This dataset has been used in other gouty time series prediction work [43] previously.

The dataset mainly consists of two parts. The first part is the masked serum biomarker data of patients. Each piece of data is the result of one examination for one patient. The serum biomarker data includes Fasting blood glucose (FBG), alanine aminotransferase (ALT), aspartate aminotransferase (AST), Urea, serum creatinine (Cr), uric acid (UA), Triglycerides, Total cholesterol, high density lipoprotein cholesterol (HDL-C), low density lipoprotein cholesterol (LDL-C), creatine kinase (CK), creatine kinase isoenzyme (CK-MB), lactate dehydrogenase (LDH), glomerular filtration rate (eGFR) and hypersensitive C-reactive protein (hCRP). Each reserved patient has at least two reviews arranged chronologically by the examine time to form the time series data. The distribution of patients with different data strips in the serum biomarker data is shown in Fig. 6. It can be seen from the figure that most of the patient data in this dataset are 7 or more, which can satisfy the prediction of the recurrence status of a single patient. Each piece of data for the patient will be considered as a step during training.

The second part of the dataset is the patient recurrence status data. The recurrence status data are matched with the masked serum biomarker data at the corresponding time period with labeled records. Two kinds of labels are displayed, "Interval Period" and "Acute Phase". The experiment data are the combination of these two original datasets. All the time series data are used for the construction and prediction of the MB-LSTM model.

Evaluation metrics

The main evaluation metrics used to evaluate the performance of models are Accuracy, Precision, Recall, and F1-score. Accuracy is the proportion of correct predictions among the total number of tested samples. Precision is the proportion of true positive predictions in the total predicted positive samples. Recall is the proportion of actual positive samples correctly identified by the model. The F1-score is the harmonic average of the precision and recall. Accuracy, Precision, Recall, and F1-score are respectively defined in Eq. (10) to Eq. (13). TP is the number of positive samples predicted to be positive, and FN is the number of positive samples predicted to be negative. FP is the number of negative samples predicted to be positive, and TN is the number of negative samples predicted to be negative.

$$Accuracy = \frac{TP + TN}{TP + TN + TP + FN}$$
(10)

$$Precision = \frac{TP}{TP + FP} \tag{11}$$

$$Recall = \frac{TP}{TP + FN}$$
(12)

$$F1 - score = \frac{2 * (Precision * Recall)}{(Precision + Recall)}$$
(13)

Data pre-processing and parameter setting

The data pre-processing steps include: removing misinput data, blank data, converting all data types to be consistent, and eliminating patients' data that display no time series peculiarity. Misinput data and the blank data can lead the neural network to learn features that do not correspond to the attributes of the data during



19. 0. Humber of putent records of the set and biomarker data.

model training. Data without time series characteristics cannot construct any time series data and should be eliminated.

The parameter setting of MB-LSTM is shown in Table 1. The relevant parameters show the experimental process that balances runtime and overhead while getting better results. The values of each parameter in the experiment were obtained by comparing the effects of several experiments. The values of each parameter in the experiment were obtained by comparing the effects of several comparative experiments. The model performs better on the dataset when epoch is 300 and batchsize is 30. The initial learning rate is set to 0.001 and is adjusted in training with the adaptive adjustment mechanism of the AdamW optimizer. The number of iterations in the Hii is set to 2. The sequence length is set to 9 to accommodate the number of records for most patients. Patients with less than 9 records are enhanced to 9 in the data augmentation part, and only the first 9 records are considered valid for patients with more than 9 records. The 9 records correspond to the three-month follow-up records of the patients. The dataset was divided into training, validation and test sets in a 7:2:1 ratio to ensure clinical generalizability of the model assessment.

Baselines

In the experiments, the proposed MB-LSTM is compared with several model that share part of the similar insights with us. The baseline methods in this paper are the most widely used in the field. Each of the baseline methods has been used as a SOTA method in gout-related prediction tasks or chronic disease progression prediction-related tasks. Two of the methods, SAnD and AdaCare, are newer methods in time-series prediction of classification research, and they have improved over the baseline methods in modeling effectiveness and operational overhead. The baseline models are listed as follows.

Support vector machines (SVM): A supervised machine learning algorithm used for classification analysis. **Gradient boosting machines (GBM)**: Builds an ensemble of decision trees sequentially where each tree corrects the errors of its predecessor, leading to a predictive model.

