Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 12, No. 1, 2025

Design and Optimization of Multimodal Hybrid Architectures Based on Quantum Computing

Siqi Liu *

College of Basic Medicine, Tianjin Medical University, Tianjin, China

*** Corresponding author Email:** susansusan0709@qq.com

Abstract: In this paper, a hybrid model framework based on quantum computing optimization is proposed for resource demand forecasting and data classification tasks. The study first constructs an AR model for demand forecasting, determines the lag order by PACF and ACF, and transforms the model parameters into QUBO form for optimization and solution after discretization. Secondly, for SVM classification, the penalty function method is adopted to transform the constraints into unconstrained optimization, and the QUBO model is constructed to achieve classification boundary optimization by discretizing the decision variables and introducing penalty function terms. Finally, in the field of deep learning, a parameter optimization method based on Adam's algorithm is proposed to construct a CNN model by combining the ReLU activation function, and the loss function is transformed into a QUBO matrix by parameter quantization technique, and the optimal parameter configurations are solved by using quantum computation. The results show that the framework achieves good results in both prediction accuracy and classification performance, providing a new solution for complex optimization situations.

Keywords: AR Model; CNN Model; SVM Model; QUBO Model.

1. Introduction

Resource demand prediction and data classification, as the core technology to support decision-making, are facing the dual challenges of explosive growth of data scale and enhanced dynamics of complex systems. Traditional prediction models such as AR models[1] suffer from high complexity of parameter optimization in time-series analysis, while classical classification algorithms such as SVM[2] face computational efficiency bottlenecks when dealing with high-dimensional data. Meanwhile, although deep learning models[3] have made significant progress through gradient descent optimization, the exponential growth of the parameter space makes them face the dual challenges of local optimal traps and computational resource limitations. With the breakthrough of quantum computing technology, its unique advantage in combinatorial optimization provides new ideas to solve the above dilemma[4].

In this paper, for the tasks of resource demand prediction and data classification, we propose a hybrid framework integrating classical models and quantum optimization techniques[5], which breaks through the performance boundaries of traditional methods by transforming the traditional model parameter optimization into the form of Quantum Unconstrained Binary Optimization (QUBO) and combining with the parallel searching capability of quantum computing[6]. The research results show that the framework significantly improves the prediction accuracy and classification efficiency while maintaining the model interpretability, providing an interdisciplinary solution for complex system optimization.

2. Projected Resource Requirements

(1) Establish the AR model:

$$\hat{y}_t = c + \sum_{i=1}^{p} \phi_i y_{t-i} + \delta_t \quad (1)$$

(2) Minimize the sum of squares of error (least squares):

$$\min_{\phi_i, c} \sum_{t=p+1}^{T} \left(y_t - c - \sum_{i=1}^{p} \phi_i y_{t-i} \right)^2 \quad (2)$$

(3) Parameter discretization: express the parameter c as:

$$c = c_{\min} + \delta_c \sum_{k=1}^{n_c} x_{c,k} 2^{k-1}, X_{c,k} \in \{0,1\} \quad (3)$$

And express the parameter ϕ_i as:

$$\phi_i = \phi_{i,\min} + \delta_\phi \sum_{k=1}^{n_\phi} x_{\phi_i,k} 2^{k-1} \quad (4)$$

(4) Set the parameter value range and accuracy: set c [8000,11000].

$$\delta_c = \frac{c_{\max} - c_{\min}}{2^{n_c} - 1} = \frac{11000 - 8000}{15} = 200 \quad (5)$$

$$\delta_\phi = \frac{\phi_{\max} - \phi_{\min}}{2^{n_\phi} - 1} = \frac{1 - (-1)}{15} \approx 0.1333 \quad (6)$$

(5) Turn the AR model into the optimization model QUBO, set the objective function as, and bring the two types in the third step, to build the objective function, and the final optimization goal is as follows:

$$f(x) = \sum_{t=p+1}^{T} \left(y_t - c - \sum_{i=1}^{p} \phi_i y_{t-i} \right)^2 f(x) = x^T Q x + q^T x + const \quad (7)$$

$$\min_{x \in \{0,1\}} f(x) = \min_{x \in \{0,1\}^n} (x^T Q x + q^T x) \quad (8)$$

Where x represents the vector containing all binary variables, the length is n, Q represents the symmetrical quadratic term coefficient matrix, size nn,q represents the vector of primary term coefficient, length n, const represents the constant term, independent of the optimization objective, can be ignored.

