

Study Protocol

# Integrating Self-Attention Mechanisms and ResNet for Grain Storage Ventilation Decision Making: A Study

Yuhua Zhu <sup>1,2</sup>, Hang Li <sup>1,2</sup>, Tong Zhen <sup>1,2</sup> and Zhihui Li <sup>1,2,\*</sup>

<sup>1</sup> College of Information Science and Engineering, Henan University of Technology, Zhengzhou 450001, China; yhzhu@haut.edu.cn (Y.Z.); lihang@stu.haut.edu.cn (H.L.); laomaoaichifanqie@163.com (T.Z.)

<sup>2</sup> Key Laboratory of Grain Information Processing and Control, Henan University of Technology, Zhengzhou 450001, China

\* Correspondence: youyou\_70@126.com

**Abstract:** Food security is a widely discussed topic globally. The key to ensuring the safety of food storage is to control temperature and humidity, with ventilation being an effective and fast method for temperature and humidity control. This paper proposes a new approach called “grain condition multimodal” based on the theory of computer multimodality. Under changing external environments, grain conditions can be classified according to different ventilation modes, including cooling ventilation, dehumidification ventilation, anti-condensation ventilation, heat dissipation ventilation, and quality adjustment ventilation. Studying intelligent ventilation decisions helps achieve grain temperature balance, prevent moisture condensation, control grain heating, reduce grain moisture, and create a low-temperature environment to improve grain storage performance. Combining deep learning models with data such as grain stack temperature and humidity can significantly improve the accuracy of ventilation decisions. This paper proposes a neural network model based on residual networks and self-attention mechanisms that performs better than basic models such as LSTM (Long Short-Term Memory), CNN (Convolutional Neural Network), GRU (Gated Recurrent Unit), and ResNet (Residual Network). The model’s accuracy, precision, recall, and F1 scores are 94.38%, 94.92%, 98.94%, and 96.89%, respectively.

**Keywords:** grain security; multimodal; ventilation model; deep learning



**Citation:** Zhu, Y.; Li, H.; Zhen, T.; Li, Z. Integrating Self-Attention Mechanisms and ResNet for Grain Storage Ventilation Decision Making: A Study. *Appl. Sci.* **2023**, *13*, 7655. <https://doi.org/10.3390/app13137655>

Academic Editor: Ángel J. Gutiérrez Fernández

Received: 15 May 2023

Revised: 14 June 2023

Accepted: 25 June 2023

Published: 28 June 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

As technology develops, crop yields have greatly increased [1]. However, due to differences in geography and climate, crop production in some regions is far from meeting the demand. As yield cannot be significantly increased, it is crucial to effectively reduce grain consumption to preserve grain. The key to scientific grain preservation is “temperature control” [2], and the key to maintaining an appropriate temperature for grain is ventilation. Effective ventilation not only prolongs the storage time of grain but also maintains its quality.

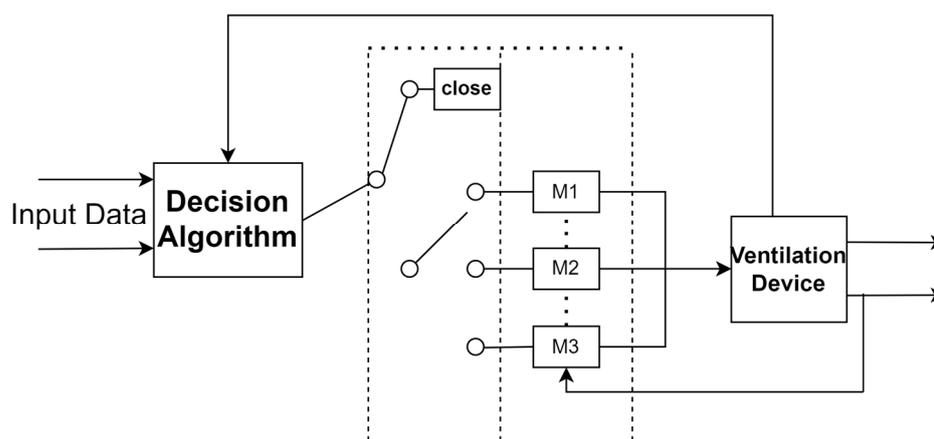
Nowadays, ventilation technology includes temperature-reducing ventilation, precipitation-reducing ventilation, anti-condensation ventilation, heat-dissipation ventilation, and conditioning ventilation [3]. Choosing the appropriate ventilation timing and completing ventilation control in different modes is the key to temperature control. However, currently, grain storage facilities can only detect “three temperatures and two humidities”, which are the temperature of the grain pile, the temperature of the atmosphere in the granary, the temperature of the external atmosphere, the humidity of the atmosphere inside the granary, and the humidity of the external atmosphere. The commonly used ventilation control still relies on human judgment to start and stop ventilation equipment. Faced with the complex and ever-changing environment inside the granary, there is no scientific, accurate, or fast response strategy.

The so-called multi-modal grain status refers to the state of different ventilation conditions under a global perspective, which is divided into temperature-reducing ventilation mode, precipitation-reducing ventilation mode, anti-condensation ventilation mode, heat dissipation ventilation mode, and conditioning ventilation mode.

There has been a rapid development of deep learning and its applications in various fields, such as computer vision [4], speech recognition [5], natural language processing [6], medical diagnosis [7,8], precision agriculture [9], stock market [10], and so on. As deep learning becomes more widespread, its applications in food security have also increased, but there have been relatively few practical applications in grain storage.

Deep learning is a learning algorithm that uses multi-layer neural networks and has strong adaptability, robustness, and learning ability [11,12]. The concept of multimodality originates at the intersection of cognitive science and computer science research. It refers to the acquisition of rich information through multiple sensory channels [13], such as vision, hearing, and touch, and the integration and joint processing of this information to obtain more accurate and comprehensive information and understanding. This multi-modal information processing approach plays an important role in human cognition and communication [14] and has also become an important research direction in fields such as computer vision, speech recognition, and natural language processing.

Based on the theory of computer multimodality, multi-modal grain storage refers to the classification of internal and external environmental conditions of a grain pile under ventilation into different modes. These modes include temperature-reducing ventilation, moisture-reducing ventilation, anti-condensation ventilation, heat dissipation ventilation, and conditioning ventilation. Multi-modal grain status control refers to the control method that changes a series of modes with the environmental changes of the grain pile. Therefore, combining deep learning with the latest multi-modal grain status control theory, researching the decision-making and control strategies of grain storage ventilation mode, achieving balanced grain temperature, preventing water condensation, stopping grain heating, reducing grain moisture, creating a low-temperature environment, improving grain storage performance, and reducing manual operation functions are of practical significance in ensuring grain storage safety [15]. Based on this theory, Figure 1 shows the multimodal division of the grain situation decision-making algorithm.