Long Short-Term memory (LSTM): A RNN designed to effectively capture long-term dependencies in sequential data by utilizing specialized memory cells and gating mechanisms.

Bidirectional long Short-Term memory (Bi-LSTM): An extension of LSTM that processes sequential data in both forward and backward directions simultaneously, allowing the model to capture contextual information from past and future time steps.

Mogrifier LSTM: Introduces gating mechanisms to enable direct interactions between hidden states, facilitating more efficient information flow and improving model performance in capturing long-range dependencies.

Simply attend and diagnose (SAnD): A deep learning model that combines attention mechanisms with diagnostic reasoning to effectively interpret medical time series data for disease diagnosis.

AdaCare: A novel method leverages advanced algorithms to dynamically adjust care plans based on individual patient responses and health data, holds promise for enhancing the effectiveness and efficiency of healthcare delivery.

Results

Comparative performance against baseline models

In order to test the effectiveness of the proposed MB-LSTM model, all the baseline methods are implemented for the performance comparison on all the evaluation metrics. In order to verify the effectiveness of KDEda and contrastive learning (Cl) embedded in the model, a performance comparison of the model with or without the embedding of KDEda and Cl is also set respectively. Each method is tested to compare the performance of our model in contrast study. As shown in Table 2, our method ranks high in all the four evaluation metrics. Our model achieves 0.728 of Accuracy, 0.901 of Precision, 0.776 of Recall and 0.836 of F1-score on the dataset. Compared with the baseline methods, our model achieves better performance in Accuracy, Recall and F1-score. In terms of Precision performance, our model also ranks among the best. Compared to the latest baseline model, our model improves by 0.004, 0.009, 0.003, and 0.017 in each of the four metrics. The previously proposed Mogrifier LSTM method is superior in Precision, and AdaCare performs slightly behind our method.

In order to validate the effect of KDEda on the effectiveness of each model, a comparison is designed to show the impact of KDEda as a data enhancement strategy. As shown in Table 3, the overall effects of four evaluation metrics are improved in all baseline methods after adding the process of the KDEda. In this experiment, the performance of our model is not the best, but still ranks among the top two. The performances of accuracy,

Parameters	Values
Epoch	300
Dropout	0.2
Batch size	30
Learning rate	0.001
Number of iterations	2
Sequence length	9
Data set segmentation	7:2:1

Table 1. Parameters setting of our model.

Methods	Accuracy	Precision	Recall	F1-score
SVM	0.572	0.889	0.596	0.714
GBM	0.639	0.900	0.671	0.769
LSTM	0.654	0.838	0.679	0.658
Bi-LSTM	0.671	0.748	0.719	0.682
Mogrifier LSTM	0.722	0.911	0.764	0.831
SAnD	0.705	0.884	0.752	0.768
AdaCare	0.724	0.892	0.773	0.819
MB-LSTM	0.728	0.901	0.776	0.836

Table 2. The performance comparison of MB-LSTM with the baseline methods.

Methods	Accuracy	Precision	Recall	F1-score
SVM + KDEda	0.667	0.878	0.714	0.788
GBM + KDEda	0.656	0.872	0.720	0.789
LSTM + KDEda	0.656	0.876	0.720	0.795
Bi-LSTM + KDEda	0.678	0.926	0.696	0.794
Mogrifier LSTM + KDEda	0.728	0.889	0.795	0.839
SAnD + KDEda	0.722	0.902	0.815	0.811
AdaCare + KDEda	0.724	0.913	0.808	0.83
MB-LSTM + KDEda	0.728	0.894	0.789	0.838



Methods	Accuracy	Precision	Recall	F1-score
SVM + Cl	0.661	0.885	0.714	0.790
GBM + Cl	0.667	0.885	0.720	0.795
LSTM + Cl	0.694	0.908	0.733	0.811
Bi-LSTM + Cl	0.722	0.899	0.776	0.834
Mogrifier LSTM + Cl	0.756	0.759	0.736	0.745
SAnD + Cl	0.749	0.88	0.797	0.825
AdaCare + Cl	0.753	0.898	0.801	0.842
MB-LSTM + Cl	0.750	0.897	0.814	0.853

Table 4. The performance comparison of MB-LSTM with the baseline methods in Cl.

precision, recall, and F1-score of our model are 0.728, 0.894, 0.789, and 0.838 respectively. Bi-LSTM, sand and mogrifier LSTM perform best in precision, recall and F1-score respectively.