(6) Make $t = 10$, bring the parameter c and the parameter in the first step sub, make $t = 10$, predict the demand in October.

3. Classification of Data Sets

(1) To express the original optimization as:

$$\min_{w,b} \frac{1}{2}\|w\|^2 \tag{9}$$

Subject to:

$$g_i(w,b) = y_i(w^T x_i + b) - 1 \geq 0, i = 1,2,\dots,N \tag{10}$$

(2) By using the penalty function method, the constraint condition is merged into the objective function through the way of the penalty term, and thus becomes the unconstrained function.

(3) Discrete the decision variables: let (w, b) be the decision variable, set:

$$w_j \in [w_{\min, w_{\max}}] \tag{11}$$

Then:

$$\Delta w = \frac{w_{\max} - w_{\min}}{2^n - 1} \tag{12}$$

And then:

$$b_j \in [b_{\min}, b_{\max}] \tag{13}$$

$$\Delta b = \frac{b_{\max} - b_{\min}}{2^n - 1} \tag{14}$$

(4) For each w_j, its discretization is expressed as:

$$w_j = w_{\min} + \sum_{k=1}^{n} b_{jk} \times 2^{k-1} \times \Delta w \tag{15}$$

For each b_j, its discretization is expressed as:

$$b_j = b_{\min} + \sum_{k=1}^{n} b_{jk} \times 2^{k-1} \times \Delta b \tag{16}$$

$$b_{jk} = 0 \; or \; 1 \tag{17}$$

(5) The discretized w and b are inserted into the objective function and the constraints to construct the objective function.

$$\min_{w} \frac{1}{2}\|w\|^2 = \frac{1}{2}\sum_{j=1}^{d} w_j^2 \tag{18}$$

Substitute discrete expression of: w_j:

$$w_j = w_{\min} + \sum_{k=1}^{n} b_{jk} \times 2^{k-1} \times \Delta w \tag{19}$$

(6) Convert the constraints into the penalty functions.

$$\text{Penalty}_i = \left[\max\left(0, 1 - y_i(w^T x_i + b)\right)\right]^2 \tag{20}$$

(7) Build the overall objective function:

$$\min \frac{1}{2}\sum_{j=1}^{d} w_j^2 + C\sum_{i=1}^{N} \left[\max\left(0, 1 - y_i(w^T x_i + b)\right)\right]^2 \tag{21}$$

(8) Convert to QUBO form:

$$w_j \in \{-1,1\}, b \in \{-1,1\} \tag{22}$$

Convert a continuous variable to a binary variable:

$$\begin{gathered} \frac{1}{2}\sum_{j=1}^{d} w_j^2 \\ \because w_j \in \{-1,1\} \\ \therefore w_j^2 = 1 \\ \frac{1}{2}\sum_{j=1}^{d} 1 = \frac{1}{2} \end{gathered} \tag{23}$$

Processing penalty function:

$$q_i = C\sum_{i=1}^{N} \left[\max\left(0, 1 - y_i(w^T x_i + b)\right)\right] \tag{24}$$

Then the penalty function is: $C\sum_{i=1}^{N} q_i^2$.
make $q_i = q_{i0} - q_{i1}$.

$$q_{i0}, q_{i1} \in \{0,1\} \tag{25}$$

$$q_{i0} + q_{i1} - 1 \leq 0 \tag{26}$$

Then,

$$q_i^2 = q_{i0}^2 + q_{i1}^2 - 2q_{i0}q_{i1} \tag{27}$$

$$q_i^2 = q_{i0} + q_{i1} - 2q_{i0}q_{i1} \tag{28}$$

$$\min \frac{d}{2} + C\sum_{i=1}^{N} q_{i0} + q_{i1} - 2q_{i0}q_{i1} \tag{29}$$

4. Deep Learning Model Building

(1) Let the loss function be $f(\theta)$, and n be the number of samples, then:

$$f(\theta) = \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(\hat{y}_i, y_i) \tag{30}$$

Where $\hat{y}_i$ is the predicted value and y_i is the true value.

(2) minimize $f(\theta)$ through the original Adam algorithm

Initialization parameters: in f(θ),θ is all the parameters and is random values, the first moment vector $m = 0$, the second moment vector $v = 0$, the hyperparameter a $0.001, \beta_1 = 0.9, \beta_2 = 0.999, \varepsilon = 10^{-8}$.

