electronics

MDPI

Article

MIFM: Multimodal Information Fusion Model for Educational Exercises

Jianfeng Song *, Hui Chen, Chuan Li and Kun Xie

School of Computer Science and Technology, Xidian University, Xi'an 710071, China; xdhchen@foxmail.com (H.C.); lichuan20000214@163.com (C.L.); xiekun@xidian.edu.cn (K.X.)
* Correspondence: jfsong@mail.xidian.edu.cn

Abstract: Educational exercises are crucial factors in the successful implementation of online education as they play a key role in assessing students' learning and supporting teachers in instruction. These exercises encompass two primary types of data: text and images. However, existing methods for extracting exercise features only focus on the textual data, neglecting the rich semantic information present in the image data. Consequently, the exercise characterization vector generated by these methods struggles to fully characterize the exercise. To address these limitations, this paper proposes a multimodal information fusion-based exercise characterization model called MIFM. The MIFM model tackles the challenges of current exercise modeling methods by performing extraction and fusion operations on the heterogeneous features present in exercises. It employs a dual-stream architecture to separately extract features from images and text, and establishes connections between heterogeneous data using cross-modality attention methods. Finally, the heterogeneous features are fused using a Bi-LSTM combined with a multi-head attention mechanism. The resulting model produces a multimodal exercise characterization vector that fuses both modalities and incorporates three knowledge elements. In the experiments, by using the model to replace the exercise characterization modules in the three educational tasks, specifically, it achieves an increased ACC value of 72.35% in the knowledge mapping task, a heightened PCC value of 46.83% in the exercise difficulty prediction task, and an elevated AUC value of 62.57% in the student performance prediction task.

Keywords: multimodal fusion; educational exercise characterization; feature extraction

Citation: Song, J.; Chen, H.; Li, C.; Xie, K. MIFM: Multimodal Information Fusion Model for Educational Exercises. *Electronics* **2023**, *12*, 3909. https://doi.org/10.3390/electronics12183909

Academic Editor: Yoichi Hayashi

Received: 16 August 2023
Revised: 14 September 2023
Accepted: 14 September 2023
Published: 16 September 2023

1. Introduction

During the COVID-19 pandemic, schools worldwide faced the threat of the outbreak, compelling the majority of teachers and students to engage in remote education activities. Consequently, discussions and research related to online education proliferated during this period. For instance, Nirmalya Thakur [1] developed a dataset comprising tweets from Twitter during the pandemic, focusing on opinions, attitudes, and feedback regarding online learning. Research conducted at the University of Cluj Napoca, Romania, revealed that 78% of students perceived online education during the pandemic as beneficial to their learning, with 41.7% expressing satisfaction with the quality of online courses [2]. Objectively, the pandemic accelerated the growth of online education, but it also exposed certain challenges. Educational exercises are pivotal in online education, and exploring how to leverage them for more intelligent and precise assessment of student learning remains an area of investigation. Deep learning methods have shown promising results in intelligent assessment of student learning using educational exercises. However, prevalent deep learning techniques still exhibit shortcomings in the extraction of multimodal data features within the realm of online education.

Data feature extraction has garnered significant attention from researchers across various fields, including natural language processing and imaging. This is due to the direct impact of an effective feature extraction method on model performance. In today's era of

abundant data, the scenarios for training deep learning models have become diverse, and the number of datasets is experiencing exponential growth. While a large amount of data can enhance model performance to some extent, it also leads to a linear increase in model training time and greater demand for computational resources [3]. The current unimodal data format no longer meets the requirements of modern models, as real-life scenarios often involve multimodal data, such as videos with subtitles and audio. Input data may contain heterogeneous information, including text, images, and videos [4]. If unimodal architecture is still employed for feature extraction, one approach is to perform feature extraction operation only for the data in the corresponding format of the model, discarding the remaining modal data, which will cause a certain degree of information loss and lead to incomplete data information extraction. The textual content of the exercises fails to sufficiently extract knowledge points that can encapsulate the overarching semantic information of the entire set of exercises, resulting in a deficiency in the generated exercise representation vectors. Traditional unimodal exercise representation models overlook heterogeneous exercise information beyond text, and output exercise text representation vectors as representations of exercise semantic information, consequently missing out on knowledge points, example images, and other highly relevant information associated with the exercises. This oversight stems from the traditional approach's exclusive focus on textual content while neglecting the broader context of exercise elements. Another approach is to use the same data feature encoder to extract features from different modalities, concatenating the resulting data feature vectors to obtain the final characterization. Although these methods have the capability to access multimodal data, they encounter issues related to parameter disarray when employing unimodal encoders for feature extraction from multimodal data. Additionally, the simple concatenation operation used to merge heterogeneous data fails to effectively integrate such data, resulting in inadequately representing the semantic information of exercises in the generated exercise representation vectors. Consequently, this approach leads to the incomplete extraction of data from various modalities. To address these challenges, this paper proposes the Multimodal Information Fusion Model (MIFM) to generate multimodal vectors for corresponding exercises. MIFM adopts a dual-stream architecture for the feature extraction of multimodal data in the feature extraction module, and constructs corresponding data feature encoders for data of different modalities, respectively. The model outputs a heterogeneous data feature vector that fully captures the semantic information of the input multimodal data. The main contributions of this work are as follows:

1. A multimodal fusion-based exercise characterization method is proposed to vectorize exercise with heterogeneous data and use the generated multimodal vectors to act on downstream tasks.
2. Propose a dual-stream architecture for the feature extraction of heterogeneous data and fuse cross-modal attention for the fusion of heterogeneous features.
3. Through experiments conducted on datasets collected in real educational settings across three distinct educational tasks, the model is demonstrated to effectively enhance the educational task implementation.