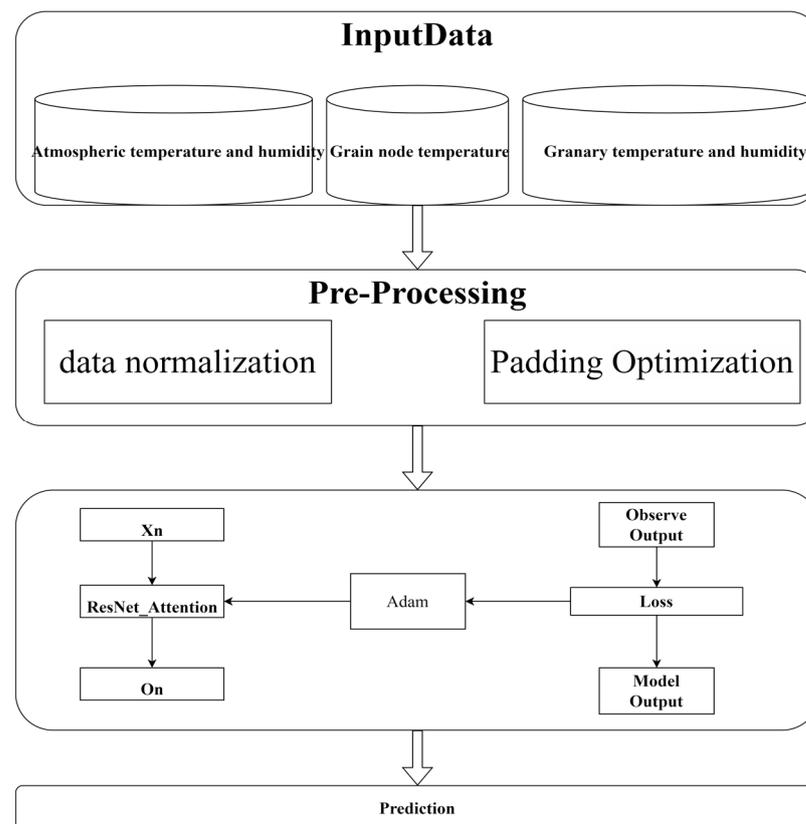
**Figure 1.** Multimodal division of grain situation decision algorithm. M1, M2, and M3 represent Modality 1, Modality 2, and Modality 3, respectively.

We conducted an extensive literature review on grain storage and found limited research in this field. However, we explored cutting-edge knowledge in related areas. One article introduced a ventilation management model for grain storage based on Bayesian networks [16]. The model utilized different factors, such as temperature, humidity, and oxygen concentration, as nodes and used mathematical analysis to determine the probability relationships between them to optimize ventilation management. Additionally, we

searched for a paper about a grain storage loss analysis model based on the decision tree algorithm [17]. By collecting relevant data during the grain storage process, including temperature, humidity, and ventilation information, we constructed a decision tree model. The model yielded important findings, identifying temperature, humidity, and ventilation as the key factors that impact grain storage losses. Effective measures should be taken to control these factors in grain storage management to reduce the occurrence of grain storage losses. These findings provide a theoretical foundation for our experiments.

This article proposes a neural network structure based on the transformer, which relies entirely on attention mechanisms to process input sequences, greatly improving the efficiency and speed of natural language processing and other sequence data tasks [18]. The self-attention mechanism allows the model to attend to different parts of the sequence at different positions, enabling it to gather more global information rather than relying on fixed-size windows as traditional recurrent and convolutional neural networks do. Applying this theory to ResNet (Residual Network) can quickly train neural networks and achieve faster convergence of network models.

Building on these two theories, this paper combines residual networks with self-attention mechanisms, leveraging the advantages of residual networks to avoid the problems of gradient vanishing and exploding and accelerate neural network training. At the same time, self-attention mechanisms are introduced to establish relationships among data autonomously. Figure 2 is a flowchart of a decision algorithm based on the combination of the self-attention mechanism and ResNet (Residual Network).



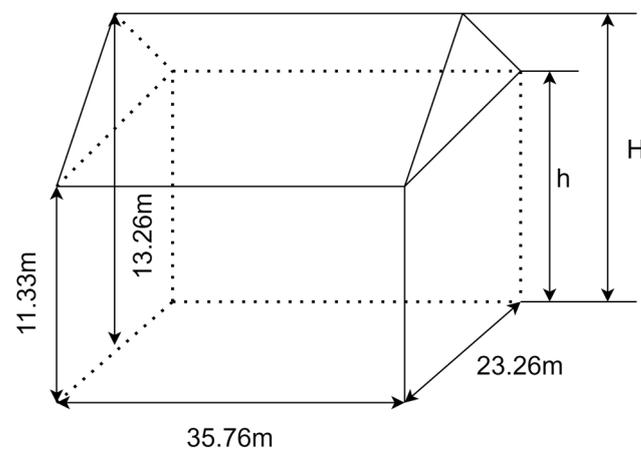
**Figure 2.** Decision algorithm flowchart.

## 2. Materials and Methods

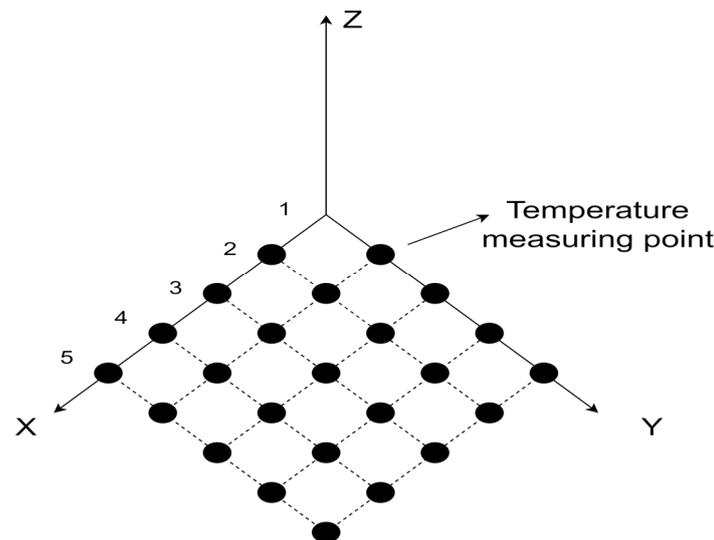
In this section, we first introduced the data collection method and the principles of creating the dataset used in this study, including how to collect grain condition data under different ventilation scenarios and how to preprocess and divide the data.

### 2.1. Data Collection

The granary data used for this experiment was obtained from a granary located in Yushu City, Jilin Province. The storage grain was directly placed beneath the 25th tall bungalow warehouse, as shown in Figure 3. The warehouse had a length of 35.76 m, a width of 23.26 m, an outer length of 36.62 m, an outer width of 25.18 m, an eaves height (h) of 11.33 m, a top height (H) of 13.26 m, and a grain pile height of 8.0 m. Distributed fiber optic temperature measurement technology was used to measure the temperature of grain piles [19], and its distribution is shown in Figure 4. Additionally, the temperature and humidity inside the warehouse and the atmospheric temperature and humidity were measured using the Sensirion digital temperature and humidity sensor model SHTW2. Since the temperature changes in the grain pile are slow, an hourly data collection strategy was employed to transfer the collected data to the MySQL database.



