



A nuclide identification method of γ spectrum and model building based on the transformer

Fei Li^{1,2,3} · Chu-Yang Luo¹ · Ying-Zi Wen¹ · Sheng Lv¹ · Feng Cheng^{1,2} · Guo-Qiang Zeng^{1,2} · Jian-Feng Jiang¹ · Bing-Hai Li^{4,5}

Received: 30 October 2023 / Revised: 8 February 2024 / Accepted: 20 February 2024 / Published online: 18 December 2024

© The Author(s), under exclusive licence to China Science Publishing & Media Ltd. (Science Press), Shanghai Institute of Applied Physics, the Chinese Academy of Sciences, Chinese Nuclear Society 2024

Abstract

In current neural network algorithms for nuclide identification in high-background, poor-resolution detectors, traditional network paradigms including back-propagation networks, convolutional neural networks, recurrent neural networks, etc., have been limited in research on γ spectrum analysis because of their inherent mathematical mechanisms. It is difficult to make progress in terms of training data requirements and prediction accuracy. In contrast to traditional network paradigms, network models based on the transformer structure have the characteristics of parallel computing, position encoding, and deep stacking, which have enabled good performance in natural language processing tasks in recent years. Therefore, in this paper, a transformer-based neural network (TBNN) model is proposed to achieve nuclide identification for the first time. First, the Geant4 program was used to generate the basic single-nuclide energy spectrum through Monte Carlo simulations. A multi-nuclide energy spectrum database was established for neural network training using random matrices of γ -ray energy, activity, and noise. Based on the encoder–decoder structure, a network topology based on the transformer was built, transforming the 1024-channel energy spectrum data into a 32×32 energy spectrum sequence as the model input. Through experiments and adjustments of model parameters, including the learning rate of the TBNN model, number of attention heads, and number of network stacking layers, the overall recognition rate reached 98.7%. Additionally, this database was used for training AI models such as back-propagation networks, convolutional neural networks, residual networks, and long short-term memory neural networks, with overall recognition rates of 92.8%, 95.3%, 96.3%, and 96.6%, respectively. This indicates that the TBNN model exhibited better nuclide identification among these AI models, providing an important reference and theoretical basis for the practical application of transformers in the qualitative and quantitative analysis of the γ spectrum.

Keywords Nuclide identification · Neural network · Transformer

1 Introduction

Multi-nuclide identification is a radioactive material detection technology that is vitally important in medicine, national militaries, and social stability [1, 2]. For the γ

This work was supported by the National Natural Science Foundation of China (No. 42127807), Natural Science Foundation of Sichuan Province (Nos. 2024NSFSC0422, 23NSFSC0116), and Nuclear Energy Development Project (No. [2021]-88).

✉ Feng Cheng
chengfeng@cdut.cn

✉ Guo-Qiang Zeng
zgq@cdut.edu.cn

¹ College of Nuclear Technology and Automation Engineering, Chengdu University of Technology, Chengdu 610000, China

² Applied Nuclear Technology in Geosciences Key Laboratory of Sichuan Province, Chengdu 610000, China

³ Department of Physics, Universitat Politècnica de Catalunya, Barcelona 08028, Spain

⁴ Airborne Survey and Remote Sensing Center of Nuclear Industry, Shijiazhuang 050000, China

⁵ Hebei Key Laboratory of Airborne Detection and Remote Sensing Technology, Shijiazhuang 050000, China

spectrum under a complex background with low count and signal-to-noise ratio, multi-nuclide identification is challenging [3]. Traditional machine learning-based methods are suitable for situations with few types of nuclides (≤ 4) and small amounts of data [4]. However, they have problems such as low recognition rates or an inability to quickly identify nuclides. Conventional shallow neural network structures tend to overfit and exhibit poor generalization during training, whereas deep network structures can alleviate these problems [5]; however, they have numerous hyperparameters and are difficult to train.

In the early days, owing to the insufficient calculation capacity of computers, nuclide identification classification mainly used traditional peak searching methods based on the full-energy peak, such as the maximum value, symmetric zero-area conversion, and derivative methods [6, 7]. However, traditional methods are not applicable for situations with low count rates or low peak-to-background ratios [8].

Classification algorithms based on machine learning, including Bayesian, decision tree, and support vector machine algorithms, as well as their variants, have been applied for nuclide identification. They have the advantages of fast recognition speed and high accuracy under low-count-rate conditions compared to traditional methods [9–13]. Additionally, machine learning algorithms require researchers to manually extract data features [14, 15], such as the peak position and peak boundaries of characteristic peaks, background of the spectrum, and noise of the spectrum, then make decision inferences based on the extracted feature data, and finally perform classification statistics.

The accuracy of nuclide identification depends on the continuous improvement of feature extraction algorithms; however, increasingly complex feature extraction steps increase the difficulty of nuclide identification [16]. The multilayer perceptron (MLP) in machine learning has been applied in fields such as nuclide identification and signal processing [17, 18]. The MLP has good approximation and generalization abilities. For this approach, the overall prediction accuracy in nuclide identification tasks can exceed 90%; however, it can easily fall into locally optimal values. Combining feature extraction algorithms with the MLP is can effectively improve the system and increase the accuracy of nuclide identification by 5–9% [19–23]. For example, Yicong et al. used an improved particle swarm algorithm to optimize the threshold and weight values of a back-propagation network (BP) [24]. However, these machine learning-based nuclide identification methods exhibit a sharp drop in the recognition rate for cases involving more than five radioactive nuclides [4], and the mode dependence on the manual extraction of data features increases the difficulty of nuclide identification.

In recent years, with improvements in the performance of hardware such as graphics processing units (GPUs) and

tensor processing units (TPUs), convolutional neural networks (CNNs), recurrent neural networks (RNNs), and extended deep learning models based on CNNs or RNNs have begun to show advantages. Deep learning, as a branch of machine learning, has been widely applied in fields such as image recognition, target detection, and semantic segmentation owing to its simplification of tensor operations, good scalability, and universality [25, 26]. Deep learning models usually consist of multiple layers, each fully connected to those in the layer below (from which they receive input) and those above (which they, in turn, influence). The entire model and its constituent layers share this structure. The entire model uses raw inputs (features), generates outputs (predictions), and possesses parameters (combined parameters from all constituent layers). Similarly, each individual layer receives inputs (supplied by the previous layer), generates outputs (the inputs to the subsequent layer), and possesses a set of tunable parameters that are updated according to the signal that flows backwards from the subsequent layer.

To date, researchers have explored the potential applications of deep learning models in the field of nuclide identification. The research focus can be summarized into three aspects: the construction of datasets, preprocessing of input data, and improvement of network models [27–29]. Increasing the depth of the network models improves nuclide identification performance to some extent, such as using ResNet-50 for full-spectrum all-nuclide identification [30]. However, the large number of hyperparameters ($\geq 10^7$) makes the model training time-consuming and difficult to adjust. Additionally, this approach has high hardware requirements. The performance of network models currently used in the field of nuclide identification, including BP, CNNs, residual networks (ResNet), long short-term memory neural networks (LSTMs), etc., is limited by their own mathematical mechanisms. It is difficult to make breakthroughs in terms of training data requirements and prediction accuracy. Therefore, the application of new network model paradigms to nuclide identification tasks represents a potential improvement.

The transformer is a neural network architecture that is different from traditional models such as RNN or CNN. It uses techniques such as residual connection and layer normalization to accelerate parallel computing [31]. It is usually used for natural language processing (NLP) tasks such as language translation, language modeling, and sentiment analysis. For example, the ChatGPT model, which received significant attention at the end of 2022, was improved based on the transformer structure. The key innovation of the transformer is its self-attention mechanism, which allows networks to focus on different parts of the input sequence at different computational stages without relying on a fixed-length context window to process the input sequence. This allows networks to dynamically focus on them based on the

relevance between different parts of the input sequence and the task at hand [32].

Although the transformer architecture was originally introduced for natural language processing tasks, researchers have also explored its potential in the field of image recognition. One of the main advantages of using a transformer for image recognition is that it can capture long-range correlations in input images. By using the self-attention mechanism, the transformer can learn to extract features that are useful for image recognition tasks [33, 34]. Dosovitskiy et al. proposed a visual transformer (ViT) method that combines a CNN and the transformer to perform image recognition tasks, whereas Si et al. established an inception transformer (iFormers) architecture to improve the deficiency of the transformer in capturing high-frequency local information [35, 36]. The ViT and iFormers methods have shown good results in various image recognition tasks, including classification, segmentation, and object detection. Nuclide identification methods based on deep learning use full-spectrum data as input, which can be used as one-dimensional sequence data and can also be transformed into a two-dimensional image form input. Therefore, the transformer architecture is theoretically applicable to nuclide identification tasks.