2. Related Works

2.1. Unimodal Characterization Method

The unimodal characterization method for educational exercises primarily focuses on extracting features from the textual content within them. The content of exercises typically includes exercise text, example images, knowledge concepts, answers, and other relevant information. Among these components, exercise text is essential as it must be present in all exercises, and it is also the most explicit part of semantic information in the text [5]. Consequently, many researchers primarily rely on exercise text modeling to characterize exercises [6–8]. For instance, Shahmirzadi et al. [9] divide the topic text into sections, assign different weights to the resulting divisions, and combine each division vector to represent a complete topic. Zhang et al. [10] exclude images and formula information from the input

data, using a prefix tree-based lexicon-free word separation algorithm to extract features from the exercise text and reference answers. The extracted results are then employed as semantic feature vectors for the exercise. Huang et al. [11] utilize Convolutional Neural Networks (CNNs) to capture word-to-word interactions on a larger scale, enabling the learning of deep semantic features for characterizing exercise sentences as vectors. Hermann et al. [12] propose a two-layer structured Long Short-Term Memory (LSTM) model to obtain exercise characterization vectors by learning the context of each exercise. Many of these unimodal representation methods primarily rely on textual information present in the exercises, such as exercise text, knowledge points, and exercise explanations. However, they tend to disregard other modal, heterogeneous data embedded within the exercises, resulting in a partial loss of exercise information within the exercise representation vectors. In contrast, multimodal exercise representations integrate image information from the exercises after appropriate feature extraction in a coherent manner with textual features, serving as the representation vector for the exercises.

2.2. Multimodal Characterization Method

Most unimodal exercise characterization methods primarily utilize the textual information contained within exercises, such as exercise text, knowledge concepts, and exercise parsing. However, they ignore other modalities of heterogeneous data present in the exercises, resulting in the omission of certain exercise information within the exercise characterization vector. The key distinction between multimodal exercise characterization and unimodal exercise characterization lies in the fusion of image information with text features in the multimodal model. Multimodal exercise representations integrate image information from the exercises after appropriate feature extraction in a coherent manner with textual features, serving as the representation vector for the exercises. Liu et al. [13] employ one-hot encoding to encode topic knowledge concepts and utilized CNN to extract image features. The resulting knowledge concept vector and image features are compared with the text feature vector converted by word2vec. Finally, the exercises are inputted into the Attention-LSTM to generate multimodal exercise characterization vectors. This approach yields exercise representation vectors that encompass richer semantic information compared to unimodal exercise representation vectors and maximally capture the semantic characteristics embedded in the exercise resources [14–16]. As a result, it exhibits a significant improvement in performance on the "Finding Similar Exercises" task compared to the existing baseline at the time. However, this approach solely assesses the relationships between exercises based on semantic similarity, without considering other factors such as exercise difficulty and knowledge mapping.

2.3. Exercise Characterization Method

Exercise characterization methods can be broadly categorized into word-level and sentence-level approaches. Word-level methods involve converting each word in the exercise text into a corresponding feature vector and combining them to create a characterization vector that preserves the original semantic information effectively. Word-level exercise characterization encompasses three main methods: one-hot encoding, TF-IDF (Term Frequency-Inverse Document Frequency) based characterization [17], and word2vec-based characterization [18]. The one-hot encoding method assumes that each word in the text is mutually independent, disregarding interactions between words and neglecting contextual semantics and sequential information within the text. TF-IDF representation solely considers word frequency, overlooking positional information and interrelationships among words. Word2Vec, while considering contextual semantic associations of words, reduces the dimensionality of word embeddings, speeding up model training. However, it still fails to address several issues pertaining to synonymy and word sequence relationships in text representation.

Sentence-level characterization methods focus on extracting features from the word sequence and text structure information of exercise text using CNN and RNN (Recurrent

Neural Network) [19]. These methods consider the textual information of the exercise from a sentence-level perspective, resulting in the generation of an exercise characterization vector [20]. Sentence-level-based exercise characterization methods can be further divided into two categories: convolutional neural network-based methods and recurrent neural network-based methods. To address the challenges associated with RNN when handling lengthy text, researchers have introduced an alternative recurrent neural network architecture known as an LSTM network. Within the LSTM network, the hidden states maintain a record of all the information in the text sequence from the input to the current text position. Consequently, when a exercise text is fed into an LSTM model, it generates a corresponding hidden state. This allows for the direct utilization of the final hidden state, which encapsulates the global semantic information of the exercise, as the vector representation of the exercise.

3. Proposed Method

To tackle the challenges posed by incomplete extraction and inadequate fusion of heterogeneous features in data with different modalities using conventional feature extraction models, this paper proposes a multimodal model called MIFM (Multimodal Information Fusion Model). MIFM is designed to extract and fuse heterogeneous features in educational exercises. The model employs distinct feature extractors for different modalities of data, introducing a dual-stream architecture within the traditional multimodal vector representation framework. This allows multiple feature extractors to concurrently extract features from different modalities of data, encoding subject knowledge points, accompanying images, and exercise text heterogeneous data using separate encoders. This approach ensures the integrity of feature extraction from multimodal data while simultaneously enhancing the efficiency of feature extraction. Additionally, a multimodal cross-attention mechanism is employed to fuse the data from different modalities. The utilization of a multimodal cross-attention mechanism enables the precise capturing of critical information, facilitating efficient allocation of information processing resources. This mechanism involves the weighted distribution of attention to specific local information within the input data, granting higher attention scores to locally significant information while selectively disregarding lower-weighted local details. This approach enhances the model's scalability and robustness, promoting adaptability to varying input conditions.

3.1. Overview of the MIFM

The multimodal information fusion model MIFM, illustrated in Figure 1 model architecture, comprises three main layers: heterogeneous data feature extraction, cross-modal attention, and modal fusion. The model takes as input heterogeneous data consisting of two modalities: text and image. The text modality includes exercise text and knowledge concept entities derived from the exercise text. The image modality mainly contains one or more images that complement the textual information by conveying image-related semantics. The model outputs a multimodal semantic characterization vector that combines both text and image data. The data feature extraction layer performs three types of element extraction: text input, knowledge embedding, and image input. In this layer, text and image encoders are used to extract text embedding, text entity embedding, and image embedding from the input data. Subsequently, the text vector and knowledge concept entity vector are concatenated with the image vector and passed to the cross-modal attention layer. In the cross-modal attention layer, attention operations are applied to the image vector using the text vector and to the text vector using the image vector. These operations establish internal connections between different modal data. Finally, the data from the cross-modal attention layer is fed into the modal fusion layer, where the heterogeneous data are fused, resulting in multimodal characterization vectors that capture the global semantic information of the multimodal data for the grouping task. The pseudocode for MIFM is shown in Algorithm 1.