In order to validate the effect of contrastive learning on the effectiveness of each model, a comparison is designed to show the impact of Cl as a loss optimization strategy. As shown in Table 4, with the addition of Cl, the performance of all models is improved, especially the accuracy. The performances of accuracy, precision, recall, and F1-score of our model MB-LSTM + Cl are 0.750, 0.897, 0.814, and 0.853 respectively. Our model obtains the best scores on recall and F1 with 0.013 and 0.011 higher than the second ranked method respectively. Compared with the performance without Cl, the accuracy, recall, and F1-score of our model have been improved by 0.022, 0.038, and 0.017, respectively.

When the KDEda and Cl are applied at the same time, the performance of our model has more advantage among all improvements of the tested methods. As shown in Table 5, our model achieves 0.783 of accuracy, 0.918 of precision, 0.857 of recall and 0.876 of F1-score on the dataset, ranking at the first among all baseline methods. Meanwhile, a noteworthy improvement is revealed compared with the results of using the MB-LSTM model alone. With the addition of the KDEda and Cl, the accuracy, precision, recall, F1-score of our model has been improved by 0.055, 0.017, 0.081, and 0.040 respectively. In addition, the results of our model in which both optimization strategies are used are improved over the results of our model in which only a single strategy is used. Comparing to the case where only KDEda is used, our model effect improves 0.055, 0.024, 0.068, and 0.038 in each of the four metrics. Comparing to the case where only Cl is used, our model effect improves 0.033, 0.021, 0.043, and 0.023. The results show that our model is more effective than the baseline method when used alone and combined with the optimization method. This is confirmed by the improvement of various evaluation metrics and the gap with other baseline methods.

Methods	Accuracy	Precision	Recall	F1-score
SVM + KDEda + Cl	0.701	0.896	0.752	0.818
GBM + KDEda + Cl	0.731	0.705	0.694	0.749
LSTM + KDEda + Cl	0.728	0.894	0.789	0.838
Bi-LSTM + KDEda + Cl	0.756	0.896	0.820	0.857
Mogrifier LSTM + KDEda + Cl	0.772	0.889	0.851	0.870
SAnD + KDEda + Cl	0.779	0.913	0.822	0.874
AdaCare + KDEda + Cl	0.78	0.918	0.838	0.869
MB-LSTM + KDEda + Cl	0.783	0.918	0.857	0.876





Fig. 7. The performance improvement comparison of MB-LSTM model and other baseline methods with different optimization methods in the experiment. (**a**) Performance improvement of the Accuracy. (**b**) Performance improvement of the Precision. (**c**) Performance improvement of the Recall. (**d**) Performance improvement of the F1-score.

Metric-Specific improvement analysis

As shown in the Fig. 7, the comparison of the improvement effect of each evaluation metrics between the models with KDEda and contrastive learning is shown on the (a) to (d). (a) shows the improvement of Accuracy, indicting the improvement of each baseline method with KDEda and Cl. When KDEda and Cl are used at the same time, the accuracy improvement of each method varies from 0.050 to 0.129. (b) shows the improvement on Precision, in which the most significantly improved part lies on LSTM and Bi-LSTM models. (c) and (d) show the improvement of Recall and F1-score. These two metrics of almost all methods have been improved, and the average amplitude of improvement is higher than that of accuracy. Among all methods, Recall has a largest increase of 0.156, and F1-score has a largest increase of 0.153. Performance on all evaluation metrics reveal improvement after using the two optimization methods no matter machine learning methods or deep learning models. It is worth noting that the evaluation metrics of the Mogrifier LSTM show a slight decrease in all three



Fig. 8. The performance comparison for different dataset sizes. (a)The performance of MB-LSTM. (b)The performance of Mogrifier LSTM.

Sequence length	Accuracy	Precision	Recall	F1-score
7	0.598	0.793	0.860	0.835
8	0.643	0.887	0.863	0.878
9	0.783	0.918	0.857	0.876
10	0.774	0.909	0.833	0.846
11	0.792	0.897	0.874	0.825
12	0.787	0.901	0.816	0.824

 Table 6. The performance comparison of different sequence length.

plots, (b), (c), and (d), and the major part of the decrease is concentrated in the use of Cl. The reason is that the Mogrifier LSTM method has already been designed to take into account the distances between similar data points, and thus further processing with Cl will lead to model overfitting and decrease the model performance. Our model is not the best in the comparison since our model has a better underlying performance without any optimization methods. The reason why our model is not optimal when using KDEda and Cl alone, but is optimal when used together, is that KDEda reduces the risk of overfitting due to noise and variability, while Cl helps the model to focus on the relevant features in the noise.