**Figure 3.** Physical model of a large square warehouse.



**Figure 4.** Distribution map of temperature measurement points in the granary.

### 2.2. Dataset Description

The data required for this experiment includes the temperature of each point in the grain pile, the temperature and humidity inside the warehouse, the atmospheric temperature and humidity, and the average temperature inside the grain pile. Table 1 describes the data used in this experiment.

**Table 1.** A detailed table of grain situation data.

N	Feature	Unit of Measurement
1	Average temperature	°C
2	Granary temperature	°C
3	Granary humidity	%
4	Atmospheric temperature	°C
5	Atmospheric humidity	%
6	Grain node temperature	°C

### 2.3. Making a Dataset

#### 2.3.1. Data Preprocessing

The form of the data obtained from Jilin Grain Depot No. 35 is shown in Figure 5, which is an Excel table of one year testing data.

row—layer	Tenth column	Ninth column	Eighth column	Seventh column	Sixth column	Fifth column	Fourth column	Third column	Second column	First column	
0	6—1	-11.20	-8.7	-9.5	-12.1	-7.5	-7.2	-8.4	-9.8	-10.4	-10.8
1	6—2	-7.80	-10.7	-6.1	-5.3	-7.9	-7.3	-7.4	-6.8	-7.5	-7.1
2	6—3	-7.40	-6.6	-6.2	-5.1	-8	-7.7	-6.6	-5.9	-6.6	-6.3
3	6—4	-7.90	-5.5	-6.2	-5.2	-8.2	-7.9	-6.8	-5.6	-6.2	-6.9
4	6—5	-8.40	-5.6	-6.4	-5.4	-7.9	-7.8	-7.2	-5.8	-6.1	-7.6
5	6—6	-6.70	-7.1	-7.4	-6.1	-8.9	-11.1	-5.8	-6.3	-6.1	-6.9
6	6—7	-5.90	-5.1	-3.6	-10.2	-4.1	-4.9	-2	-1.6	-2.2	-4.9
7	5—1	-12.30	-8.4	-10.2	-11.8	-8.4	-7.6	-11.1	-8.1	-9.9	-11.2
8	5—2	-8.90	-7.2	-8.9	-7.4	-8.6	-8.4	-7.4	-7.4	-8.6	-10.7
9	5—3	-8.90	-6.6	-8.5	-6.8	-8.6	-8.6	-6.6	-5.9	-8.2	-11.1
10	5—4	-9.50	-7.4	-8.7	-6.8	-8.8	-8.9	-7.2	-5.8	-8.7	-11.1
11	5—5	-9.90	-8.2	-9.2	-7.1	-9.1	-8.9	-7.8	-6.2	-9.4	-9.6
12	5—6	-7.40	-7.2	-8.1	-7.2	-10.3	-12.7	-6.9	-6.4	-7.4	-9.8
13	5—7	-6.20	-5.6	-5.4	-11.3	-4.5	-6.2	-5.2	-2.9	-4.5	-10.6
14	4—1	-11.20	-7.4	-13.2	-13.6	-8.4	-7.2	-11.3	-8.8	-12.3	-13.2
15	4—2	-10.00	-7.6	-9.7	-7.8	-8.8	-7.6	-5.9	-6.1	-8.5	-11.2
16	4—3	-10.20	-7.2	-9.4	-7.4	-9.1	-8.1	-5	-4.9	-8.2	-11.4
17	4—4	-10.80	-7.8	-10.4	-7.6	-9	-8.4	-5.2	-4.9	-8.7	-11.3
18	4—5	-10.80	-8.6	-11.2	-7.9	-9.1	-8.6	-5.6	-5.2	-9.6	-10.2
19	4—6	-8.80	-7.5	-8.4	-7.6	-10.8	-9.7	-6.2	-6.2	-8.2	-8.8
20	4—7	-9.70	-4.2	-8.9	-12.9	-4.6	-9.2	-2.8	-2.5	-5.2	-7.7
21	3—1	-13.00	-8.9	-11.2	-12.9	-8.2	-7.7	-10.9	-9.3	-9.9	-12.3

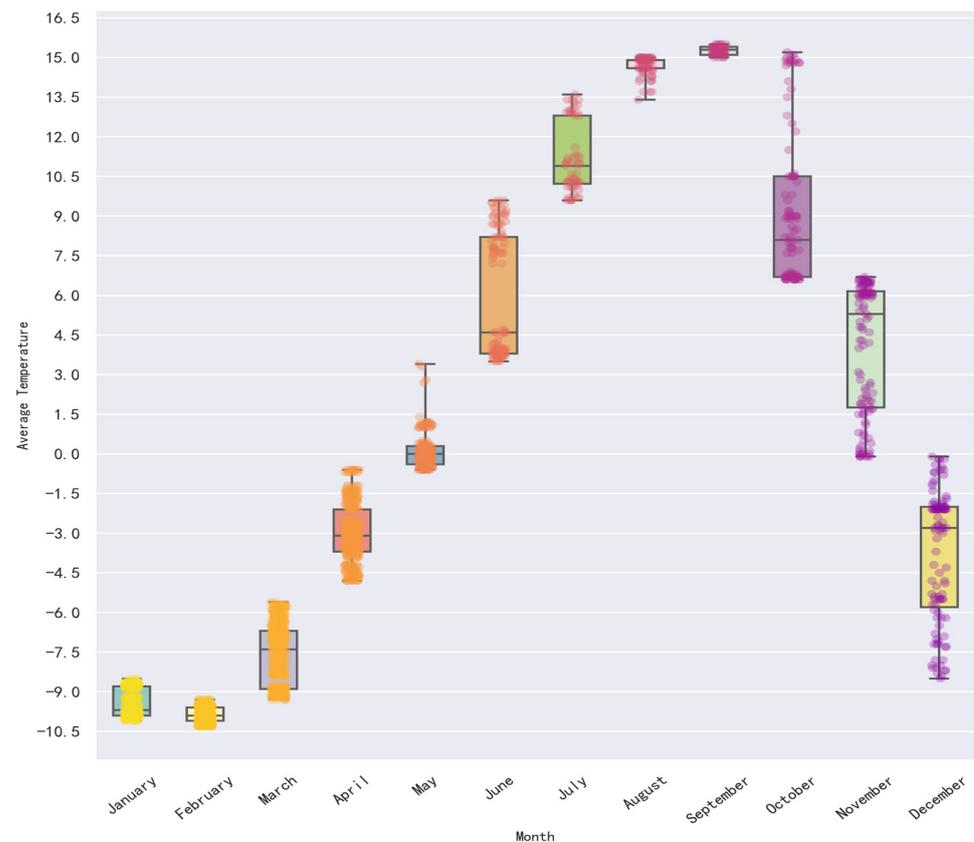
**Figure 5.** Unprocessed grain situation data graph.

Summary and aggregation were performed on one year’s worth of data, which was then plotted on a single table. This table includes the temperature of each point in the grain pile, the temperature and humidity inside the warehouse, the atmospheric temperature and humidity, and the average temperature inside the grain pile. The results are summarized in Table 2 below.