Algorithm 1 Pseudocode for MIFM

Input: Exercise $(text_i, image_i, knowledge_i)$
Output:Exercise Characterization Vector

procedure MIFM($Exercise_i$)
  $text_{emb_i} = Transformer(text_i)$ ▷ text embedding
  $image_{emb_i} = DenseNet(image_i)$ ▷ image embedding
  $knowledge_{emb_i} = Glove(knowledge_i)$ ▷ kc embedding
  $text_{emb_i} = text_{emb_i} \oplus knowledge_{emb_i}$
  $Matrix_{text} = shift(text_{emb_i})$
  $Matrix_{image} = shift(image_{emb_i})$
  $n = len(Matrix_{text})$ ▷ multimodal cross attention
  for each $i \in [0, n-1]$ **do**
    $q = selfAttention(Matrix_{text_i}, Matrix_{image_i})$
    $h = selfAttention(Matrix_{image_i}, Matrix_{text_i})$
  end for ▷ modal fusion
  $v^f = MultiHeadAttention(BiLSTM(q, h))$
  return v^f
end procedure

The algorithm employs a transformer to extract text features, utilizes DenseNet for image feature extraction, and encodes knowledge concepts using GloVe embeddings. The temporal and spatial complexity of these three processes depends on the temporal and spatial complexity of the chosen network models. Let m denote the number of samples in the dataset, and θ represent the length of the feature vectors. Subsequently, the fusion of text and knowledge concept vectors, along with the vector shift operation, can both be executed in $O(m)$ time complexity. The resultant text and image matrices, denoted as n, would require $O(mn\theta)$ space. The operation of feature fusion for each vector in the text and image matrices, employing self-attention mechanisms, incurs a time complexity of $O(mn)$. Finally, feeding the fused vectors into a dual-stream structure with multi-head operations results in the creation of the ultimate feature fusion vector v^f, with a time complexity of $O(m)$. The resulting v^f vector would occupy the $O(m\theta)$ space. Consequently, the overall time complexity of the algorithm is $O(mn)$ plus the time spent on the feature extraction models, while the space complexity is $O(mn\theta)$ in addition to the space occupied by the feature extraction models.

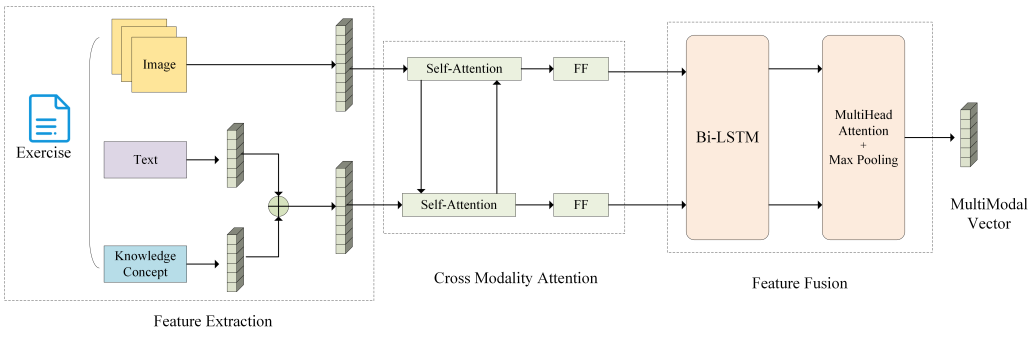

Figure 1. MIFM model architecture.

3.2. Text-Encoding Layer

The main purpose of the exercise text-encoding layer is to use the encoder structure in the transformer model [21] for feature extraction of the textual content in the exercise text data input to the model. The transformer is a deep neural network that leverages a multi-head attention mechanism, enabling the parallel processing of input data. The model is built upon an encoder-decoder structure, where feature extraction is accomplished by stacking encoders and decoders. Within the model, a self-attentive mechanism and positional

embedding method are employed to effectively extract semantic information from the input text. Compared to traditional models such as LSTM and so on, the transformer architecture relies solely on attention mechanisms to model dependencies between input and output, facilitating parallelization and yielding more interpretable models. Each attention head within the transformer can perform distinct tasks, aligning well with our objective of feature extraction from multimodal data.

Originally designed for text translation tasks in Natural Language Processing (NLP) [22], the transformer model utilizes its built-in encoder to encode the input text, which is then fed to the decoder for generating corresponding translation results. In this paper, our objective is to vectorize the text; thus, this paper only utilizes a portion of the encoder function within the transformer model. The encoder comprises two primary components: a multi-head attention mechanism and a fully connected feedforward network. The output of each component is passed through the corresponding residual network structure for further processing.

The model takes word embeddings as inputs, enriched with positional embedding information to express word-to-word distances. This is demonstrated in Equations (1) and (2), which outline the specific positional encoding.

$$p_{2i} = \sin\left(p/10000^{2i/d_{\text{pos}}}\right) \tag{1}$$

$$p_{2i+1} = \cos\left(p/10000^{2i/d_{\text{pos}}}\right) \tag{2}$$

where d_{pos} represents the length of the word embedding, and the positional encoding vector is also of length d_{pos}. The position vector value p_{2i} corresponds to the even-dimensional index i, calculated using the sine function. Similarly, the position vector value p_{2i+1} corresponds to the odd-dimensional index i, computed using the cosine function. By dividing by $10000^{2i/d_{\text{pos}}}$ within the trigonometric function, the correlation between words with greater relative distance in the input text becomes weaker. The p in the formula represents the position of the word in the input text, where p_i denotes the value of the number ith element in the pth word position vector. Finally, the word embedding is directly added to its corresponding position vector, serving as the input to the encoder.

The model utilizes a multi-head attention mechanism, which shares similarities with the conventional method. However, the key distinction lies in the fact that the ordinary attention mechanism produces a single attention value following the attention operation, while the multi-head attention mechanism comprises h Scale Dot-Product Attention submodules that are stacked together. As a result, h attention values are generated after the attention operation. These h vectors are then concatenated and subjected to a linear projection to obtain the final attention value. The multi-head attention mechanism is implemented based on the self-attention mechanism. Within each self-attentive module, independent linear mapping matrices W_i^V, W_i^K, and W_i^Q are maintained, since the weights are not shared among the modules. The self-attention inputs Q, K, and V are obtained by multiplying the input matrix X (word embedding of input text and positional embedding) with the three mapping matrices. In the encoder of the transformer model, each attention module is followed by a fully connected feedforward network layer and a residual connection. This allows for the addition of input and output data at each position, enabling the training of a deeper and more efficient network.