This paper proposes, for the first time, the use of the transformer model to replace the traditional network model paradigms used for nuclide identification and explores the potential application of the transformer in nuclide identification. This study verifies the scalability of this method, as well as its stable gradient propagation ability, high accuracy of spectrum nuclide identification, and good robustness through training and testing with simulated spectrum data. Moreover, we provide a comparison with the traditional BP, CNN, and RNN in terms of the full-spectrum recognition rate and convergence speed.

2 Algorithm and model

2.1 Construction of training set through MC simulation method

According to the IAEA-2006 standard industrial nuclide library [37], the radioactive nuclides used for nuclide identification in this study were ^{241}Am , ^{192}Ir , ^{226}Ra , ^{133}Ba , ^{60}Co , ^{57}Co , ^{137}Cs and ^{152}Eu . Currently, Monte Carlo (MC)-based methods are commonly used in nuclear science and nuclear detection to simulate radioactive nuclide spectra. Geant4 is a software package developed by the European Organization for Nuclear Research (CERN) using the C++ language platform, which is widely used in nuclear physics, radiation protection, and detection [38]. This study used Geant4 to simulate the NaI detector response in

nuclide spectrum simulation experiments. The simulated NaI detector uses a standard cylinder of size $\phi 5\text{ cm} \times 5\text{ cm}$.

The radiation source was set as a point source located $5 \sim 15\text{ cm}$ (randomly selected) directly in front of the detector, and the number of emitted particles was 1.0×10^8 . Considering the different energy resolutions of the detector in real scenarios, this study adopted the Gaussian broadening formula as follows:

$$E_{\text{new}} = \text{Gauss}(x_{\text{Random}}; E_i, \delta_{\text{FWHM}}), \quad (1)$$

$$\delta_{\text{FWHM}} = \frac{a + b\sqrt{E_i + cE_i}}{2.3548}, \quad (2)$$

where x_{Random} is a random number; E_i is the initial energy of the i -th channel; and a , b , c are the broadening coefficients of the detector [39]. After obtaining the simulated radioactive nuclide spectrum data through an MC simulation, we obtained the initial 80 spectrum data (ten spectra for each nuclide), each with 1024 channels (0–3 MeV). Considering the activity size of radioactive nuclides and the impact of environmental noise in actual measurements, this study expanded the nuclide spectrum dataset by adding random noise and amplitude transformation and used random combinations to obtain mixed spectra of multiple source nuclides. The calculation formula is as follows:

$$\begin{aligned} \text{Spec}_{\text{new}} = & \text{Add}_{\text{noise}} \\ & \{ \text{Spec}_{^{241}\text{Am}} \times \text{random}(0.1 \sim 10) \text{random}[0, 1] \\ & + \text{Spec}_{^{192}\text{Ir}} \times \text{random}(0.1 \sim 10) \text{random}[0, 1] \\ & + \cdots + \text{Spec}_{^{152}\text{Eu}} \times \text{random}(0.1 \sim 10) \text{random}[0, 1] \} \end{aligned} \quad (3)$$

where $\text{Add}_{\text{noise}}$ refers to adding Gaussian white noise with a signal-to-noise ratio (SNR) between 25 and 35, $\text{random}(0.1 \sim 10)$ is a random number from 0.1~10, $\text{random}[0, 1]$ refers to a random value of 0 or 1, and Spec_n is the corresponding nuclide spectrum. The spectrum dataset for this study was constructed in this way with a database volume of 1.0×10^4 .

2.2 Construction of neural network models

2.2.1 Transformer-based neural network (TBNN) nuclide identification model

The attention mechanism can be considered as a simulation of the human visual mechanism. The processing of visual information consumes a large amount of brain resources. To make full use of brain resources, it will not process all information at the highest granularity, but will focus on the parts of interest and apply brain resources to

them [40]. The multi-head attention mechanism is derived from the self-attention mechanism, which is also the core part of the transformer.

As shown in Fig. 1, the self-attention mechanism is a top-down mechanism used to calculate the similarity and feature associations between different data [41]. The self-attention mechanism can be represented by the following formula:

$$Q, K, V = sW^Q, sW^K, sW^V \quad (4)$$

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (5)$$

where $s \in R^{l \times h}$ represents an input sequence with a sequence length of l and a feature dimension of h , and W^Q, W^K, W^V are fully connected layers of size $h \times k$, which map each piece of input s data to q, k, v vectors of size $1 \times k$, respectively, and combine them into Q, K, V matrices of size $l \times k$. By calculating the corresponding attention scores α using the dot product of q and each k vector, then summing the corresponding v vectors (weighted by each normalized attention score), we obtain a new output corresponding to this q . In the matrix representation, the attention score matrix of size $l \times l$ is $softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)$.

The multi-head attention mechanism calculates $nhead$ attention vectors by dimensionally increasing through a fully connected layer to $nhead \times$ dimension and then uses a fully connected layer to fuse the features calculated through multi-head attention. The multi-head attention mechanism can be represented by the following formula:

$$\begin{aligned} Q_{nhead}, K_{nhead}, V_{nhead} \\ = QW_{nhead}^Q, KW_{nhead}^K, VW_{nhead}^V \end{aligned} \quad (6)$$

$$Multihead = Attention(Q_{nhead}, K_{nhead}, V_{nhead}), \quad (7)$$

$$MultiHead(Q, K, V) = MultiHead \cdot W^0, \quad (8)$$

$W_{nhead}^Q, W_{nhead}^K, W_{nhead}^V$ correspond to the multi-head dimensional matrices of Q, K, V respectively, and W^0 is a dimensional reduction matrix used for fusing multi-head attention features.

The transformer is a model based on the multi-head attention mechanism, where each sub-layer uses residual connections and then performs layer normalization. The transformer-based neural network (TBNN) model designed in this study based on the transformer structure is shown in Fig. 2. In the training stage, the input “Targets” of the decoder part denotes the predicted targets corresponding to the dataset and is used to enforce teaching on the network. In the prediction stage, the decoder part performs cyclic prediction until 32 sequence predictions are completed, and finally outputs the final result through the “FN+Flatten” layer.

The spectrum output simulated by the NaI detector consists of 1024 channels. The model first transforms the 1×1024 spectrum into a 32×32 sequence, and no longer uses the embedding layer of the original transformer model to encode the 1024 data. Positional encoding adds absolute or relative positional information to the input sequence. Fixed-position encoding based on sine and cosine functions was used, and the calculation formula is as follows:

$$PE_{m,2n} = \sin\left(\frac{m}{10000^{2n/32}}\right), \quad (9)$$

$$PE_{m,2n+1} = \cos\left(\frac{m}{10000^{2n/32}}\right), \quad (10)$$

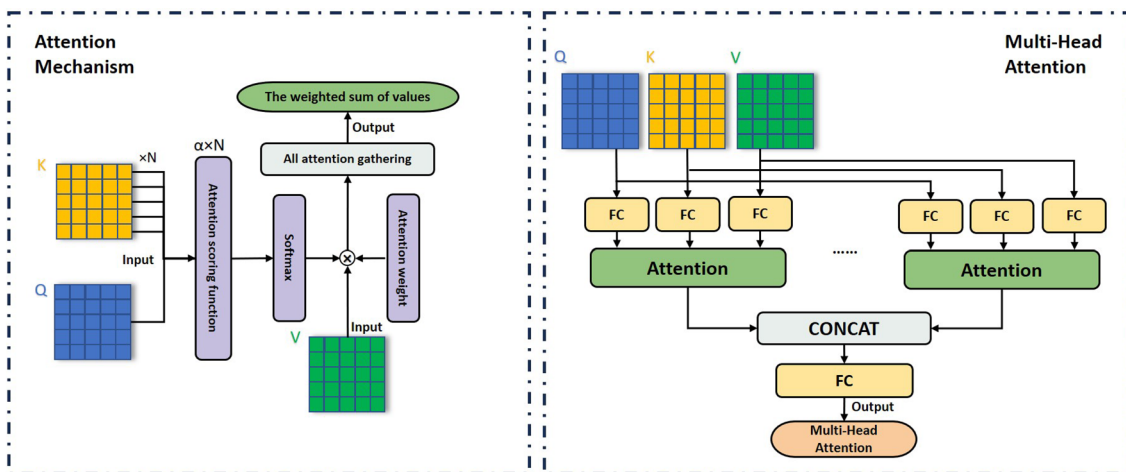


Fig. 1 (Color online) Schematic diagram of attention mechanism and multi-head attention