**Table 2.** Grain situation data summary result chart.

Test Time	Grain Node Temperature	Granary_Temp	Granary_Hum	Atmospheric_Temp	Atmospheric_Hum	Average_Temp
1 January 2020: 8:00	−11.2...−9.4	−14.5	54.3	−22.1	73.4	−8.5
...	...	...	...	...	...	...
29 December 2021: 12:00	−5.9...−3.9	−9.8	60.6	−16.4	53.9	−8.5

The unprocessed data was compared and analyzed to observe the distribution of average temperature data for each month and identify any outliers that may have been caused by other incidents. Figure 6 shows the distribution of the average temperature data for each month. The data was then validated to determine whether any outliers were caused by other incidents, and if so, they were removed.



**Figure 6.** Distribution of monthly average temperature data.

After outlier processing, since the data types of temperature and humidity are different, it is necessary to normalize the data to eliminate the impact of data inconsistency on the experiment. To prevent the standardized data from being close to zero and not differentiating the data, we choose z-score normalization, and its formula is as follows:

$$Z = \frac{(X - \mu)}{\sigma}$$

where  $X$  is the sample value, mean is the mean value of the sample data, and standard deviation is the standard deviation of the sample data. After normalization, the resulting  $Z$  value indicates the degree of deviation between the original data and the sample mean:  $Z < 0$  indicates that the data is smaller than the mean, and  $Z > 0$  indicates that the data is larger than the mean.

### 2.3.2. Labeling Data

An excellent deep learning model requires accurate data classification. However, since the humidity collected in the grain depot is relative humidity, labeling directly according to the grain depot ventilation regulations is not possible. Therefore, the grain ventilation equation CAE (Chen–Clayton Approximation Equation) is used to fit the grain equilibrium absolute humidity and grain dew point temperature. During this process, the influence of different grains on the CAE (Chen–Clayton Approximation Equation) equation needs to be considered. Table 3 provides a detailed explanation of the various parameters of the CAE (Chen–Clayton Approximation Equation) equation for different grain categories [20]. Finally, the fitted data is labeled according to the grain depot ventilation regulations, and the formula is as follows:

$$EAH_r = \exp\left\{\frac{\left[\frac{D}{222} \times \left(e^{\frac{B_1 - M}{A_1}} - e^{\frac{B_2 - M}{A_2}}\right) + 0.9845\right] \times \left(1737.1 - \frac{474242}{273 + t}\right) + D \times \left(1 - e^{\frac{B_1 - M}{A_1}}\right) - 68.57}{87.72}\right\}$$

where:

- $EAH_r$ : grain equilibrium absolute humidity, mmHg;
- $M$ : grain moisture content, % (wet basis);
- $t$ : grain temperature;
- $A_1, B_1, A_2, B_2, D$ : the five parameters of the CAE equation.

$$DPT_a = \left\{ \frac{474242}{\frac{474242}{273 + T_a} - 89.1 \times 1g(RH_a) + 410.34} \right\} - 273$$

where:

- $RH_a$ : atmospheric relative humidity, %;
- $T_a$ : atmospheric temperature, °C;
- $DPT_a$ : atmospheric dew point temperature, °C.

**Table 3.** Parameters of the CAE equation for the main grain types.

Classification	Aspiration Type	CAE Equation Parameters				
		$A_1$	$A_2$	$B_1$	$B_2$	$D$
Wheat	Desorption	4.212	4.796	7.493	4.028	202.031
	Adsorption	4.874	4.767	4.671	3.639	201.676
Paddy	Desorption	4.431	4.883	7.758	4.373	205.097
	Adsorption	4.606	4.561	4.918	3.613	202.632
Corn	Desorption	4.393	4.845	7.843	3.858	203.892
	Adsorption	4.812	4.479	4.783	3.799	202.164

#### 2.4. Data Set Partitioning

To divide the processed and labeled data into training, testing, and validation sets, we will use the DataLoader package in Python. Firstly, we will set the batch size to 32 and the random state to 42 to ensure consistent results each time. This will guarantee that the dataset is shuffled in a reproducible manner.

Next, we will create three data loaders: the training data loader, the testing data loader, and the validation data loader. The data loaders will allow us to efficiently load and iterate through the data during model training and evaluation.

The training data loader will be responsible for providing batches of data during the training process. It will randomly sample 70% of the data for training.

The testing data loader will be used to evaluate the model’s performance. It will contain 15% of the data and will be used to assess how well the model generalizes to unseen examples.

The validation data loader will also contain 15% of the data and will be used to finetune the model’s parameters and assess its performance on a separate dataset. This will help ensure that the model’s weights are optimized and prevent overfitting to the training data.

By using the DataLoader package and setting the appropriate parameters, we can create data loaders that provide randomized and rigorous training, testing, and validation data for our model.

## 2.5. Neural Network Model

### 2.5.1. CNN

CNN (Convolutional Neural Network) is a type of deep learning neural network widely used in fields such as computer vision and natural language processing. It consists of convolutional layers, pooling layers, and fully connected layers. The convolutional layer extracts features from the data, while the pooling layer performs downsampling to reduce the number of parameters in the feature map, thereby reducing computational complexity, preventing overfitting, and improving model robustness [21]. The fully connected layer mainly works with the softmax function to normalize the output and obtain the probability of each category, thereby achieving the classification task.

The CNN model used in this experiment is based on VGGNet16, which consists of 13 convolutional layers and 3 fully connected layers. Each convolutional layer uses a  $3 \times 3$  convolutional kernel and the ReLU activation function. The first 12 convolutional layers follow the same configuration, with a  $2 \times 2$  max pooling layer following each layer, using a stride of 2 to reduce the dimensionality of the feature maps. The first two fully connected layers have 4096 neurons each, and the last fully connected layer has 1000 neurons. The output of the last fully connected layer is passed through a softmax operation to convert it into a probability distribution.

### 2.5.2. ResNet

ResNet (Residual Network) is a deep learning neural network composed of multiple residual blocks. Each residual block consists of two main parts: the main path and the skip connection [22]. The main path is composed of a series of convolutional layers, batch normalization layers, and activation functions, which are used for feature extraction of the input signal. The skip connection is a direct connection that adds the input signal directly to the output signal, thus preserving the information of the input signal and allowing it to bypass the convolutional layers in the main path and be directly passed to the subsequent layers. This structure makes the network easier to train, avoids problems such as gradient disappearance and explosion, and makes the network deeper, which improves the accuracy of the network. In this article, ResNet (Residual Network) is used as the basic model, and the network structure of its residual blocks is shown in Figure 7.