3.3. Image-Encoding Layer

The purpose of the exercise image-encoding layer is mainly to extract features from the exercise accompanying images contained in the input exercise and convert each image data into the input data into a vector characterization. Similar to text processing, this layer acts as an image encoder, converting all images in the input data into the corresponding characterization vectors. While traditional neural networks often deepen layers or widen the network structure to enhance data feature extraction, this approach faces challenges

such as the vanishing gradient problem during backpropagation and the difficulty in training models with increased parameters. To avoid the above problems and improve the effect of the model for feature extraction, this paper uses DenseNet [23] to extract the image features from multimodal data and convert the attached images contained in the exercise into corresponding image vectors. DenseNet is a convolutional neural network with dense connections, which mitigates the gradient disappearance while enhancing the transfer of features from layer to layer. The model structure is illustrated in Figure 2. The network takes the feature maps of all previous layers as input for each layer while ensuring maximum information transfer between layers. Additionally, the feature maps of the current layer are used as input for all subsequent layers. The feature learned by the network in the current layer will be used as input data in the subsequent layers, which makes the number of feature maps output from each convolutional layer in the network small, the feature transfer between layers is more efficient, and the network is easier to train, thus enabling the extraction of more diverse image features and a significant improvement in network performance.

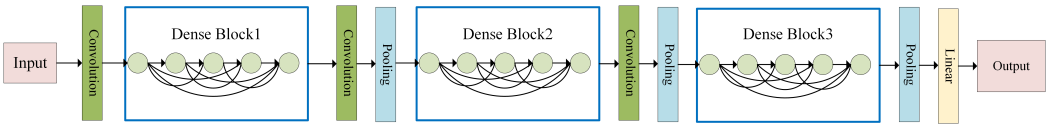

Figure 2. DenseNet model architecture.

In order to improve the ability of extract features from the images contained in the multimodal data of the exercises, DenseNet utilizes an encoder-decoder approach [24] to pre-training the network using specific image data before its formal training operation. Firstly, DenseNet serves as the image encoder, and then the image feature vector generated by the encoder is reduced to the corresponding image using the deconvolutional neural network [25] as the image decoder (DenseDec) for the purpose of pre-training. The number of deconvolutional layers in the decoder matches the number of image encoder layers. Throughout the pre-training process, the network parameters are adjusted and optimized by minimizing the loss function L_{dense}:

$$L_{\text{dense}} = \sum_{p}(\text{DenseDec}\ (\text{DenseNet}(I)) - I)^2 \tag{3}$$

The loss function L_{dense} in Equation (3) has I as the exercise attachment in the pre-training dataset. When extracting features from image data, images are converted into a fixed-length vector denoted by $v_i = \sigma(\text{DenseNet}(I_i))$. The vector captures the maximum characterization of image features after passing through the image feature extractor DenseNet. Here, $\sigma = \frac{1}{1+e^{-z}}$, I_i denotes the ith image in exercise. v_i represents the characterization vector corresponding to the ith image. If the exercise has one or more images, after feature extraction using the image feature extractor, a vector matrix will be output as the image features, and the vector matrix of images is represented as $\text{V} = (v_1, v_2 \ldots v_n)$, where n is the total number of images contained in exercise Q.

3.4. Knowledge Concept Embedding Layer

The purpose of the knowledge concept-embedding layer is mainly to perform vectorization operations on the exercise text input to the model to obtain a low-dimensional dense vectorized characterization of the subject knowledge concept as one of the constituent elements of the multimodal exercise characterization vector. Specifically, words are input into the model and the corresponding word embeddings are output. Since knowledge concepts consist of multiple individual words, there is no strong correlation between the words and the contextual semantic requirements are not high. Therefore, simple word-embedding models are generally used for implementation when doing word vectorization operations. The commonly used method involves obtaining word embeddings through one-hot encoding for each word. However, using one-hot encoding leads to excessively

sparse word embeddings, which hampers the model convergence during training and impacts the final model performance [26]. On the other hand, the word2vec model generates word embeddings in a static manner, providing general applicability, but it lacks the ability to dynamically optimize for specific tasks.

This paper employs the Glove word-embedding model [27] to implement knowledge concept vectorization operations in the knowledge concept-embedding layer. Glove is a global logarithmic bilinear model that utilizes unsupervised learning methods to train word embeddings. The model combines the advantages of both global matrix decomposition and local context windows. Unlike other models that train the entire sparse matrix or a single context window in a large corpus, Glove directly trains on the word–word co-occurrence matrix. This approach effectively utilizes statistical information to generate a semantically rich vector space. The most crucial module in the Glove model is the training of the co-occurrence matrix. There are two ways to train co-occurrence matrices in the model: symmetric window training, which does not consider word order, and asymmetric window training, which considers word contextual order. To obtain high-quality word embeddings, the model trains the co-occurrence matrix using the asymmetric window method. The specific steps are as follows:

1. Generate a word list by counting the number of occurrences of each word in the exercise text corpus. Sort the words in the word list based on their occurrence frequency, from highest to lowest. Let c_i denote the ith word, f_i denotes the number of occurrences of the ith word, and n denotes the size of the word list, which refers to the number of different words in the exercise text corpus.
2. Set the sliding window size to w and traverse all words in the corpus. Record the frequency of word occurrences in the fixed window on the left side of the target word. Generate a left co-occurrence matrix X^L, with X^L_{ij} denoting the words in the ith row and jth column of the left co-occurrence matrix.
3. Use V^A to represent the low-dimensional word-embedding characterization based on the left co-occurrence matrix training. Train the model through the loss function J^A [27], which is calculated as shown in Equation (4).

$$J^A = \sum_{i,j=1}^{n} f\left(X^L_{ij}\right)\left(\left(v^A_i\right)^T v^A_j + b^A_i + b^A_j - \log X^L_{ij}\right)^2 \tag{4}$$

As described in Equation (4), n is the size of the lexicon (with the co-occurrence matrix having dimensions of $n \times n$). The vectors v^A_i and v^A_j are asymmetric low-dimensional word characterization vectors of words c_i and words c_j, respectively. The bias terms corresponding to v^A_i and v^A_j are denoted as b^A_i , b^A_j, respectively. Additionally, the function $f\left(X^L_{ij}\right)$ is the weighting function.