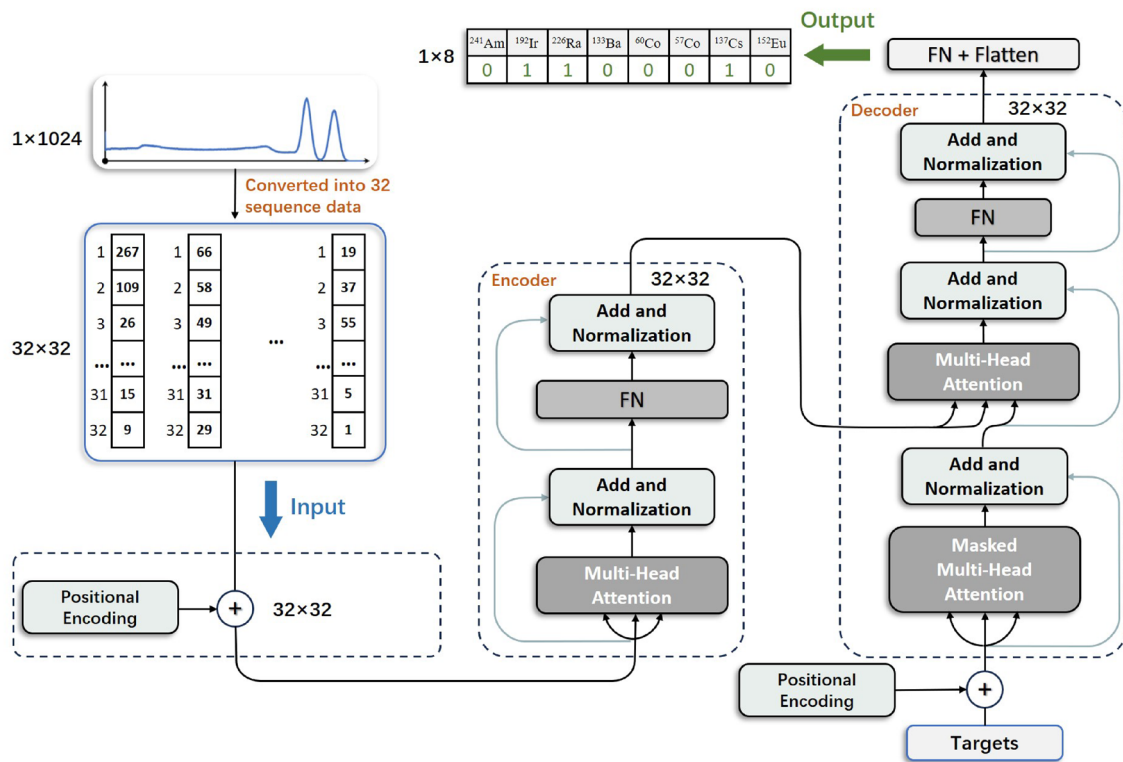


Fig. 2 The structure of the transformer-based nuclide identification model in the paper

where $PE_{m,2n}$ and $PE_{m,2n+1}$ represent the position encoding calculation results of indices $(m, 2n)$ and $(m, 2n + 1)$ in the sequence, respectively. Adding them to the sequence before encoding yields the output of the position encoding layer.

The output of the position encoding enters the encoder and decoder structure and then flattens the sequence and connects it to a fully connected layer, finally outputting a nuclide identification result with a dimension of 1×8 .

2.2.2 Other neural network models for comparison

To evaluate the performance of the newly constructed model, this study selects four neural network models that have been widely used in the field of nuclide identification in recent years for comparison. These networks are BP, CNN, ResNet, and LSTM [42, 43].

The CNN is essentially a multilayer perceptron composed of a convolutional layer, a pooling layer, an input layer, a fully connected layer, and an output layer [44]. The convolutional and pooling layers in the hidden layer form the core of the CNN for achieving feature extraction. The convolution operation of the convolutional layer achieves local connections; its output data are calculated for each neuron using the same convolution kernel (shared weight) and then added with the same bias (shared bias). The pooling layer also contains something similar to a

convolutional kernel for performing pooling operations on data, but it does not have any parameters to learn; it only takes the maximum or average value from the target region. The pooling operation utilizes the principle of local image correlation to sample the data, which increases the robustness of the CNN to small positional changes while retaining useful information. These two main layers effectively reduce the number of network parameters and alleviate the problem of model overfitting. Assuming that the input is N , the convolution kernel is L , and the feature mapping is M , the corresponding two-dimensional convolution operation expression is as follows:

$$S(n, m) = (1 \cdot L)(n, m) = \sum_{i=0}^I \sum_{j=0}^J N(i, j) L(n - i, m - j) \quad (11)$$

where I is the width of the convolution kernel and J is the height of the convolution kernel.

ResNet is an improved CNN model. It is built from residual blocks, which are connected to form a residual network [45]. Among them, residual connections allow the network to reach more ancestors during back-propagation, thereby alleviating network degradation. ResNet performs well when training deep networks [30]. Effective deep neural networks can be trained using ResNet. The main formula for the residual network is:

$$F(x) = g(x) + x, \quad (12)$$

where $F(x)$ is the final output of ResNet, $g(x)$ is the output of the two convolutions, and x is the sample data.

LSTM was proposed by Hochreiter and Schmidhuber in 1997 to solve problems such as those faced by traditional RNN models, which have difficulty learning dependencies between long-term information and can easily cause gradient vanishing and gradient explosion [46]. The LSTM model selects additions and deletions during data input. Its key part mainly relies on three gate structures: forget, update, and output gates [47]. The forget gate determines the data to be discarded or retained through the sigmoid layer. Then, the update gate mainly screens the data content and selects data updates to the status, mainly determined through the tanh and sigmoid layers. Finally, the output gate combines the current memory with the long-term memory and then judges whether this result should be output through the sigmoid activation function layer, thus passing it to the next neuron cell. The sigmoid and tanh layers use the sigmoid and tanh functions as activation functions, respectively, to enhance the nonlinear relationships between neurons.

Theoretically, for BP, a three-layer neural network (with a concise structure and relatively few parameters) can approximate a given function with arbitrary accuracy, which is a tempting prospect [19]. Simultaneously, BP has a certain ability to promote and summarize. However, from a mathematical perspective, the BP is a local search method. When solving for the global extremum of a complex nonlinear function, it is highly likely to fall into local extrema, causing training failures [20, 21]. CNNs perform excellently in feature extraction for two-dimensional inputs, and their structural features are local connections, weight sharing, and pooling operations [18, 28]. Owing to the balance between local feature extraction and global feature interaction, CNNs usually outperform BP [27]. The main reason for using ResNet in this study was to test whether it could achieve better performance than a CNN when increasing the depth of the convolutional networks. In theory, owing to the use of residual blocks, ResNet can train deeper neural networks, such as ResNet-50, which performs well on the nuclide recognition task [30]. The LSTM model exhibits outstanding performance in processing sequence inputs, which mitigates the long-term dependency problem in RNNs [29]. Each LSTM cell has several MLPs, and if the time span of the LSTM is large and the network is deep, the training phase is computationally intensive and time-consuming (because parallel computing is not used). The parallel computation of the multi-head attention mechanism in the transformer model significantly improves the efficiency of training and inference, allowing larger models and processing of longer sequences. However, its high computational cost, structural complexity, and number of hyperparameters increase the

difficulty of optimization, requiring careful adjustment of hyperparameters such as the learning rate and batch size to achieve better performance [33, 34]. The energy spectrum data can be regarded as both a two-dimensional image and a sequence; therefore, the above models were selected to test their nuclide recognition performance.

3 Results and Discussion

Eight nuclides from the industrial nuclide library were selected as the nuclides to be analyzed, and the Geant4 simulation NaI detector was used to establish a spectrum database corresponding to the eight nuclides. To bring the simulated data closer to the actual spectrum, the simulated database was preprocessed using Gaussian broadening and random noise superposition. Based on the principle of data augmentation, the database size was expanded to 1.0×10^4 . TBNN was established for the characteristics of input spectrum data, and traditional network paradigms including BP, CNN, and RNN were also constructed for comparative experiments to analyze the recognition rate, convergence speed, and other aspects of the performance of the TBNN model. The overall process is illustrated in Fig. 3.

3.1 Data preparation

Through the method in Sect. 2.1, a nuclide library was generated, containing eight industrial radioactive nuclides: ^{241}Am , ^{192}Ir , ^{226}Ra , ^{133}Ba , ^{60}Co , ^{57}Co , ^{137}Cs , and ^{152}Eu . The volume of the database was 1.0×10^4 . Each γ spectrum in the database was formed using a random combination of eight nuclides with random activity. Some γ spectra are shown in Fig. 4. The dataset annotation in this paper adopts the $[1 \times 8]$ matrix label to be annotated, each column in the matrix corresponds to ^{241}Am , ^{192}Ir , ^{226}Ra , ^{133}Ba , ^{60}Co , ^{57}Co , ^{137}Cs , and ^{152}Eu , respectively, “0” indicates the absence of the radionuclide, and “1” indicates the presence of the radionuclide. For example, if a spectrum contains ^{192}Ir , ^{152}Eu , ^{57}Co three radionuclides, then the label is “[0, 0, 1, 0, 0, 1, 0, 1].” Therefore, the spectrum recognition task in this study corresponds to a multi-label classification task. Each label was annotated after the MC simulation.