Robustness to training data scale

To further analyze the prediction performance of our model in dealing with different sizes of the dataset, experiments with different sizes of training sets are conducted to estimate the performance of MB-LSTM and Mogrifier LSTM with KDEda and Cl. As shown in Fig. 8, our model has relatively stable prediction performance when the training set is more than 60% of the original training size. The scores difference of the four metrics compared with 100% of the dataset was 0.079 at most. When the training set size is reduced to 40% of the original training set, the performance of three metrics is significantly reduced. The performance of Morgrifier LSTM under the same condition of the training set is shown on (b). Many of the evaluation metrics of Mogrifier LSTM are slightly inferior to our model at the same training set size. Even when the training set size is only 20% of the original data set, the performance in the evaluation metrics of MB-LSTM are 0.043, 0.082, 0.048, and 0.051 higher than Mogrifier LSTM respectively.

Discussion

We investigate the effect of the difference of sequence lengths in KDEda for the gouty arthritis patient data. The results are shown in Table 6. When the sequence length is 8 or lower, Accuracy drops much compared to the sequence length of 9 although Recall and F1-score increase slightly. The reason for the poor performance is that the reduction of sequence length has an impact on the effective feature extraction for each epoch, ultimately resulting in poorer prediction performance. When the sequence length is 10 to 12, the overall prediction performance is about the same as sequence length 9, but F1-score is higher with the sequence length 9. The reason for the lower performance for sequence length 10 to 12 is that most patient data in the original data set are no more than 9 records, and excessive data augmentation causes the real features in the data to be covered

up, which ends up with lower performance in F1-score. When the sequence length exceeds 12, the running time of the model is too long and the running cost is too high.

We explore the effect of the number of iterations of the Hidden layer information interaction intensifier (Hii) in MB-LSTM. The number of iterations of Hii mainly affects the number of interactions between hidden layers and inputs in the model. When the number of iterations is too small, the extracted features are not effective, and when the number of iterations is too large, the model may be overfitting, both of which may reduce the prediction performance. In the experiment on the gouty arthritis dataset, the model may be overfit when the number of iterations of Hii exceed 4. When the number of iterations of Hii is 3 and 4, the overall performance of the model is not significantly improved compared with that when the number of iterations is 2. Considering that Hii is set in each unit in the MB-LSTM model, the number of iterations will rapidly increase the running time and the running cost of the model. Therefore, the number of iterations is set to 2.

The MB-LSTM method is different from other related methods, which are optimized from the aspects of training speed or occupying memory space. Instead, MB-LSTM improves the generalization ability of language modeling and provides a richer space for the interaction between input and context information. This improved idea is aimed at the processing and prediction of time series data, which makes our model more suitable for the field of time series data processing. In the medical domain, accurate time series prediction is crucial for effective diagnosis and treatment planning. Time series prediction can provide valuable insights into disease progression, treatment outcomes, and the overall management of the condition for gout patients. Healthcare providers can make informed decisions to improve the care and outcomes by analyzing the former data of the patient. Time series data can also help in predicting the onset of the acute phases and the response to different treatment regimens. This proactive approach allows for timely interventions treatment plans tailored to the individual needs of the patient, which has a promising future for personalized treatments for gout patients. The ability of MB-LSTM to effectively process and learn from complex time series data suggests the strong potential for other applications in healthcare time series analytics, such as predicting disease progression, optimizing treatment regimens, and managing healthcare resources.

Conclusions

This paper presents a contextual interaction refined model named MB-LSTM for time series prediction in gouty arthritis treatment. The model is charactered as the utilization of KDE-based data augmentation and the implementation of contrastive learning for advanced feature learning. Experiments evaluate the effectiveness of the proposed model by comparing with baseline methods. The results show that our model outperforms baseline methods in terms of four evaluation metrics, indicating that our model is more effective in handling medical time series prediction tasks.