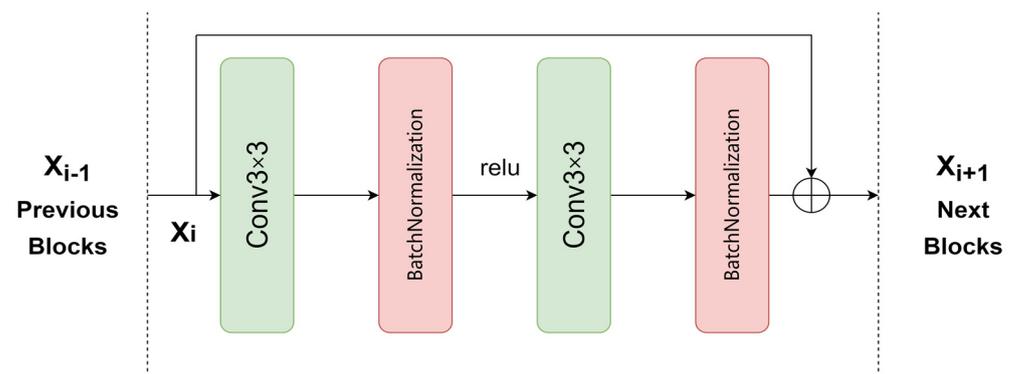


Figure 7. Residual blocks.

### 2.5.3. GRU

GRU (Gated Recurrent Unit) is a type of recurrent neural network that consists of an update gate, a reset gate, and a hidden state vector [23,24]. The GRU model used in this study consists of 16 hidden layers and 2 GRU layers. The hidden layers are responsible for computing the hidden state at the current time step based on the previous information in the input sequence and the current input. The hidden state contains information from previous time steps and is passed to the model at the next time step, enabling the model to capture the temporal dependencies in the sequence data.

### 2.5.4. LSTM

LSTM stands for Long Short-Term Memory, which is a type of recurrent neural network [25,26]. It was designed to overcome the vanishing gradient problem in traditional RNNs and allow for the processing of long-term dependencies. LSTMs use a series of gates, including an input gate, a forget gate, and an output gate, to selectively allow information to flow through the network and control the memory stored in the hidden state. This enables LSTMs to selectively remember or forget information from previous time steps as needed, making them well-suited for tasks such as language modeling, speech recognition, and handwriting recognition. The LSTM utilized in this article consists of 16 hidden layers and 2 LSTM layers. The purpose of the hidden layers is to introduce non-linear mappings, thereby enhancing the expressive capacity of the network. LSTM, a special type of recurrent neural network (RNN) architecture, addresses the issue of vanishing and exploding gradients encountered by traditional RNNs by incorporating gate mechanisms. The LSTM layer exhibits memory capabilities when processing sequential data, enabling effective handling of long-term dependencies.

### 2.5.5. Self-Attention

Self-attention is a mechanism used in deep learning to balance the importance of different parts of a sequence when predicting or generating the next element. Self-attention allows the model to focus on different parts of the input sequence during prediction without using recursive or convolutional operations.

The core idea of self-attention is to calculate the influence of each element on other elements by computing the associated weights. This weight can be calculated using various methods, but the most common approach is to use dot-product attention. This measures the degree of association between a query vector and a key vector by calculating their dot product and using it as the attention weight. Figure 8 for self-attention is as follows:

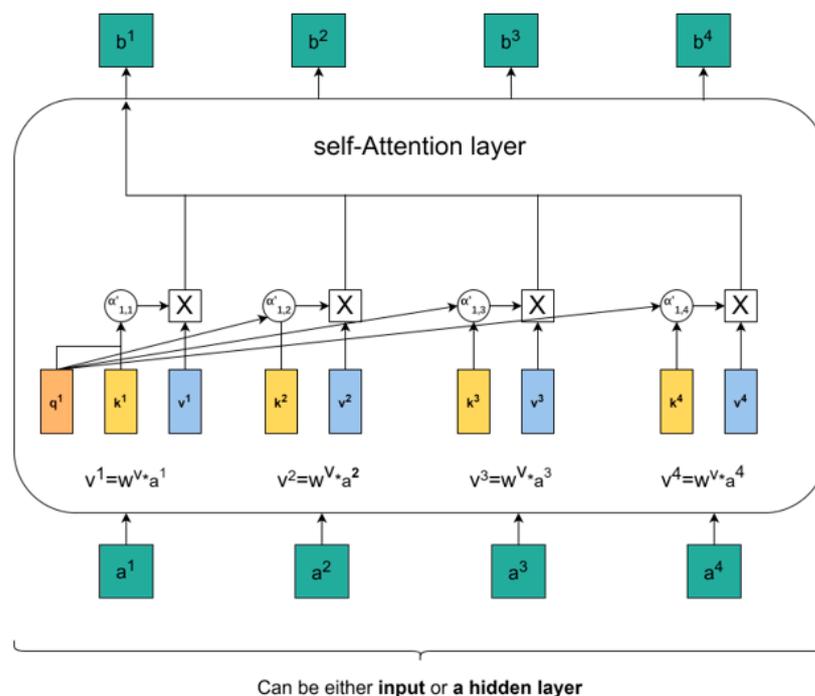
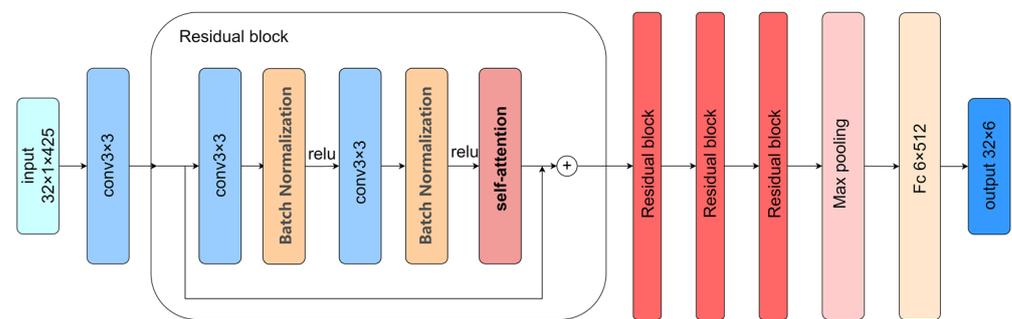


Figure 8. Self-attention.

### 2.5.6. ResNet\_Attention

This network is a 1D convolutional neural network based on the ResNet architecture. It is mainly used for grain quality sequence data classification tasks, where the input is one-dimensional grain quality sequence data. The network includes a convolutional layer

(Conv3×3), a maximum pooling layer, four residual blocks, and a self-attention layer. Each residual block contains two convolutional layers (Conv3×3) and a self-attention layer. Inside the residual block, the input signal is passed through two 1D convolutional layers (Conv3×3) with the same kernel size, followed by a self-attention layer for feature extraction and adaptive feature weighting. The self-attention layer adjusts the weights of the feature vectors by calculating attention weights so that important features get larger weights while unimportant features get smaller weights. Self-attention can help the network better understand the long-term dependencies and importance of the input signal, thus improving classification performance. Finally, after global average pooling, a fixed-size feature vector is obtained, which is then fed to a fully connected layer to output the classification result for the grain quality data. The neural network structure used in this experiment is shown in Figure 9.



**Figure 9.** ResNet\_Attention.

## 2.6. Evaluation Criteria

This article explores the use of residual neural networks with self-attention mechanism for making ventilation decisions in granaries under multiple modalities. Five evaluation metrics, including loss, accuracy, precision, F1 score, and recall, are used to compare the performance of the proposed model against other models.