Traditional nuclide library comparison methods are highly complex and inaccurate in identifying nuclides during the process [4, 22]. In the nuclide library selected in this study, there were some nuclides with similar characteristic γ energies, leading to overlapping peaks. For example, the difference between the characteristic γ ray energy of ^{57}Co ’s 122.0614 keV (85.60%) and ^{152}Eu ’s 121.782 keV (39.76%) is far less than the energy resolution of the NaI scintillator detector (7% ~ 9%, ^{137}Cs @ 661 keV). As shown in Fig. 5, although they have other smaller branch ratio characteristic

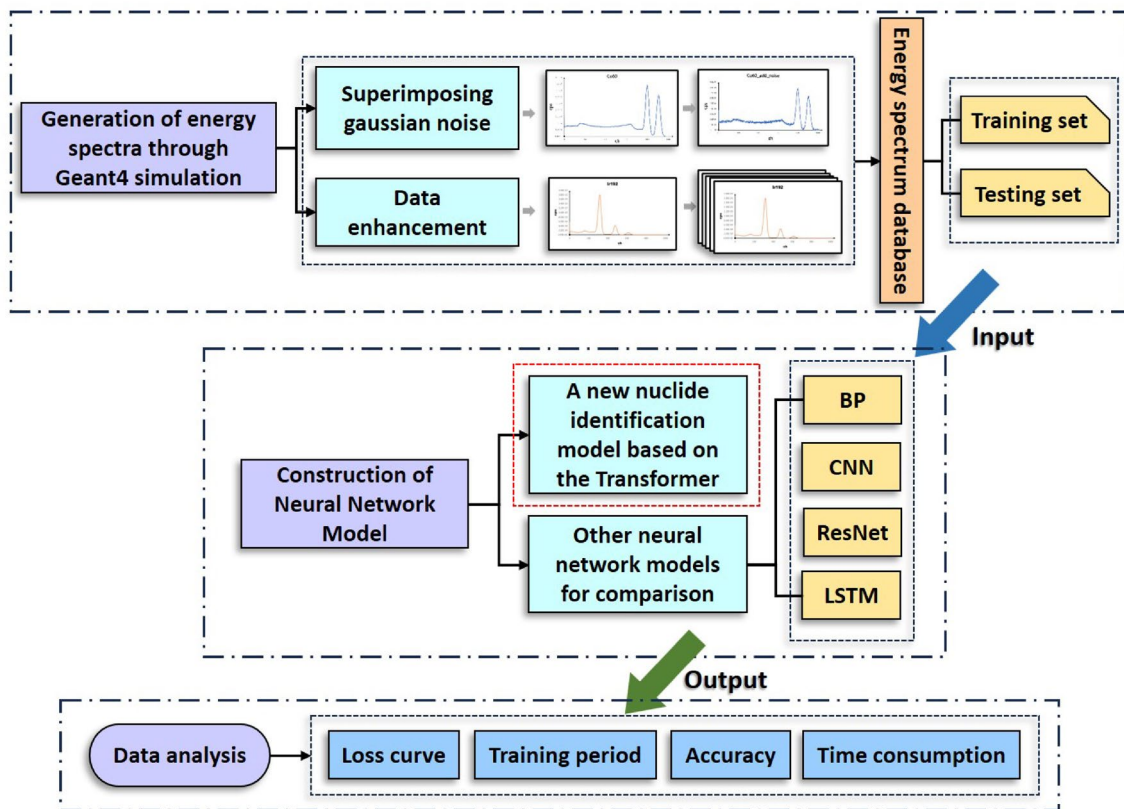
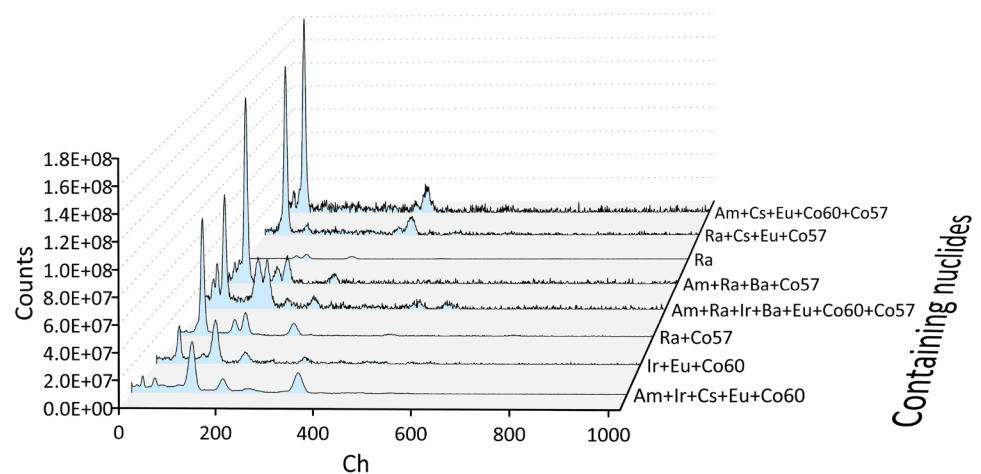


Fig. 3 (Color online) The workflow diagram of the research process in this study. It mainly includes three parts: dataset preparation, model construction, and data analysis

Fig. 4 Some γ spectrum samples generated by the MC simulation. The horizontal axis represents the number of energy channels, and the vertical axis represents the count of each energy channel. The range of energy is 0 ~ 3 MeV



peaks, it is difficult to identify the two types of nuclides from the spectrum under high-background and low-activity conditions. The database generated by Geant4 in this study used a combination spectrum method of random activity, random noise, and random nuclide types to objectively verify the ability of the TBNN model to identify overlapping peaks.

Neural network models usually require a database independent of the training set to evaluate the inference ability

of the model; therefore, the nuclide library obtained by simulation was divided into a training set and a validation set at a ratio of 9:1. To make the model easier to train, before inputting data into the model for training, it was necessary to normalize the spectrum data of 1×1024 . Z-score normalization was used to transform the data to follow a normal distribution. The formula for this is as follows:

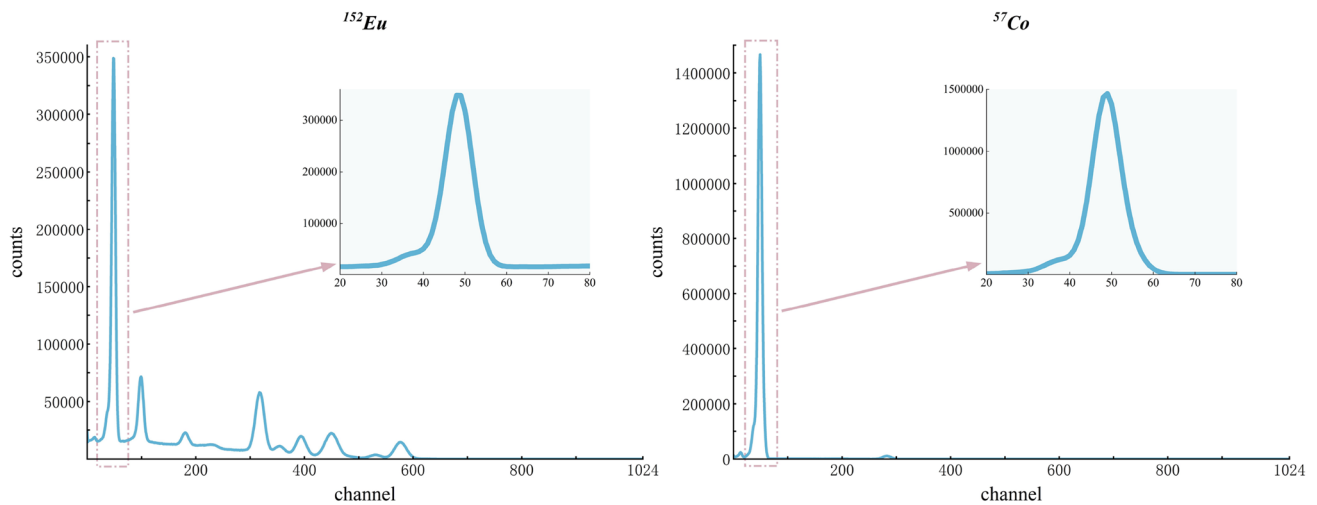


Fig. 5 The original γ spectrum of ^{57}Co (122.0614 keV, 85.60%) and ^{152}Eu (121.782 keV, 39.76%). The peak positions of ^{57}Co and ^{152}Eu almost overlap

$$X_{i\text{Norm}} = \frac{X_i - \mu}{\sigma} \quad (13)$$

where X_i denotes the i -th channel spectrum data, μ represents the mean value of this spectrum, σ represents the standard deviation of this spectrum, and $X_{i\text{Norm}}$ is the normalization result of the i -th channel spectrum data.