3.5. Modal Fusion Layer

The main purpose of the modality fusion layer is to fuse features from the text vector of the exercise text-encoding layer, the image vector of the exercise image-encoding layer, and the knowledge-embedding vector of the knowledge-embedding layer. This fusion is achieved using a cross-modal attention method based on the attention mechanism, which establishes connections between data from different modalities through self-attention. The resulting text feature vector and image feature vector, obtained after the cross-modal attention operation, are then input into the multi-head attention module for fusing heterogeneous data features. The final output is a multimodal exercise characterization vector that fuses two modalities and three features, effectively representing heterogeneous data features for downstream tasks. The incorporation of attention mechanisms into respective domain-specific base models has consistently demonstrated significant improvements in performance across various tasks. This serves as compelling evidence of the efficacy of attention mechanisms in the realm of feature extraction. The approach employed in the modal fusion layer effectively combines the advantages of cross-attention and multi-head

attention mechanisms, ensuring comprehensive feature extraction for each modality when dealing with cross-modal data.

The text characterization vector $sentence_{emb}$ is obtained from the text-encoding layer. Similarly, the image feature vector, represented as $image_{emb}$, is obtained from the image-encoding layer, and the knowledge-embedding vector, denotes as kc_{emb}, is obtained from the knowledge-embedding layer. Initially, the text characterization vector and the knowledge-embedding vector, both of the same modality (text), are combined through summation to generate $text_{emb}$ as a unified characterization of the input text data. The matrix characterization, resulting from translating both the text vector $text_{emb}$ and the image vector $image_{emb}$, is then input into the cross-modal attention module to facilitate the exchange of heterogeneous information between the two modalities. The cross-modal attention module, illustrated in Figure 3, processes each element of the embedded vector obtained from the encoding layer and translates it into a matrix characterization specific to its corresponding modality. Subsequently, attention operations are performed between the vectors and matrices of different modalities, establishing connections between the data from each modality.

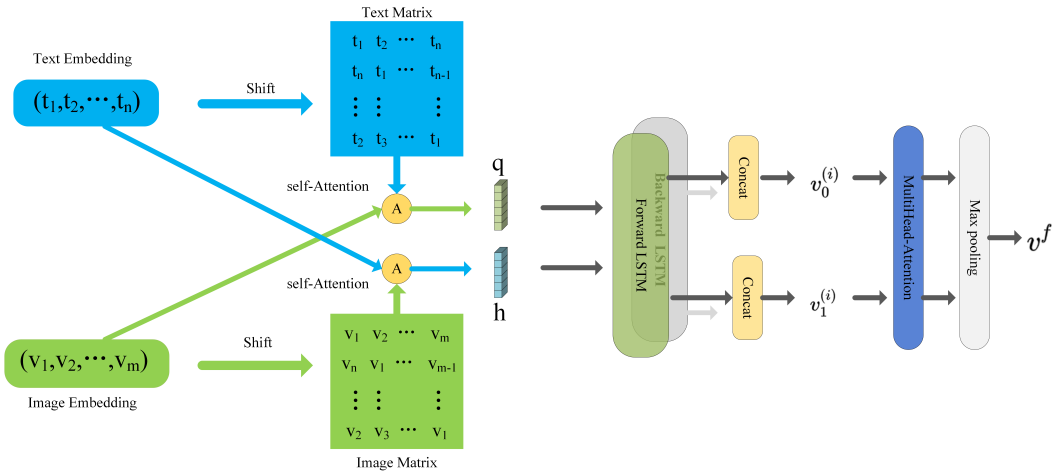

Figure 3. Modal Fusion Procedure.

This is achieved by performing attention operations between two modalities: text-based attention operations on images and image-based attention operations on text. This process results in a text characterization vector h that incorporates image features and an image characterization vector q that incorporates text features. The calculation formulas are shown in Equations (5) and (6):

$$h_i^k = \text{CrossAttention}\left(t_i^{k-1}, \left\{v_1^{k-1}, \ldots, v_m^{k-1}\right\}\right) \tag{5}$$

$$q_j^k = \text{CrossAttention}\left(v_j^{k-1}, \left\{t_1^{k-1}, \ldots, t_n^{k-1}\right\}\right) \tag{6}$$

where t_i^k denotes the text vector, v_j^k denotes the image vector, and CrossAttention denotes the attention operation.

As depicted in Figure 3, which contains fused heterogeneous features, there are inputs into the bidirectional Long Short-Term Memory (Bi-LSTM) [28] network to further establish connections between the heterogeneous features. The forward hidden state volume $\overrightarrow{h_t}$ at time step t in the Bi-LSTM network is computed based on the previous forward hidden state volume $\overrightarrow{h_{t-1}}$ and the current input vector w_t. Similarly, the reverse hidden state volume $\overleftarrow{h_t}$ at time step t is calculated using the next moment's hidden state volume $\overleftarrow{h_{t+1}}$ and the current input vector w_t. Consequently, the hidden state characterization of each input vector can be calculated by concatenating the hidden states in both directions: $v_w = concatenate(\overrightarrow{h_w}, \overleftarrow{h_w})$. After the operation of Bi-LSTM, the vector matrix v containing heterogeneous features is

produced. To encode the position of each vector v_i in the matrix, one-hot encoding is applied to obtain the position-encoding vector. Then, the vector position v_i and its corresponding position of the encoding vector are concatenated to form the feature vector v_c^i that fuses position information. Each feature vector v_c^i serves as input for the multihead attention mechanism, and the maximum value among all the results from the multihead attention mechanism is selected as the final multimodal characterization vector. The utilization of a multi-head attention mechanism enables the model to focus on information from different modalities at various positions, thereby enriching the captured feature information. This facilitates the model in harnessing the full potential of diverse modal data to enhance its performance. The computation of the multimodal characterization vector is described by Equation (7):

$$v^f = \max\left\{\text{MultiHead}\left(\text{LayerNorm}\left(v_c^i, v_c^i, v_c^i\right) + v_c^i\right)\right\} \tag{7}$$

where v_c^i represents the feature vector fused with location information, and LayerNorm refers to the multilayer normalization technique, while MultiHead denotes the multi-head attention mechanism [21,29]. The three vectors, denoted as v_c^i, undergoing LayerNorm processing and adding v_c^i, are subsequently employed as input parameters Q, K, and V for the MultiHead mechanism.