### 2.6.1. Cross-Entropy Loss

The cross-entropy loss is a commonly used loss function in deep learning, especially in classification tasks. In classification tasks, we want the model to assign each input sample to the correct category. The purpose of the cross-entropy loss is to measure the difference between the predicted and actual categories. Its formula is as follows:

$$J(\theta) = -\frac{1}{n} \sum_{i=1}^n \sum_{j=1}^m y_{ij} \log(p_{ij})$$

where:

$n$ : represents the total number of samples;

$m$ : represents the total number of categories;

$y_{ij}$ : represents the true label of the  $i$ th sample, which is 1 if the  $i$ th sample belongs to the  $j$ th category, or 0 otherwise;

$p_{ij}$ : represents the probability that the model predicts the  $i$ th sample belongs to the  $j$ th category.

### 2.6.2. Accuracy

Accuracy is one of the most commonly used metrics for comparing model performance, and is used to evaluate the accuracy of a model's classification. However, accuracy is not a universal evaluation metric. In some cases, it may be misleading because it only considers

the number of correctly classified samples while ignoring the errors made by the model on misclassified samples. Its formula is as follows:

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

where:

*TP* (True Positive): predictions that are positive and that are actually positive;

*TN* (True Negative): predictions that are negative and that are actually negative;

*FP* (False Positive): predictions that are positive but are actually negative;

*FN* (False Negative): predictions that are negative but are actually positive.

### 2.6.3. Precision

Precision is used to evaluate the proportion of true positive samples among all samples that the model predicts as positive, so it can be used to measure the prediction accuracy of the model. Its formula is as follows:

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

### 2.6.4. Recall

Recall, also known as sensitivity, is the proportion of true positive samples that are correctly identified by the classifier among all positive samples. It can be understood as the ability of the model to correctly identify positive samples and is also referred to as the model's "true positive rate" or "hit rate". Its formula is as follows:

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

### 2.6.5. F1 Score

F1 score is a metric that considers both precision and recall of a classification model and is commonly used to evaluate the performance of binary classification models. Its formula is as follows:

$$F1Score = 2 \left( \frac{precision \times recall}{precision + recall} \right)$$

### 2.6.6. Confusion Matrix

The confusion matrix is a tool to evaluate classification models. It is a matrix that shows the cross-occurrence of actual and predicted classifications. Rows in the matrix represent the actual classes, while columns represent the predicted classes. By computing the confusion matrix, we can obtain evaluation indicators such as accuracy, precision, recall, and F1 score for the model.

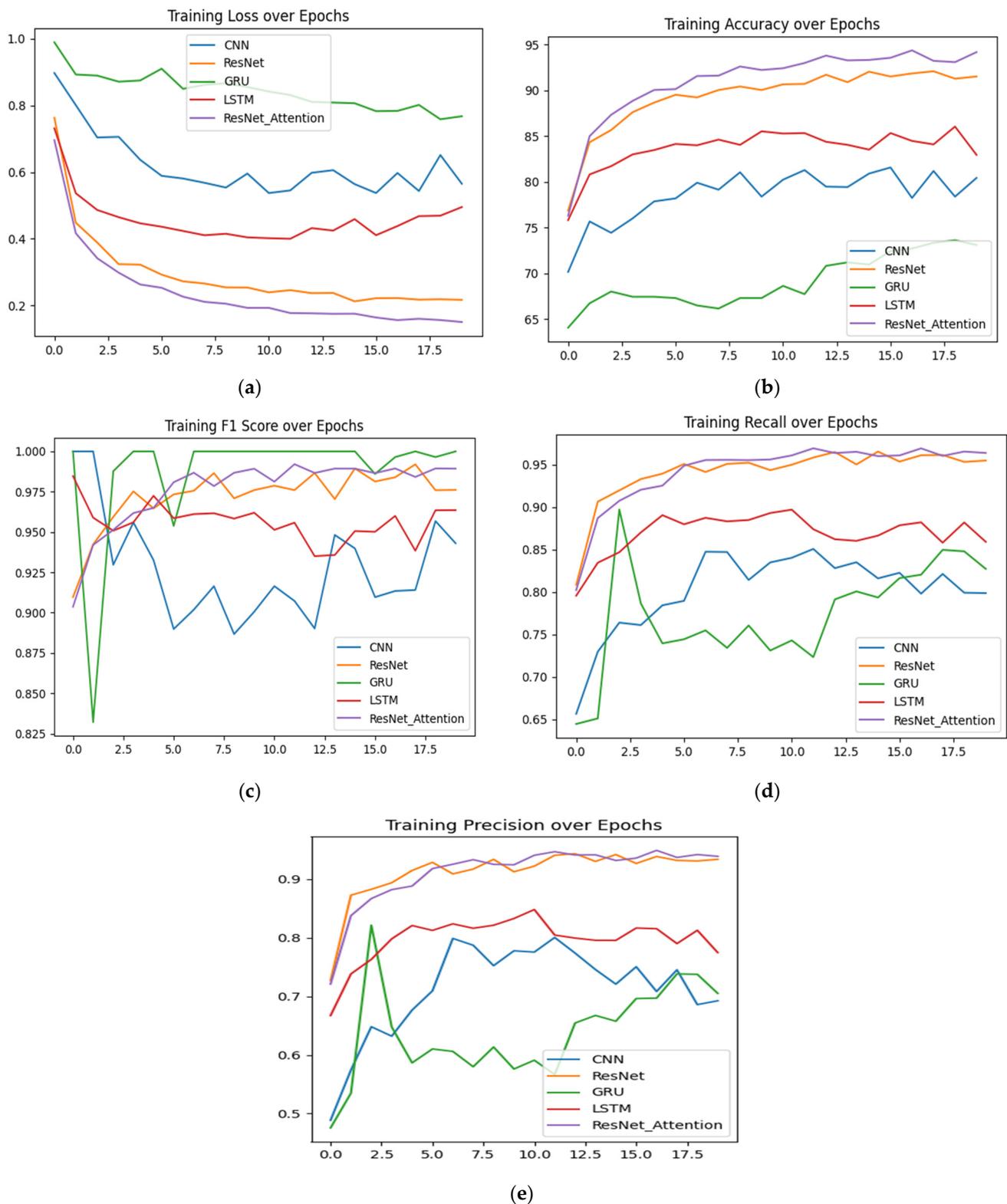
## 3. Results

In this section, we compare the evaluation metrics of the LSTM, GRU, ResNet, and CNN models, and evaluate their accuracy rates and other indicators for different ventilation categories.

### 3.1. Model Training and Test Results

#### 3.1.1. Model Training

To validate the superiority of the ResNet\_Attention model over other sequence processing models, comparative experiments were conducted using the same training dataset. The initial learning rate for all models was set to 0.001, and the model weights were randomly initialized. A batch size of 32 was used, and the models were trained for 20 epochs. The results are shown in Figure 10.