3.2 Training of models

3.2.1 Settings of models for comparison

Referring to the principles of model design in Sect. 2, the model designed in this study was built based on Python and used Keras and TensorFlow2.6.0. The experimental platform was a RTX 2060 12GB GPU. In the training phase, the model performs a large number of calculations, such as gradient calculations and parameter updates, so it imposes certain requirements on the memory size and computing performance of the GPU. In addition, the overhead of the model weights, gradient propagation, optimizer parameters, input data and their labels, intermediate calculations, temporary buffers, hardware, and dependency libraries also consume some memory. Taking the TBNN model constructed in this study as an example, if the 1×1024 energy spectrum is regarded as a sequence of length 1024 in the input phase, then when the embedding operation is performed on it (mapping the data to a high-dimensional space) [48], the shape of the model input is transformed into 1024×200 (taking the 200-dimensional embedding as an example). During the test, a 12 GB GPU prompts an “Out Of Memory (OOM)” error; that is, the memory required to train the model exceeds the current graphics card resources. Therefore, the input

sequence was set to a 32×32 shape, and the embedding process was directly skipped. By monitoring the memory status of the GPU, we can conclude that the graphics card occupied approximately 1.5 G (the maximum value in all models) during the training phase.

The TBNN model designed in this study adopts a structure with 1 to 8 encoder–decoder layers, and the number of attention heads was set to 2^n ($n = 0, 1, 2, 3$). The models built for comparison included a BP with four hidden layers, a CNN with two convolutional layers and one fully connected layer, a ResNet with three residual blocks, and a unidirectional LSTM model with two hidden layers, as shown in Fig. 6.

The task of identifying the eight radioactive nuclides in this study can be regarded as a multi-label task; therefore, the loss function for network model training was a binary cross-entropy function (a measure of the difference between two probability distributions, often used in binary classification problems). The weight initialization method for all models was random initialization. The adaptive moment estimation (Adam) algorithm was used as the optimization function. Adam combines the driven gradient descent and root-mean-square prop (RMSProp) algorithms and can reduce the number of iterations required to reach the optimal value and improve the ability of the optimization algorithm.

3.2.2 Parameters tuning of TBNN model

Learning rate To avoid overfitting or gradient explosion of the model while ensuring a certain convergence speed, we set different learning rates for testing. The learning rates were set to 0.01, 0.005, 0.003, 0.001, 0.0008, and 0.0005, respectively; the number of epochs, attention head count,

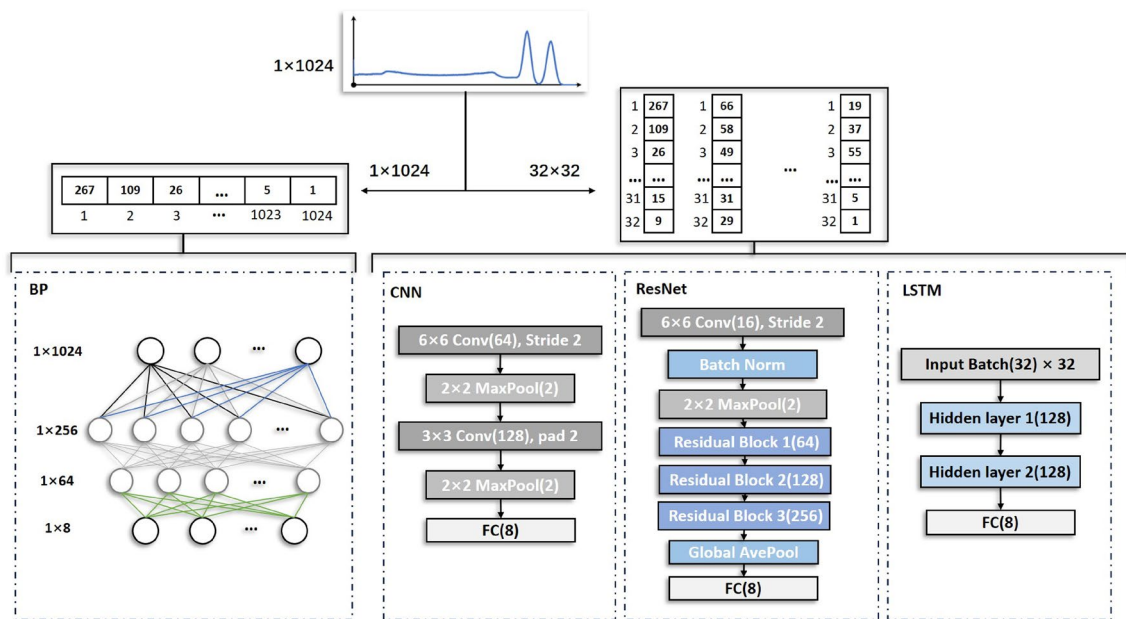


Fig. 6 Construction of other models including BP, CNN, ResNet, and LSTM for comparison

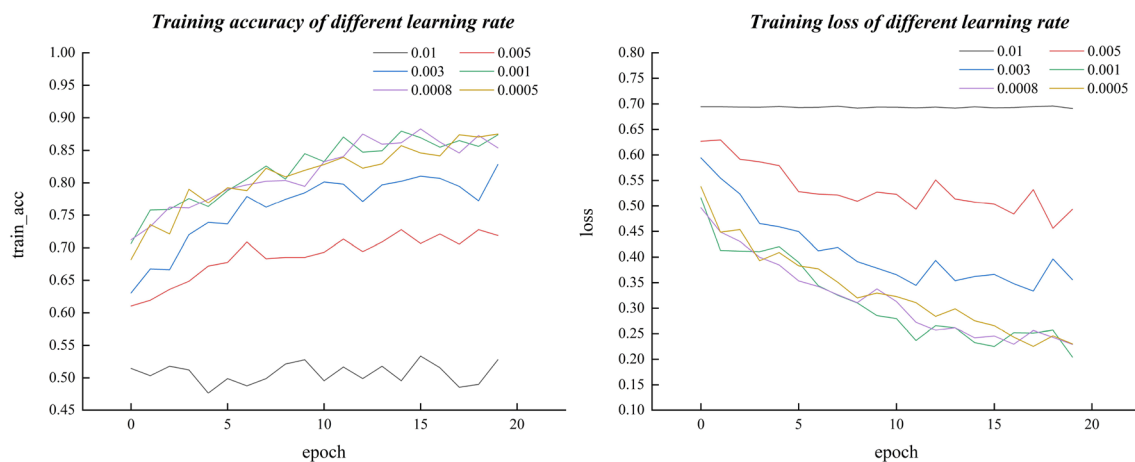


Fig. 7 (Color online) Training curves for different learning rates of the transformer-based neural network. The left and right subfigures show the changes in training accuracy and training loss, respectively, over 20 epochs

and layer count were set to 20, 4, and 4, respectively, and the test results are shown in Fig. 7.

When the learning rate was set to 0.01, the model did not converge, indicating that the initial learning rate was too large; therefore, it was appropriate to reduce the learning rate to try to obtain the optimal initial learning rate. A better convergence speed was obtained when the learning rate was less than 0.001. Table 1 lists the test accuracy for different learning rates within 20 training steps. Therefore, subsequent training of this model was performed with the learning rate set to 0.0008.

Number of attention heads and layers Next, we analyzed the recognition rate of TBNN models composed of different numbers of attention heads and transformer layers. We set the learning rate to 0.0008 and number of epochs to 20 (not considering the slowdown in convergence speed due to increased model parameters). Because the parameter

Table 1 Test accuracy for different learning rates (20 epochs)

lr	0.01	0.005	0.001	0.0008	0.0005
Accuracy	0.6%	15.1%	75.2%	78.3%	77.1%

initialization method of the model was set to random initialization, all models were trained three times, and the best recognition rate was used as the final model for horizontal comparison. The best-performing model for each combination of the attention head count and network layer count was selected to obtain the results, as shown in Fig. 8. Note that the surface in the figure was obtained through linear fitting of the measured points, which can intuitively display the accuracy trends with parameter changes.

From the perspective of changing the number of network layers with a fixed attention head count within 20 training cycles, as the number of network layers changes, models with different attention head counts can all achieve an accuracy of over 80%, and they all reach their current best when there are two to four layers in the network. However, from the perspective of changing the attention head count with a fixed number of network layers within 20 training cycles, an increase in the attention head count does not significantly affect network accuracy.