Data availability

The following information was supplied regarding data availability: The code and the data is available at GitHub: https://github.com/Maple-NTF/MBLSTM.

Received: 10 January 2025; Accepted: 19 May 2025 Published online: 27 May 2025

References

- Dehlin, M., Jacobsson, L. & Roddy, E. Global epidemiology of gout: prevalence, incidence, treatment patterns and risk factors. *Nat. Rev. Rheumatol.* 16 (7), 380–390. https://doi.org/10.1038/s41584-020-0441-1 (2020).
- Liu, R. et al. Prevalence of hyperuricemia and gout in Mainland China from 2000 to 2014: a systematic review and meta-analysis. Biomed. Res. Int. 2015 (1), 762820. https://doi.org/10.1155/2015/762820 (2015).
- The, C. L., Cheong, Y. K., Wan, S. A. & Ling, G. R. Treat-to-target (T2T) of serum urate (SUA) in gout: a clinical audit in real-world gout patients. *Reumatismo* 71 (3), 154–159. https://doi.org/10.4081/reumatismo.2019.1225 (2019).
- Pungitore, S. & Subbian, V. Assessment of prediction tasks and time window selection in Temporal modeling of electronic health record data: a systematic review. *Journal Healthc. Inf. Research* 7 (3), 313–331. https://doi.org/10.1007/s41666-023-00143-4(2023).
- El-Sappagh, S., Alonso, J. M., Islam, S. M. R., Sultan, A. M. & Kwak, K. S. A multilayer multimodal detection and prediction model based on explainable artificial intelligence for Alzheimer's disease. *Sci. Rep.* 11 (1), 2660. https://doi.org/10.1038/s41598-021-8209 8-3 (2021).
- Krittanawong, C. et al. Machine learning prediction in cardiovascular diseases: a meta-analysis. Sci. Rep. 10 (1), 16057. https://doi .org/10.1038/s41598-020-72685-1 (2020).
- Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet classification with deep convolutional neural networks. *Commun. ACM.* 60 (6), 84–90. https://doi.org/10.1145/3065386 (2017).
- 8. Sutskever, I., Vinyals, O. & Le, Q. V. Sequence to Sequence Learning with Neural Networks. arXiv preprint arXiv:1409.3215 (2014).
- 9. Lipton, Z. C., Kale, D. C. & Wetzel, R. Modeling missing data in clinical time series with Rnns. Mach. Learn. Healthc. 56 (56), 253–270 (2016).
- Zheng, H. & Shi, D. Using a LSTM-RNN based deep learning framework for ICU mortality prediction. Web Information Systems and Applications: 15th International Conference, WISA Taiyuan, China, September 14–15, 2018, Proceedings 15. Springer International Publishing. 2018, 60–67. (2018). https://doi.org/10.1007/978-3-030-02934-0_6 (2018).
- Rajkomar, A. et al. Scalable and accurate deep learning with electronic health records. NPJ Digit. Med. 1 (1), 1–10. https://doi.org/ 10.1038/s41746-018-0029-1 (2018).
- 12. Suresh, H. et al. Clinical intervention prediction and Understanding with deep neural networks. *Mach. Learn. Healthc. Conf.* PMLR 2017, 322–337 (2017).
- 13. Aczon, M. et al. Dynamic mortality risk predictions in pediatric critical care using recurrent neural networks. *ArXiv Preprint* (2017). http://arxiv.org/abs/1701.06675.
- Choi, E., Bahadori, M. T., Schuetz, A., Stewart, W. F. & Sun, J. Doctor Ai: predicting clinical events via recurrent neural networks. Mach. Learn. Healthc. Conf. PMLR, 301–318 (2016). 2016.
- Harutyunyan, H., Khachatrian, H., Kale, D. C., Ver, S. G. & Galstyan, A. Multitask learning and benchmarking with clinical time series data. Sci. Data. 6 (1), 96. https://doi.org/10.1038/s41597-019-0103-9 (2019).