**Figure 10.** Comparison of different evaluation results for different models. (a) Training loss curve. (b) Training accuracy curve. (c) Training F1 score curve. (d) Training recall curve. (e) Training precision curve.

### 3.1.2. Model Testing

By using the test set, it was verified that ResNet with the self-attention mechanism had better performance on the grain situation sequence data. The specific numerical

values for the evaluation indicators, including accuracy, loss, and F1 score, showed that ResNet\_Attention performed better, as shown in Table 4.

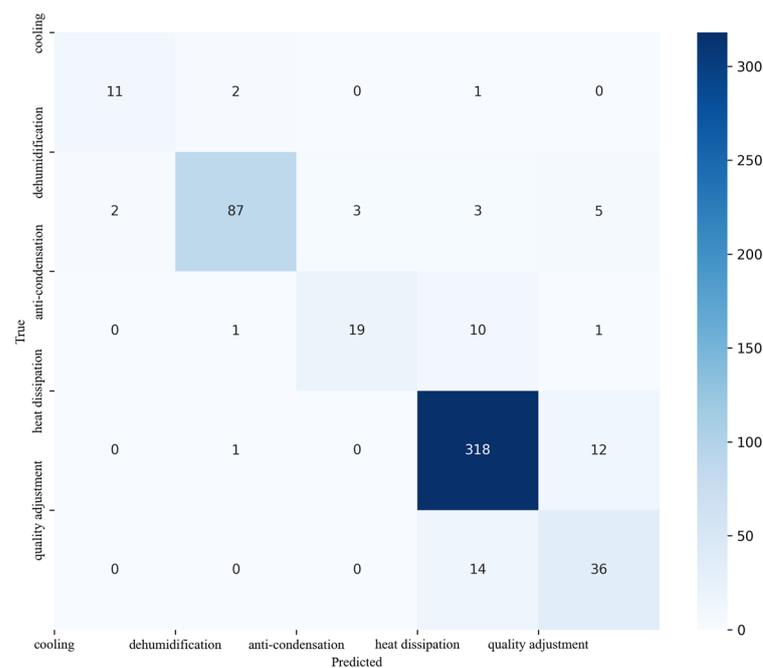
**Table 4.** Evaluation form for different model testing.

Model	Accuracy	Precision	Recall	F1 Score	Cross-Entropy Loss
CNN	0.84	0.72	0.64	0.66	0.506667
GRU	0.75	0.89	0.31	0.32	0.738257
LSTM	0.87	0.87	0.56	0.55	0.346726
ResNet	0.88	0.78	0.80	0.78	0.438510
ResNet_Attention	0.91	0.88	0.77	0.81	0.228690

### 3.2. Comparison of Identification Results

#### 3.2.1. Compare the Decision Results under Different Ventilation Modes

Based on grain data, we tested the accuracy, precision, recall, F1 score, and other indicators of cooling ventilation, dehumidification ventilation, anti-condensation ventilation, heat dissipation ventilation, and quality adjustment ventilation. The results were calculated based on the confusion matrix in Figure 11. The accuracy of each category of the model was generally above 90%, with the comparison results for each category shown in Table 5.



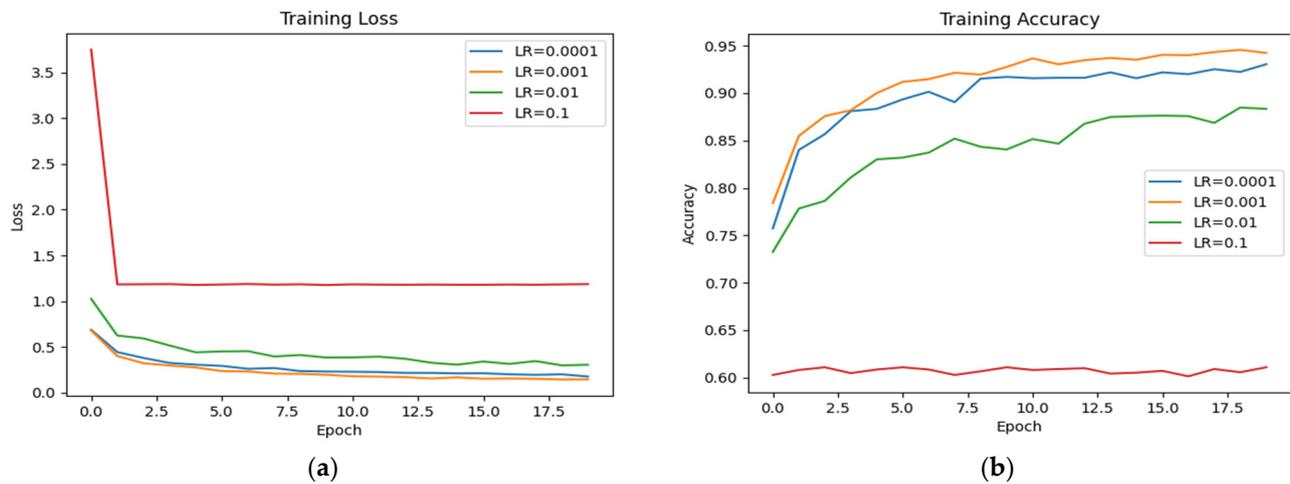
**Figure 11.** Confusion matrix.

**Table 5.** Evaluation results for different categories.

Ventilation Modes	Accuracy	Precision	Recall	F1 Score
Cooling ventilation	0.99	0.85	0.79	0.82
Dehumidification ventilation	0.97	0.96	0.87	0.91
Anti-condensation ventilation	0.97	0.86	0.61	0.71
Heat dissipation ventilation	0.88	0.92	0.68	0.79
Quality adjustment ventilation	0.94	0.67	0.75	0.70

### 3.2.2. Compare Training Results under Different Learning Rates

The following figure (Figure 12) shows the loss curves for the ResNet\_Attention model trained with different learning rates after being trained on the same dataset. These experiments were conducted to find the optimal learning rate for the model.



**Figure 12.** Model evaluation chart under different learning rates. (a) Training loss curve. (b) Training accuracy curve.

Based on the experiments, it was found that a learning rate of 0.01 resulted in a faster reduction of loss and quicker convergence of the model.

## 4. Discussion

In this study, we compared the proposed residual network model with self-attention mechanisms to several common sequence models, including LSTM, GRU, ResNet, and CNN. LSTM and GRU are widely used in sequence modeling tasks, while ResNet and CNN are popular deep learning architectures for image processing and classification. Compared to these models, our proposed model achieved higher accuracy in all ventilation categories, especially for cooling ventilation, where it reached 99%.

One of the advantages of the proposed model is its ability to capture the temporal and spatial dependencies in the ventilation data, which is important for accurately identifying different ventilation categories. Unlike the other models, our proposed model incorporates the self-attention mechanism, which allows it to focus on important features and enhance their representation. This mechanism also enables the model to learn more complex relationships between the inputs and outputs, which is crucial for achieving high accuracy in the ventilation classification task.