From Fig. 8, it can be concluded that optimal accuracy is achieved when the number of attention heads is two and the number of layers is four. For models with deeper network layers and more attention heads, owing to a surge in the number of hyperparameters, it is difficult for the model to achieve ideal accuracy within 20 cycles. Simultaneously, the use of a large number of hyperparameters makes network training more time-consuming and difficult to converge. Table 2 lists the time costs of training networks with different layer counts and attention head counts. Training time was positively correlated with both.

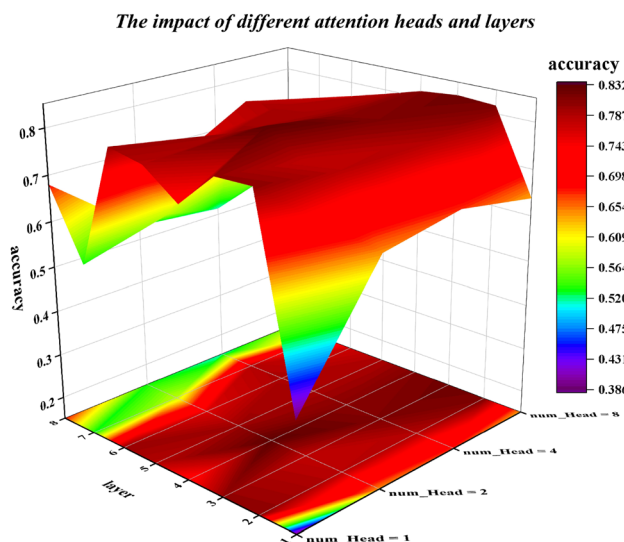


Fig. 8 (Color online) The impact of selecting numbers of different attention heads and layers on accuracy ($lr = 0.0008$, $epoch = 20$). Here, “layer” can only take integers from 1 to 8

Table 2 The time (sec) spent for different numbers of attention heads and network layers

Layer(hyperparameters)	Head				
	1	2	4	8	16
1(186700)	21.5	21.5	21.8	22.8	28.1
2(336016)	31.6	32.8	34.5	37.7	46.9
4(634648)	54.3	56.7	60.1	66.7	84.3
8(1231912)	98.2	103.3	110.1	123.6	160.1

Epoch The aforementioned training process exceeded 20 training cycles in all cases. Models with smaller parameter quantities may reach their optimal fitting ability within 20 cycles; however, deeper models may be far from convergent. A reasonable epoch can give the model good predictive ability without overfitting. Therefore, the next step was to set the epochs to 200 to test the convergence accuracy of the different models. According to the results in Sects. 1 and 2, models named “(num_Head, num_Layer)” were selected, including (2,2), (2,4), (2,8), and so on. Each model was trained three times by taking the step length when each model reached optimal accuracy; the final step length results are shown in Table 3 and Fig. 9. Note that the purpose of Fig. 9 is similar to that of Fig. 8. The surface in the figure was obtained through linear fitting of the measured points, which can visually display the accuracy trends with parameter changes.

At 20 training cycles, the models with smaller parameter values (smaller numbers of attention heads and model layers) achieved higher accuracy than models with larger parameter values in a shorter time span. When the number of training cycles was increased to 200, models with larger parameter values were sufficiently trained, and their accuracy also increased to above 95%. However, increasing the model’s parameter values will not continuously increase its accuracy, as shown in Fig. 9. When the number of attention heads was four and the number of layers was four, the model’s recognition rate reached its highest value of 98.7%. This might be because an excessive number of hyperparameters increase unnecessary or redundant connections in the network layer, which affects the model’s ability to extract data features and increases the training cost. However, this does not mean that models with fewer hyperparameters often achieve superior results. The model with parameters of (4, 4)

Table 3 Number of epochs used to train each model to achieve optimal accuracy

num_Head	num_Layer		
	2	4	8
2	136	179	180
4	129	130	186
8	66	154	178

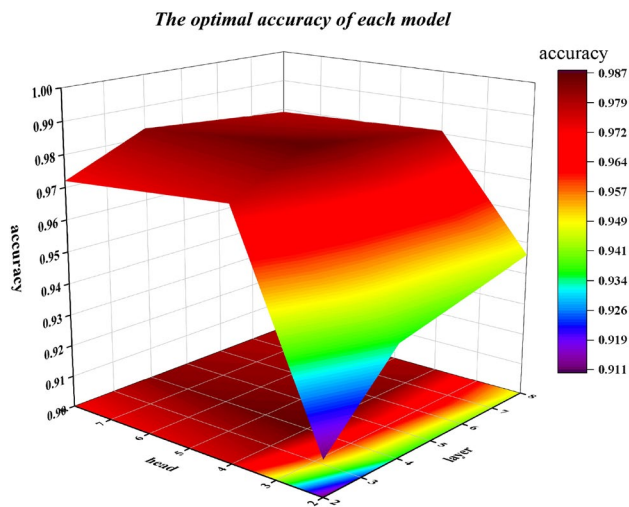


Fig. 9 (Color online) The optimal accuracy that each model can achieve within 200 epochs. Here, “layer” can only take integers from 2 to 8, and “head” can only take values of 2, 4, or 8

exhibited the best nuclide recognition for the database established in this paper and was used for comparison with other neural network models, as described in the next section.

3.2.3 Comparison with other neural network models

The four models shown in Fig. 6 were trained for comparison with the TBNN model proposed in this paper. For BP, which has fewer network nodes, a large step size can easily lead to overfitting. Therefore, the learning rate was set to 0.01. For the CNN, ResNet, and LSTM, the learning rate was set to 0.0001. The epoch number for each model was set to 200. The network training loss and accuracy are shown in Fig. 10.

All of these models can converge quickly (the training loss decreases exponentially and steadily during training and can be reduced to approximately 0.1 over 50 epochs); the final nuclide identification accuracy of each model is shown in Table 4. Overall, the accuracy of the identification method based on neural networks exceeded 90%. BP has a simple structure and can stabilize within 50 epochs;

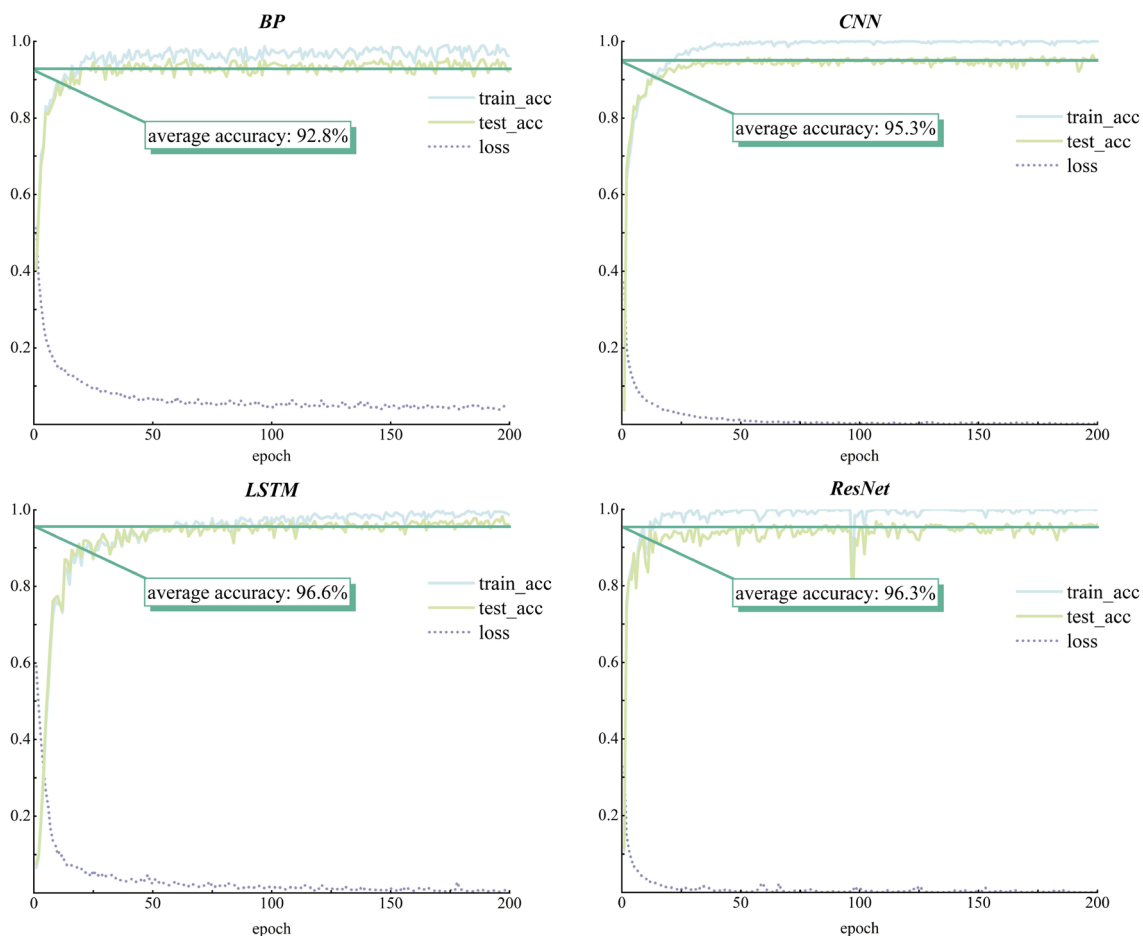


Fig. 10 Training curves for different learning rates of models including BP, CNN, ResNet, and LSTM for comparison. The average accuracy was obtained after each model training process became basically stable (epoch ≥ 150)