- Hou, J., Saad, S. & Omar, N. Enhancing traditional Chinese medical named entity recognition with Dyn-Att Net: a dynamic attention approach. *PeerJ Computer Science*. 10, e (2022). https://doi.org/10.7717/peerj-cs.2022 (2024).
- Zhang, M. et al. Hybrid mRMR and multi-objective particle swarm feature selection methods and application to metabolomics of traditional Chinese medicine. *PeerJ Computer Science*. 10, e (2073). https://doi.org/10.7717/peerj-cs.2073 (2024).
- 18. Eldele, E. et al. Label-efficient time series representation learning: A review. *IEEE Trans. Artif. Intell.* 6027–6042. https://doi.org/10.1109/TAI.2024.3430236 (2024).
- 19. Khan, A. H. et al. Deep learning in the diagnosis and management of arrhythmias. J. Social Res. 4 (1), 50–66. https://doi.org/10.55 324/josr.v4i1.2362 (2024).
- 20. Alsheheri, G. Time series forecasting in healthcare: A comparative study of statistical models and neural networks. *J. Appl. Math. Phys.* **13** (2), 633–663. https://doi.org/10.4236/jamp.2025.132035 (2025).
- Xu, Z. et al. Predicting ICU interventions: A transparent decision support model based on multivariate time series graph convolutional neural network. *IEEE J. Biomedical Health Inf.* 3709–3720. https://doi.org/10.1109/JBHI.2024.3379998 (2024).
- 22. Hochreiter, S. Long Short-term Memory (Neural Computation MIT-, 1997).
- Xia, P., Hu, J. & Peng, Y. EMG-based Estimation of limb movement using deep learning with recurrent convolutional neural networks. Artif. Organs. 42 (5), E67–E77. https://doi.org/10.1111/aor.13004 (2018).
- Schuster, M. & Paliwal, K. K. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* 45 (11), 2673–2681. https://doi. org/10.1109/78.650093 (1997).
- Graves, A. & Schmidhuber, J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Netw. 18 (5–6), 602–610. https://doi.org/10.1016/j.neunet.2005.06.042 (2005).
- 26. Box, G. E. P., Jenkins, G. M., Reinsel, G. C. & Ljung, G. M. Time Series Analysis: Forecasting and Control (Wiley, 2015).
- Benvenuto, D., Giovanetti, M., Vassallo, L., Angeletti, S. & Ciccozzi, M. Application of the ARIMA model on the COVID-2019 epidemic dataset. *Data Brief.* 29, 105340. https://doi.org/10.1016/j.dib.2020.105340 (2020).
- Chakraborty, T. & Ghosh, I. Real-time forecasts and risk assessment of novel coronavirus (COVID-19) cases: A data-driven analysis. *Chaos Solitons Fractals.* 135, 109850. https://doi.org/10.1016/j.chaos.2020.109850 (2020).
- 29. Jakkula, V. Tutorial on support vector machine (svm). School of EECS, Washington State University. 37(2.5), 3 (2006).
- Khalaf, M. et al. An application of using support vector machine based on classification technique for predicting medical data sets. Intelligent Computing Theories and Application: 15th International Conference, ICIC Nanchang, China, August 3–6, 2019, Proceedings, Part II 15. Springer International Publishing. 580–591. (2019). https://doi.org/10.1007/978-3-030-26969-2_55 (2019).
- 31. Breiman, L. Random forests. Mach. Learn. 45, 5-32 (2001).
- 32. Altman, N. S. An introduction to kernel and nearest-neighbor nonparametric regression. Am. Stat. 46 (3), 175–185 (1992).
- 33. Friedman, J. H. Greedy function approximation: a gradient boosting machine. Ann. Stat. 1189–1232. https://doi.org/10.1214/aos/1013203451 (2001).
 24. Sidar Gibbarg, J. A. M. & Sidar Gibbarg, C. L. Making learning in medicine a gravital intra learning. *BMC Med Pape Medical*.
- Sidey-Gibbons, J. A. M. & Sidey-Gibbons, C. J. Machine learning in medicine: a practical introduction. *BMC Med. Res. Methodol.* 19, 1–18. https://doi.org/10.1186/s12874-019-0681-4 (2019).
- Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R. & Schmidhuber, J. LSTM: A search space odyssey. *IEEE Trans. Neural Networks Learn. Syst.* 28 (10), 2222–2232. https://doi.org/10.1109/TNNLS.2016.2582924 (2016).
- Xu, Y. et al. Recurrent attentive and intensive model of multimodal patient monitoring data. Proceedings of the 24th ACM SIGKDD international conference on Knowledge Discovery & Data Mining. 2565–2573. (2018). https://doi.org/10.1145/3219819.3220051