However, there are still some limitations that need to be addressed in future work. The model exhibits high computational complexity due to its complex network structure and the inclusion of self-attention mechanisms within the residual block structure, especially considering the input sequence length of 425 and the computational demands of self-attention. In some cases, this high computational complexity may be even higher. The current model training and evaluation times are excessively long. The average training time exceeds 23 min and 16 s, while the evaluation time is approximately 1 min and 16 s. Moreover, the dataset used in this study only covers a certain range of grain conditions and ventilation scenarios, which may not fully represent real-world situations.

Overall, our proposed model provides a promising approach to the development of more accurate and efficient ventilation control systems for grain storage by leveraging the principles of computer modeling and self-attention mechanisms. The model has demonstrated superior performance compared to several commonly used models, and future work can further improve its robustness and efficiency.

## 5. Conclusions

This paper discusses the current status of intelligent ventilation management in grain storage and its main challenges, which are due to a lack of clarity around the concept of intelligent ventilation and grain storage data. To address this, a multimodal concept for grain storage is proposed, which transforms the traditional ventilation problem into a pattern selection problem. This allows decision-makers to make informed decisions based on multiple factors rather than solely relying on ventilation regulations to determine the existence of a problem.

The study combines self-attention mechanisms with residual network models to solve decision-making problems in abnormal grain situations. The experimental results demonstrate that residual networks with self-attention mechanisms converge faster and have smaller losses, providing more accurate and efficient decision support for grain storage managers. Moreover, the use of multi-head attention mechanisms significantly improves feature extraction for sequence data, and adjusting these mechanisms for grain situation data in the future may further improve the accuracy of residual networks and shorten decision-making time.

Compared with traditional methods, this approach has significant advantages in dealing with decision-making problems in abnormal grain situations. By considering multiple factors and utilizing self-attention mechanisms, this method provides more accurate and efficient decision support for grain storage managers. In the future, this method can be extended to other fields, providing valuable insights and solutions to a wider range of decision-making problems.

**Author Contributions:** Conceptualization, H.L., T.Z. and Z.L.; methodology, H.L., Y.Z. and Z.L.; investigation, H.L.; writing—original draft preparation, H.L.; writing—review and editing, H.L.; supervision, T.Z. and H.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** This data comes from the grain depot in Yushu, Jilin, and are private data. It cannot be made public.

**Acknowledgments:** This experimental research project is supported by the National Key Research and Development Program, specifically the project titled “Research and Development of Quality Information Control Technology for Efficient Multimodal Grain Transportation Connection” (Project No. 2022YFD2100202).

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Lobell, D.B.; Cassman, K.G.; Field, C.B. Crop yield gaps: Their importance, magnitudes, and causes. *Annu. Rev. Environ. Resour.* **2009**, *34*, 179–204. [[CrossRef](#)]
2. Liu, F.J.; Zhong, L.X.; Liu, X.H.; Zhang, R.T. Research and implementation of improved grain drying and storage techniques for farmers in Northeast China. In Proceedings of the 5th Asia-Pacific Drying Conference, Hong Kong, 13–15 August 2007; World Scientific: Singapore, 2007; pp. 520–525.
3. Zhang, J.; Chen, Q. Research on intelligent grain depot system based on information technology. In Proceedings of the International Conference on Forthcoming Networks and Sustainability (FoNeS 2022), Nicosia, Cyprus, 3–5 October 2022.
4. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)] [[PubMed](#)]
5. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; Riedmiller, M. Playing Atari with deep reinforcement learning. *arXiv* **2013**, arXiv:1312.5602.
6. Young, T.; Hazarika, D.; Poria, S.; Cambria, E. Recent trends in deep learning based natural language processing. *IEEE Comput. Intell. Mag.* **2018**, *13*, 55–75. [[CrossRef](#)]
7. Schuster, M.; Paliwal, K.K. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681. [[CrossRef](#)]
8. Sitaula, C.; Shahi, T.B.; Aryal, S.; Marzbanrad, F. Fusion of multi-scale bag of deep visual words features of chest X-ray images to detect COVID-19 infection. *Sci. Rep.* **2021**, *11*, 23914. [[CrossRef](#)]

9. Shahi, T.B.; Xu, C.-Y.; Neupane, A.; Guo, W. Recent Advances in Crop Disease Detection Using UAV and Deep Learning Techniques. *Remote Sens.* **2023**, *15*, 2450. [CrossRef]
10. Shahi, T.B.; Shrestha, A.; Neupane, A.; Guo, W. Stock Price Forecasting with Deep Learning: A Comparative Study. *Mathematics* **2020**, *8*, 1441. [CrossRef]
11. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]
12. Deng, L.; Yu, D. Deep learning: Methods and applications. *Found. Trends Signal Process.* **2014**, *7*, 197–387. [CrossRef]
13. Chen, P.; Li, Q.; Zhang, D.-Z.; Yang, Y.-H.; Cai, Z.; Lu, Z.-Y. A survey of multimodal machine learning. *J. Eng. Sci.* **2020**, *42*, 557–569.
14. Ramachandram, D.; Taylor, G.W. Deep multimodal learning: A survey on recent advances and trends. *IEEE Signal Process. Mag.* **2017**, *34*, 96–108. [CrossRef]
15. Xie, Y.; Wang, Z.; Wang, Y.; Liu, C. Research progress in ultra low temperature grain storage technology. *J. Agric. Eng.* **2018**, *34*, 1–106.
16. Zhu, Z.; Sun, L. A Model for Aeration Management of Stored Grain Based on Bayesian Network. *IFAC Proc. Vol.* **1997**, *30*, 93–98.
17. Liu, X.; Li, B.; Shen, D.; Cao, J.; Mao, B. Analysis of Grain Storage Loss Based on Decision Tree Algorithm. *Procedia Comput. Sci.* **2017**, *122*, 130–137. [CrossRef]
18. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*. Available online: <https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf> (accessed on 24 June 2023).
19. Zhang, K.P.; Fu, Z.L. Design and application of distributed fiber optic temperature sensing technology. *Gen. Mech.* **2018**, *9*, 54–56+58.
20. Li, X.J.; Wu, Z.D. Determination method of equilibrium absolute humidity and dew point temperature for grain stack. *Grain Process.* **2011**, *36*, 34–37.
21. Xu, M.; Liu, Y.; Ma, Z.; Huang, L.; Wang, H.; Ma, Y.; Liu, Y.; Liu, Y. A hybrid convolutional neural network for rainfall-runoff modeling. *Water* **2019**, *11*, 1462.
22. Wang, F.; Jiang, M.; Qian, C.; Yang, S.; Li, C.; Zhang, H.; Wang, X.; Tang, X. Residual attention network for image classification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017.
23. Chung, J.; Gulcehre, C.; Cho, K.H.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv* **2014**, arXiv:1412.3555.
24. Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv* **2014**, arXiv:1406.1078.
25. Yu, Y.; Si, X.; Hu, C.; Zhang, J. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput.* **2019**, *31*, 1235–1270. [CrossRef] [PubMed]
26. Smagulova, K.; James, A.P. A survey on LSTM memristive neural network architectures and applications. *Eur. Phys. J. Spec. Top.* **2019**, *228*, 2313–2324. [CrossRef]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.