Table 4 The recognition rate of each model (training three times)

	BP (%)	CNN (%)	ResNet (%)	LSTM (%)	TBNN (%)
Single nuclide accuracy	99.1	99.3	99.4	99.3	99.3
Total nuclide accuracy	92.8	95.3	96.3	96.6	98.7

Single nuclide accuracy refers to the recognition rate of the model for eight types of nuclides, while total nuclide accuracy refers to the correct recognition of all nuclides contained in a spectrum

however, it easily reaches a local optimum, resulting in no further improvement in the final accuracy. The “dropout layer” can effectively alleviate the phenomenon of BP overfitting, and the worst accuracy rate of BP without the dropout layer after training was 83%. This shows that BP has the problem of falling into local optima. After adding a dropout layer and setting the dropout rate to 0.2, BP’s convergence accuracy reached 92.8%. The CNN and LSTM networks had better global optimization capabilities than BP, and all three training methods converged stably at approximately 95%. Compared with the CNN, the RNN has the disadvantage that it cannot be parallel, which is an important factor that causes its longer training cycle. The step length of LSTM (which is based on the RNN) is longer than that of the RNN, and the corresponding forward- and back-propagation steps are also increased, resulting in greater training resource consumption. Therefore, the LSTM requires a longer training cycle. ResNet, as an improved CNN model, allows deeper convolutions to be fully trained. Its ability to extract data features is better than that of a CNN, and the overall recognition rate increased by approximately 1%. The TBNN model proposed in this paper has a hyperparameter magnitude similar to that of ResNet, both of which are on the order of millions. Within 20 training cycles, the recognition rate is approximately 83%. However, as the number of training cycles increased, the potential of TBNN was revealed. The optimal recognition rate reached 98.7%, which is at least 2.4% higher than that of the other models. This indicates that the transformer network structure paradigm has significant potential for nuclide identification.

4 Conclusion

A nuclide identification model based on the transformer is proposed for the first time in this paper. Unlike traditional neural network architectures such as BP and CNN, this study explores the potential of the encoder–decoder structure paradigm based on the self-attention mechanism in the field of artificial intelligence nuclide identification. In the field of NLP, the numbers of hyperparameters of large language models have reached hundreds of millions. The input of the transformer is usually a sequence; therefore,

this study converts the 1024-channel one-dimensional spectrum into a 32×32 spectrum sequence as the input of the TBNN. Experimental verification demonstrated that converting the spectrum into a sequence is an effective processing method that retains all data of the spectrum and does not generate excessive model nodes; thus, the model can be trained under limited graphics card resources.

We established a database for eight industrial radioactive nuclides: ^{241}Am , ^{192}Ir , ^{226}Ra , ^{133}Ba , ^{60}Co , ^{57}Co , ^{137}Cs and ^{152}Eu , and established four representative neural network models: BP, CNN, ResNet, and LSTM, to verify the effect of the new model proposed in this paper. When the number of attention heads was four and the number of layers was four, the TBNN model achieved the best nuclide identification effect on the dataset established in this study, with an identification rate of 98.7%. Based on the comparison results, it is inferred that ability of the TBNN to extract data features is not weaker than that of traditional neural network paradigms. It is also worth noting that in the recognition task of 8 nuclides, TBNN with four attention heads and four network layers achieved the best results, and increasing attention heads and network layers did not significantly reduce recognition rate. This indicates that in more complex tasks, such as nuclide identification tasks on databases with richer nuclide types, deeper TBNN models still be highly competitive, which is a direction for future research. Overall, our research demonstrates the effectiveness of introducing transformer models into the field of artificial intelligence for nuclide identification.

Acknowledgements We thank the Sichuan Provincial Key Laboratory of Applied Nuclear Techniques in Geosciences and CDUT Team 203 for their English language review. We thank Professors Liang-Quan Ge, David Cohen, Sheng-Qing Xiong, and Si-Chun Zhou for their support.