$$MultiHead(Q, K, V) = Concat(head_1, ..., head_h)W^O \tag{8}$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \tag{9}$$

The MultiHead operation in Equation (7) is depicted as shown in Equations (8) and (9). Here, V represents the input feature vector, and Q and K are feature vectors used to compute the attention weights. In this paper, the parameters Q, K, and V in Equations (8) and (9) are all set to the value v_c^i. Within the $Attention(QW_i^Q, KW_i^K, VW_i^V)$ expression, each feature vector is multiplied by its corresponding mapping matrix to produce the $head_i$ result after the attention calculation. Subsequently, the MultiHead operation concatenates multiple $head_i$ results and multiplies them by the mapping matrix W^O. The dimension of W^O is the same as that of the h concatenated $head_i$ results. Lastly, v^f corresponds to the multimodal characterization vector of the exercise, which is formed by fusing the knowledge-embedding vector, exercise text vector, and exercise-accompanying feature vector from the input exercise. This fusion process results in the final global characterization vector of the exercise.

4. Experiment

This section presents experiments conducted on three tasks in the education domain: knowledge mapping [30], exercise difficulty prediction [11], and student performance prediction [31]. The knowledge mapping task is a multi-class classification task, while the exercise difficulty prediction is a regression task, representing distinct tasks within the domain of educational exercise modeling. The knowledge mapping task accurately delineates the knowledge points encompassed by each exercise, allowing for the tracking of a student's engagement with specific knowledge points and identification of their weak areas. Exercise difficulty prediction, on the other hand, enables a more rational assembly of exercise sets of varying difficulties based on individual student learning profiles, with the goal of enhancing overall learning outcomes. Finally, the student performance prediction task serves as a valuable tool for analyzing the student learning situation, facilitating personalized educational services through tailored exercise sets. These three tasks stand as quintessential and representative challenges in the field of educational exercise modeling, and their effective utilization represents a pivotal step in advancing the field of intelligent education. The exercise characterization modules in these tasks are replaced with the comparison models selected in the experiments. The primary goal is to compare the effectiveness of these models in extracting and fusing heterogeneous features across the three tasks. Additionally, this experiment visualizes the vectors generated by the

multimodal characterization model using T-SNE [32] to analyze their ability to capture the semantic information of the input exercise. This analysis serves to verify the effectiveness and advancement of the proposed MIFM model in the task of multimodal exercise characterization.

4.1. DataSet

The experimental dataset consisted of a high school chemistry exercise obtained from the educational website https://www.jyeoo.com using crawling techniques. The code of the web crawler process was implemented using various libraries and tools, including Requests, BeautifulSoup, Scrapy, and Selenium, among others. Throughout the data collection process, the web crawler strictly adheres to the website's crawling protocol, limiting data crawling within the boundaries defined by the website's terms of use. Additionally, it is important to note that the obtained data are exclusively utilized to investigate the model proposed in this paper, without engaging in any unauthorized or illicit activities. This approach ensures both the integrity of the data collection process and compliance with legal and ethical standards. The experimental dataset mainly includes three types of exercises: multiple choice, judgment, and quiz exercises. The collected information includes exercise text, exercises with diagrams, exercise scores, exercise difficulty, exercise types, knowledge concepts, and other relevant information. Information pertaining to exercise difficulty ranges from 1 to 10, categorizing the difficulty level of each exercise into ten distinct grades. A higher numerical value indicates a greater level of difficulty for the respective exercise. The final format for the difficulty data of the exercises is as follows: (Exercise Number, Exercise Text, Knowledge Concept Number, Exercise Attached Diagram Path, Difficulty Level (1–10)). During the crawling process, the issue of duplicate the data retrieval was encountered. To address this, the experiment utilizes the Glove word-embedding model [27] mentioned earlier to obtain word embeddings for the crawled exercises. Subsequently, a simple cosine calculation is applied to these word embeddings to determine the similarity between the exercises. Exercises with a similarity score exceeding 0.5 are excluded from the database insertion process. Additionally, for exercises containing formulas, a third-party tool, MML2OMML.xsl, is employed to convert the formulas into MathML format. This conversion facilitates further processing of the formula within the exercises. Ultimately, the dataset of exercises, obtained through a process of duplicate data removal, formula conversion, and punctuation elimination, comprises a total of 186,525 exercises. The dataset comprises two types of exercises: text-only exercises and exercises containing both text and image data. The distribution of exercises with different difficulty levels and exercises containing different knowledge points is shown in Figure 4. Figure 5 illustrates the distribution of exercises with varying lengths after processing.

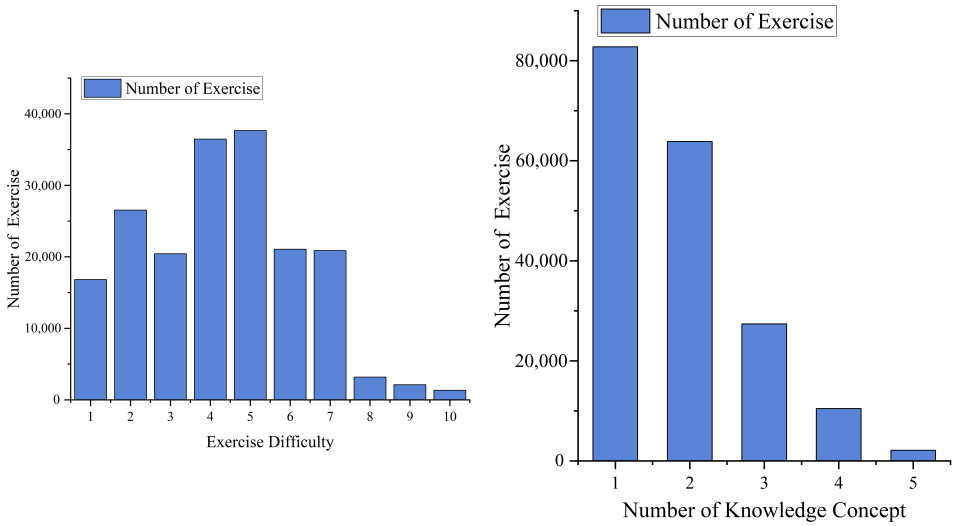

Figure 4. Difficulty of Exercises and Knowledge Distribution.