- 37. Hu, F., Warren, J. & Exeter, D. J. Interrupted time series analysis on first cardiovascular disease hospitalization for adherence to lipid-lowering therapy. *Pharmacoepidemiol. Drug Saf.* 29 (2), 150–160. https://doi.org/10.1002/pds.4916 (2020).
- 38. Pietrzykowski, Ł. et al. Medication adherence and its determinants in patients after myocardial infarction. *Sci. Rep.* **10** (1), 12028. https://doi.org/10.1038/s41598-020-68915-1 (2020).
- 39. Li, S. C. X. & Marlin, B. M. A scalable end-to-end Gaussian process adapter for irregularly sampled time series classification. Advances Neural Inform. Process. Systems 29 (2016).
- 40. Melis, G., Kočiský, T. & Blunsom, P. Mogrifier lstm. arXiv preprint arXiv:1909.01792, 2019. (2019). http://arxiv.org/abs/1909.01792
- Zhang, Y., Lan, P., Wang, Y. & Xiang, H. Spatio-temporal Mogrifier LSTM and attention network for next poi recommendation. IEEE International Conference on Web Services (ICWS). IEEE. 2022, 17–26. 2022, 17–26. (2022). https://doi.org/10.1109/ICWS556 10.2022.00019 (2022).
- Che, Z., Purushotham, S., Cho, K., Sontag, D. & Liu, Y. Recurrent neural networks for multivariate time series with missing values. Sci. Rep. 8 (1), 6085. https://doi.org/10.1038/s41598-018-24271-9 (2018).
- Song, H., Rajan, D., Thiagarajan, J. & Spanias, A. Attend and diagnose: Clinical time series analysis using attention models. Proceedings of the AAAI conference on artificial intelligence. 32(1). (2018). https://doi.org/10.1609/aaai.v32i1.11635
- 44. Ma, L. et al. Adacare: explainable clinical health status representation learning via scale-adaptive feature extraction and recalibration. *Proc. AAAI Conf. Artif. Intell.* **34(01)**, 825–832. https://doi.org/10.1609/aaai.v34i01.5427 (2020).
- Li, Z., Li, A., Bai, F., Zuo, H. & Zhang, Y. Remaining useful life prediction of lithium battery based on ACNN-Mogrifier LSTM-MMD. Meas. Sci. Technol. 35 (1), 016101. https://doi.org/10.1088/1361-6501/ad006d (2023).
- Hadsell, R., Chopra, S. & LeCun, Y. Dimensionality reduction by learning an invariant mapping. *IEEE computer society conference* on computer vision and pattern recognition (CVPR'06). *IEEE*. 2, 1735–1742. (2006). https://doi.org/10.1109/CVPR.2006.100
- 47. Schroff, F., Kalenichenko, D., Philbin, J. & Facenet A unified embedding for face recognition and clustering. *Proceedings of the IEEE conference on computer vision and pattern recognition.* 815–823 (2015). (2015).
- Davis, R. A., Lii, K. S. & Politis, D. N. Remarks on some nonparametric estimates of a density function. Selected Works of Murray Rosenblatt. 95–100 (2011). (2011).
- Parzen, E. On Estimation of a probability density function and mode. Ann. Math. Stat. 33 (3), 1065–1076. https://doi.org/10.1214 /aoms/1177704472 (1962).
- Węglarczyk, S. Kernel density estimation and its application. ITM web of conferences. EDP Sciences. 23, 00037. (2018). https://doi. org/10.1051/itmconf/20182300037
- 51. Epanechnikov, V. A. Non-parametric Estimation of a multivariate probability density. *Theory Probab. Its Appl.* 14 (1), 153–158. https://doi.org/10.1137/1114019 (1969).

Author contributions

Experiments design: W.Q. and T.H.; data analysis: F.Z. and M.W.; writing—original draft preparation, W.Q. and T.H.; writing—review and editing, W.Q. and T.H.; visualization, F.Z.; supervision, M.W. and R.H.; project administration, R.H.; funding acquisition, M.W. and R.H. All authors have read and agreed to the published version of the manuscript.

Funding

This work was supported by: the Key-Area Research and Development Program of Guangdong Province (2020B1111100010), the Guangdong Provincial Hospital of Chinese Medicine special projects(YN2023HL03, YN2023ZH06), Science and Technology Planning Project of Guangdong Province (2023B1212060063).

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to T.H. or M.W.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

© The Author(s) 2025