Author Contributions All authors contributed to the study conception and design. Material preparation, data collection, and analysis were performed by Guo-Qiang Zeng, Sheng Lv, Chu-Yang Luo, and Fei Li. The first draft of the manuscript was written by Guo-Qiang Zeng, Sheng Lv, Chu-Yang Luo, Ying-Zi Wen, Xiao-Hui Liu, and Fei Li, and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Data availability The data that support the findings of this study are openly available in Science Data Bank at <https://cstr.cn/31253.11.sciencedb.j00186.00252> and <https://www.doi.org/10.57760/sciencedb.j00186.00252>.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. X. Li, C. Dong, Q. Zhang et al., Research and design of a rapid nuclide recognition system. *J. Instrum.* **17**, T06008 (2022). <https://doi.org/10.1088/1748-0221/17/06/T06008>
2. A.W. Ajlouni, M.M. Alnairi, K.S. Albarkaty et al., Nuclear security in public events. *J. Radiat. Res. Appl. Sci.* **16**, 100572 (2023). <https://doi.org/10.1016/j.jrras.2023.100572>
3. H. Xu, X. Ai, Y. Wang et al., Ground radioactivity distribution reconstruction and dose rate estimation based on spectrum deconvolution. *Sensors-basel.* **23**, 5628 (2023). <https://doi.org/10.3390/s23125628>
4. S. Qi, W. Zhao, Y. Chen et al., Comparison of machine learning approaches for radioisotope identification using nai (ti) gamma-ray spectrum. *Appl. Radiat. Isotopes* **186**, 110212 (2022). <https://doi.org/10.1016/j.apradiso.2022.110212>
5. J. Suto, S. Oniga, Efficiency investigation from shallow to deep neural network techniques in human activity recognition. *Cognit. Syst. Res.* (in Chinese) **54**, 37–49 (2019). <https://doi.org/10.1016/j.cogsys.2018.11.009>
6. F. Li, Z.X. Gu, L.Q. Ge, Review of recent gamma spectrum unfolding algorithms and their application. *Results Phys.* **13**, 102211 (2019). <https://doi.org/10.1016/j.rinp.2019.102211>
7. F. Li, Z.Y. Cheng, C.S. Tian, Progress in recent airborne gamma ray spectrometry measurement technology. *Appl. Spectrosc. Rev.* **56**(4), 255–288 (2021). <https://doi.org/10.1080/05704928.2020.1768107>
8. X.Z. Li, Q.X. Zhang, H.Y. Tan et al., Fast nuclide identification based on a sequential bayesian method. *Nucl. Sci. Tech.* **32**, 143 (2021). <https://doi.org/10.1007/s41365-021-00982-z>
9. Z.J. Yang, W.B. Wang, P. Huang et al., Local weighted representation based linear regression classifier and face recognition. *Comput. Sci.* **48**, 351–359 (2021). <https://doi.org/10.11896/jsjxk.210100173>
10. J.M. Wang, C. Land-Zhang, Rapid detection of weak radionuclide in moving target. *Inf. Technol. Netw. Secur.* **37**, 137–139 (2018). <https://doi.org/10.19358/j.issn.2096-5133.2018.03.033>
11. Y.G. Huo, H.Z. An, Y.L. Wu et al., Fast radioactive nuclide recognition method study based on pattern recognition. *Nucl. Electron. Detect. Technol.* **34**, 51–53 (2014). <https://doi.org/10.3969/j.issn.0258-0934.2014.01.013>
12. H.L. Liu, The study of radioactive nuclide identification method based on fuzzy decision tree. Master's thesis, Southwest University of Science and Technology. (2018). <https://d.wanfangdata.com.cn/thesis/D01476997>
13. X.B. Xie, H. Zhang, The study of fast nuclide identification method with svm. *Comput. Knowl. Technol.* pp. 355–358 (2014). <https://d.wanfangdata.com.cn/periodical/dnzsyjs-itrzyksb201402044>
14. J.M. Zhang, H.B. Ji, X.H. Feng et al., Nuclide spectrum feature extraction and nuclide identification based on sparserepresentation. *High Power Laser Particle Beams* (in Chinese) **30**, 046003 (2018). <https://doi.org/10.11884/hplpb201830.170435>
15. J.S. Ren, J.M. Zhang, K.P. Wang, Radioactive nuclide identification method based on svd and svm. *Ordinance Ind. Autom.* (in Chinese) **36**, 50–53 (2017). <https://doi.org/10.7690/bgzd.2017.05.014>
16. S. Zhou, Research on nuclide recognition method based on fuzzy decision tree with multi-category weight label. Master's thesis, Southwest University of Science and Technology. <https://d.wanfangdata.com.cn/thesis/D02661842> (2022)
17. G. Juan, *The principle and simulation example of artificial neural network*, (Beijing: China Machine Press, 2003), <https://www.zhangqiaokeyan.com/book-cn/081503037434.html>
18. J. Wang, W. Gu, H. Yang et al., Analytical method for γ energy spectrum of radioactive waste drum based on deep neural network. *Nucl. Tech.* (in Chinese) **45**, 53–59 (2022). <https://doi.org/10.11889/j.0253-3219.2022.hjs.45.040501>
19. N. He, H.Y. Lv, B. Wang et al., Nuclide identification method based on artificial neural network. *Ordinance Ind. Autom.* (in Chinese) **41**, 91–96 (2022). <https://doi.org/10.7690/bgzd.2022.03.018>
20. M.Y. Zhu, C.Q. Yuan, Y.F. Liu et al., Research on the fast recognition algorithm of nuclide based on BP neural network. *Nucl. Electron. Detect. Technol.* (in Chinese) **38**, 138–142 (2018). <https://doi.org/10.3969/j.issn.0258-0934.2018.02.026>
21. Y.C. Liu, W. Wang, D.Q. Niu, Nuclide identification and analysis using artificial neural network. *Ordinance Ind. Autom.* **34**, 86–91 (2015). <https://doi.org/10.7690/bgzd.2015.11.022>
22. J. He, X. Tang, P. Gong et al., Rapid radionuclide identification algorithm based on the discrete cosine transform and bp neural network. *Ann. Nucl. Energy* **112**, 1–8 (2018). <https://doi.org/10.1016/j.anucene.2017.09.032>
23. J. Kim, K. Park, G. Cho, Multi-radioisotope identification algorithm using an artificial neural network for plastic gamma spectra. *Appl. Radiat. Isotopes* **147**, 83–90 (2019). <https://doi.org/10.1016/j.apradiso.2019.01.005>
24. Y.C. Liu, H.G. Zhu, Y.Q. Song, Optimize BP neural network by an improved particle swarm optimization to implement nuclide identification. *Ordinance Ind. Autom.* **35**, 88–92 (2016). <https://doi.org/10.7690/bgzd.2016.04.023>
25. K. He, X. Zhang, S. Ren et al., Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770–778 (2016).
26. C. Janiesch, P. Zschech, K. Heinrich, Machine learning and deep learning. *Electron. Mark.* **31**, 685–695 (2021). <https://doi.org/10.1007/s12525-021-00475-2>
27. H.H. Hu, J.M. Zhang, K.P. Wang et al., Application of convolutional neural networks in identification of complex nuclides. *Transd. Microsyst. Technol.* (in Chinese) **38**, 154–156 (2019). [https://doi.org/10.13873/J.1000-9787\(2019\)10-0154-03](https://doi.org/10.13873/J.1000-9787(2019)10-0154-03)
28. D. Liang, P. Gong, X. Tang et al., Rapid nuclide identification algorithm based on convolutional neural network. *Ann. Nucl. Energy* **133**, 483–490 (2019). <https://doi.org/10.1016/j.anucene.2019.05.051>
29. Y. Wang, Z.M. Liu, Y.P. Wan et al., Energy spectrum nuclide recognition method based on long short-term memory neural network. *High Power Laser Particle Beams* (in Chinese) **32**, 106001 (2020). <https://doi.org/10.11884/HPLPB202032.200118>
30. R. Zhao, N. Liu, Low-resolution gamma-ray spectrum analysis using comprehensive training set and deep ResNet architecture. *Nucl. Instrum. Meth. Phys. Res. Sect. A* **1050**, 168135 (2023). <https://doi.org/10.1016/j.nima.2023.168135>
31. J.Y. Guo, Q. Cai, J.P. An et al., A transformer based neural network for emotion recognition and visualizations of crucial EEG channels. *Physica A* **603**, 127700 (2022). <https://doi.org/10.1016/j.physa.2022.127700>
32. F. Alamri, A. Dutta, Implicit and explicit attention mechanisms for zero-shot learning. *Neurocomputing* **534**, 55–66 (2023). <https://doi.org/10.1016/j.neucom.2023.03.009>
33. M. Springenberg, A. Frommholz, M. Wenzel et al., From modern cnns to vision transformers: Assessing the performance, robustness, and classification strategies of deep learning models

- in histopathology. *Med. Image Anal.* **87**, 102809 (2023). <https://doi.org/10.1016/j.media.2023.102809>
34. Y. Xia, Y. Xiong, K. Wang, A transformer model blended with CNN and denoising autoencoder for inter-patient ECG arrhythmia classification. *Biomed. Signal Proces.* **86**, 105271 (2023). <https://doi.org/10.1016/j.bspc.2023.105271>
 35. A. Dosovitskiy, L. Beyer, A. Kolesnikov et al., An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. arXiv e-prints [arXiv:2010.11929](https://arxiv.org/abs/2010.11929) (2020). <https://doi.org/10.48550/arXiv.2010.11929>
 36. C. Si, W. Yu, P. Zhou et al., Inception transformer. *Adv. Neur. In.* **35**, 23495–23509 (2022)
 37. R. Czarwinski, W. Weiss, Safety and security of radioactive sources-international provisions. *Kerntechnik* **70**, 315–321 (2005). <https://doi.org/10.3139/124.100262>
 38. V. Smolyar, V. Tarasov, A. Mileva et al., Geant4 simulation of the moderating neutrons spectrum. *Radiat. Phys. Chem.* **212**, 111151 (2023). <https://doi.org/10.1016/j.radphyschem.2023.111151>
 39. L. Zhe, L. Min, S. Rui et al., Monte carlo simulation and gaussian broaden techniques for full energy peak of characteristic x-ray in edxrf. *Nucl. Tech.* (in Chinese) **35**, 911–915 (2012)
 40. M.H. Guo, T.X. Xu, J.J. Liu et al., Attention mechanisms in computer vision: a survey. *Comput. Vis. Media* **8**, 331–368 (2022). <https://doi.org/10.1007/s41095-022-0271-y>
 41. A. Vaswani, N. Shazeer, N. Parmar et al., Attention is all you need. arXiv e-prints [arXiv:1706.03762](https://arxiv.org/abs/1706.03762) (2017). <https://doi.org/10.48550/arXiv.1706.03762>
 42. C. Van Hiep, D.T. Hung, N.N. Anh et al., Nuclide identification algorithm for the large-size plastic detectors based on artificial neural network. *IEEE Trans. Nucl. Sci.* **69**, 1203–1211 (2022). <https://doi.org/10.1109/TNS.2022.3173371>
 43. H.C. Lee, B.T. Koo, J.Y. Jeon et al., Radionuclide identification based on energy-weighted algorithm and machine learning applied to a multi-array plastic scintillator. *Nucl. Eng. Technol.* **55**, 3907–3912 (2023). <https://doi.org/10.1016/j.net.2023.07.005>
 44. A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks. *Commun. ACM* **60**, 84–90 (2017). <https://doi.org/10.1145/3065386>
 45. S.K. Sahu, R.N. Yadav, Key facial points recognition using resnet. *Mater. Today: Proc.* **66**, 3651–3656 (2022). <https://doi.org/10.1016/j.matpr.2022.07.342>
 46. D. Yilmaz, İ.E. Büyüktaktın, in *Operations Research Forum*, Learning optimal solutions via an lstm-optimization framework. Vol. 4, Springer, p. 48. (2023). <https://doi.org/10.48550/arXiv.2207.02937>
 47. N. Pei, Y. Wu, R. Su et al., Interval prediction of the permeability of granite bodies in a high-level radioactive waste disposal site using lstm-rnns and probability distribution. *Front. Earth Sci.* **10**, 835308 (2022). <https://doi.org/10.3389/feart.2022.835308>
 48. A. Neelakantan, T. Xu, R. Puri et al., Text and Code Embeddings by Contrastive Pre-Training. arXiv e-prints [arXiv:2201.10005](https://arxiv.org/abs/2201.10005) (2022). <https://doi.org/10.48550/arXiv.2201.10005>

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.