Iterative update parameters:

$$g_t = \nabla f(\theta) \tag{31}$$

$$\begin{cases} m_t = \beta_1 m_{t-1} + (1-\beta_1)g_t \\ v_t = \beta_2 v_{t-1} + (1-\beta_2)g^2 t \end{cases} \tag{32}$$

$$\begin{cases} \hat{m}_t = \frac{m_t}{1-\beta_1^t} \\ \hat{v}_t = \frac{v_t}{1-\beta_2^t} \end{cases} \tag{33}$$

$$\theta_{t+1} = \theta_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \tag{34}$$

Continuous iteration until convergence.

(3) Parameter quantization: Let each parameter θ_j as m-bit binary number, defined $x_{j,k}$ as the k -bit binary variable of the j -th parameter:

$$\theta_j = \sum_{k=1}^{m} x_{j,k} 2^k \tag{35}$$

Where $x_{j,k} \in \{0,1\}$, representing the state of the Merer binary, m is the quantization accuracy.

(4) In the quantum optimization, the optimization problem is converted into the QUBO form:

$$\min_x x^T Q x \tag{36}$$

Where x is the vector of all the binary decision variables, and Q is the QUBO matrix, representing the quadratic term function of the loss function.

(5) Convert the loss function $L(\theta)$ into the QUBO form Q_{jk}. The element of the QUBO matrix corresponds to the quadratic partial derivative of each parameter in the loss function:

$$Q_{jk} = \frac{\partial^2 L(\theta)}{\partial x_j \partial x_k} \tag{37}$$

(6) ReLU activation function: For the input x,

$$\text{ReLU}(x) = \max(0, x) \tag{38}$$

(7) Suppose there is an L layer network, for each layer 1, the goal of minimizing is to use the overloss function to find the best parameter θ_l. The QUBO of this layer is expressed as:

$$\min_{x_l} x_l^T Q_l x_l \tag{39}$$

Where, x_l is the binary decision variable of the 1 layer, Q_l is the corresponding QUBO matrix. The objective function of the whole network is the sum of the objective functions of all the layers:

$$\min_{x} \sum_{l=1}^{L} x_l^T Q_l x_l \tag{40}$$

(8) QUBO solution:

Data coding: encode the input data and labels in a binary format suitable for quantum computing.

Build the QUBO matrix: calculate the QUBO matrix of each CNN, which contains the interaction information between various parameters Q_l.

Quantum computing solution: solve the QUBO problem by simulated annealing and obtain the optimal parameter configuration θ.

Decoding and model training: the resulting binary system is decoded as the network parameters to update the weight and configuration of the network.

5. Conclusion

In this paper, we propose a quantum-inspired hybrid optimization framework for key scientific situations in resource management and pattern recognition, which enhances the global search capability by transforming the traditional model parameter learning process into quantum unconstrained binary optimization (QUBO) and exploiting the parallel state superposition property of quantum computing. The research focuses on revealing the transformation mechanism of different models (AR, SVM, CNN) to the QUBO form, constructing a multimodal optimization model, and verifying its improvement in prediction accuracy and classification efficiency through numerical experiments. This research not only expands the application boundary of quantum computing in classical optimization, but also provides an interdisciplinary solution for intelligent decision-making of complex systems.

References

[1] QIAN Yu, WANG Lixin, ZHANG Heng, et al. Research on cruise flight parameter prediction based on adaptive AR model[J]. Computer Application and Software,2024, 41(04): 73-79.

[2] LI Zhong, JIANG Yibo, CAO Kexin, et al. Research on SVM-based quantum line output calibration method[J]. Journal of Quantum Electronics,2024,41(02):357-366.

[3] LI Jiansong, LI Hai, YU Wenli, et al. Optimization of quantum Otto loop performance based on deep reinforcement learning[J]. Journal of Quantum Electronics,2025,42(01):70-79.

[4] Guo Guoping. Industrial change and ecological construction of quantum computing technology[J]. People's Forum, 2023, (16): 13-17.

[5] HE Jianhao, LI Luzhou. A review of quantum optimization algorithms[J]. Computer Research and Development, 2021,58 (09): 1823-1834.

[6] WU Yongfei, WANG Yanbo, XU Qi, et al. Research on the application of quantum QUBO-SVM technology in finance[J]. Banker, 2025, (01):125-128.