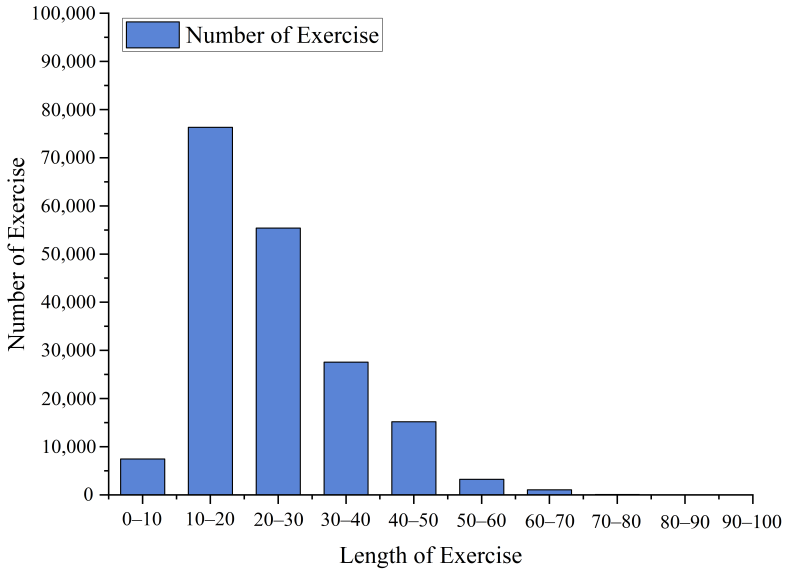

Figure 5. Length of Exercises Distribution.

4.2. Experimental Evaluation Metrics

In the experimental process, various evaluation metrics are employed to assess the model performance across different tasks. For the knowledge mapping task, the model performance is evaluated using ACC, Precision, Recall, and F1 values. In the exercise difficulty prediction task, MAE, RMSE, and PCC are utilized to evaluate the model's predictive accuracy. In the student performance prediction task, MAE, RMSE, ACC, and AUC values are employed to evaluate the model's performance. Among them, ACC, Precision, Recall and F1 evaluation metrics are relatively common and easy to calculate, so the calculation of other evaluation metrics and related concepts will be introduced.

MAE (Mean Absolute Error) measures the average absolute difference between the true value and the predicted value, and it is calculated using Equation (10):

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{10}$$

where n is the number of samples, y_i is the true value, and $\widehat{y_i}$ is the predicted value.

RMSE (Root Mean Square Error) calculates the square root of the mean of the squared differences between the true value and the predicted value. It is computed using Equation (11):

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \tag{11}$$

where n is the number of samples, y_i is the true value, and $\widehat{y_i}$ is the predicted value.

AUC (Area Under the ROC Curve) measures the performance of a binary classifier, and it is determined by calculating the area under the receiver operating characteristic curve. Equation (12) demonstrates its calculation:

$$AUC = \frac{\sum_{ins_i \in positive} rank_{ins_i} - \frac{M\times(M+1)}{2}}{M \times N} \tag{12}$$

where $rank_{ins_i}$ denotes the ordinal number of the ith sample, M and N are the number of positive and negative samples, respectively, and $\sum_{\text{ins}_i \in \text{ positive}} rank_{\text{ins}_i}$ represents the summation of the ordinal numbers of the positive samples.

PCC (Pearson Correlation Coefficient) quantifies the linear relationship between two variables X and Y. It is computed using Equation (13):

$$\mathrm{PCC} = \frac{1}{n-1}\sum_{i=1}^{n}\left(\frac{x_i-\bar{x}}{s_x}\right)\left(\frac{y_i-\bar{y}}{s_y}\right) \tag{13}$$

where $\bar{x}$ and s_x denote the mean and standard deviation of variable X, respectively, x_i denotes the sample, and y_i denotes the prediction label.

4.3. Experimental Environment and Parameters

This experiment was conducted using Pytorch, and the experimental hardware and software environment configuration is provided in Table 1.

Table 1. Experimental environment.

Project	Enviroment
Memory	32 GB
GPU	NVIDIA GeForce RTX3060
Python Version	Python3.9.1
Pytorch Version	Pytorch1.13.0

Several key parameters in the model are set as follows: the embedding dimension for both the exercise text characterization vector and the exercise accompanying image characterization vector output in the exercise feature extraction module is set to 256, the number of LSTM layers is set to 2, and the learning rate for model training is set to 0.001 using the Adam optimizer. The maximum length of the input exercise text is set to 128, the image dimension is $64 \times 64 \times 3$, the drop_out is set to 0.5, and the batch_size is set to 128.

4.4. Experiment and Analysis

In order to assess the effectiveness of the model MIFM, comparative experiments were conducted on three educational tasks. The first task, knowledge concept mapping, is a multi-category task where the input data are in the format of (e, kc), where 'e' represents the exercise, including its text of the exercise and accompanying diagram, and 'kc' denotes the knowledge concept associated with the exercise. In this task, the comparison model replaces the data characterization module of the classification model. The exercise vector is obtained through the exercise characterization model, and then the classification task assigns the exercise to its corresponding knowledge concept. Evaluation metrics such as ACC, Precision, Recall, and F1 are primarily used to assess the performance of the model in the knowledge concept-mapping task.

The second task, exercise difficulty prediction, involves a regression task aimed at estimating the difficulty level of an exercise on a scale of 1 to 10. The input data for this task follows the format (e, diff), where 'e' represents the exercise and 'diff' represents the exercise difficulty, with $0 < \mathrm{diff} <= 10$. The comparison model replaces the data encoding module in the exercise difficulty prediction model to compare prediction results. In this task, model performance is evaluated using metrics such as MAE, RMSE, and PCC.

The last task is student performance prediction, which aims to predict student performance on each exercise based on student answer records. Similar to the previous tasks, the selected comparison model replaces the exercise characterization module in the task model. Model performance is then evaluated using evaluation metrics such as MAE, RMSE, ACC, and AUC.

In this experiment, the HAN model is utilized for exercise knowledge concept mapping, the TACNN model for exercise difficulty prediction, and the EERNN for student performance prediction. These models serve as the baseline models for each respective task. To evaluate the performance of the exercise characterization models, the exercise

characterization module in the aforementioned task models is replaced, and the model's performance on the three tasks is compared. The selected exercise characterization models include the following:

ELMo: This model is a pre-trained language model based on LSTM for feature extraction, generating dynamic word embeddings [33]. It employs a Bi-LSTM network to extract features from input text. When the input data include other modalities besides text, only the text data are considered for feature extraction, while other modal data are ignored.

BERT: This model is a text-based pre-trained model [34], implemented internally using a stack of transformer models. It generates text characterization vectors that capture rich contextual semantic information through self-supervised learning. Similar to ELMo, it accepts only textual data input and disregards data from other modalities.

m-CNN: This model is an enhanced multimodal characterization model based on CNN that can handle heterogeneous data input [35]. It employs a multimodal convolutional approach to fuse exercise text and accompanying exercise images, generating multimodal characterization vectors for the exercise.

MIFM: The model proposed in this paper is a multimodal information fusion model for exercise characterization. It accepts data input from both text and image modalities, using a dual-stream architecture with different modal encoders. Features are extracted from the heterogeneous input data, followed by feature fusion using a cross-modal attention mechanism. The model outputs a multimodal characterization vector that combines text and image features, resulting in a comprehensive exercise characterization fusing both heterogeneous features.

For the three aforementioned tasks and the selected comparison models, if it includes the same model/network as MIFM proposed in this paper, the relevant parameters of the same modules are tuned to be the same in order to ensure the rigor of the experimental results. When the input exercise comprises only text data, the vectorization process is performed solely on the text data. Conversely, if the exercise contains both text and image data, feature extraction and fusion are carried out to create the exercise's characterization vector. Some of the comparison models chosen for the experiment only support unimodal data. In such cases, feature extraction is conducted solely on the required input data type. However, if the comparison model is multimodal, data from both modalities are obtained concurrently. Subsequently, experiments are conducted for each educational task to assess the performance of the selected comparison model.

Table 2 presents the experimental results of each model on the knowledge mapping task. Upon examining the performance of each model across different metrics, it is evident that the original model, ELMo, and BERT, being unimodal feature extraction models, can only extract features from the text data input and do not accommodate data from other modalities. Consequently, the exercise characterization vectors generated by these models struggle to capture the complete semantic information of the exercises, resulting in subpar overall performance. In contrast, the m-CNN model possesses multimodal feature extraction and fusion capabilities. It accepts input data from both text and image modalities, extracting text and image features and fusing them using multimodal convolution to create exercise characterization vectors with fused heterogeneous features. However, m-CNN employs a unimodal feature extractor to extract multimodal features, which can lead to parameter confusion and information loss during both feature extraction and fusion processes. As a result, the performance improvement achieved by this model is limited. On the other hand, the MIFM model proposed in this paper adopts a dual-stream architecture and distinct modality-specific feature extractors. It separately extracts features from data of different modalities and employs cross-modal fusion to integrate heterogeneous features. This approach yields significant improvements across all four performance metrics. Tables 3 and 4 display the performance of each model on the exercise difficulty prediction task and the student–answer performance prediction task, respectively. A comparison of the metrics for these two tasks reveals that models with multimodal data extraction and fusion capabilities outperform the unimodal models across various

evaluation metrics. The effectiveness of the feature extraction and fusion method directly influences the semantic richness of the exercise characterization vectors generated by the multimodal models, thereby impacting the performance of downstream tasks.

Table 2. Knowledge Mapping. Bold data is the best result under the same criterion.

	ACC	*Precision*	*Recall*	*F1*
Original [30]	0.5281	0.3973	0.7521	0.5219
ELMo [33]	0.6435	0.7508	0.7274	0.7389
BERT [34]	0.5983	0.7102	0.6376	0.6719
m-CNN [35]	0.6521	0.7420	0.6697	0.7039
MIFM	**0.7235**	**0.8164**	**0.7583**	**0.8065**

Table 3. Exercise Difficulty Estimation. Bold data is the best result under the same criterion.

	MAE	*RMSE*	*PCC*
Original [11]	0.2308	0.2801	0.3231
ELMo [33]	0.2378	0.2776	0.4421
BERT [34]	0.2303	0.3105	0.3753
m-CNN [35]	0.2108	0.2721	0.3809
MIFM	**0.2076**	**0.2632**	**0.4683**

Table 4. Student Performance Prediction. Bold data is the best result under the same criterion.

	MAE	*RMSE*	*ACC*	*AUC*
Original [31]	0.4362	0.4653	0.7417	0.5279
ELMo [33]	0.3635	0.4672	0.7731	0.5535
BERT [34]	0.4239	0.4534	0.7263	0.5107
m-CNN [35]	0.4152	**0.4398**	0.7535	0.5631
MIFM	**0.3512**	0.4521	**0.7736**	**0.6257**

As shown in Figure 6, the visualization experiment selects 300 exercises encompassing five different knowledge concepts. Through the exercise multimodal characterization model, which fuses heterogeneous exercise data (exercise knowledge concepts, exercise text, and exercise example images), the corresponding characterization vectors are obtained. These vectors are then visualized using T-SNE, a technique that enhances the randomized nearest neighbor embedding algorithm to visualize high-dimensional data. It accomplishes this by converting data similarity into joint probabilities and minimizing the dispersion between joint probabilities of different dimensional embedding data. Consequently, each data point is assigned a position in two- or three-dimensional space. The visualization results reveal that the multimodal characterization vectors of exercises sharing the same knowledge concept are closely grouped together, while exercises with different knowledge concepts are scattered across the visualization graph. Some exercises with similar content intersect on the graph. These visualization outcomes demonstrate that the multimodal characterization model MIFM effectively extracts corresponding features from the input heterogeneous data and produces multimodal exercise characterization vectors that highly preserve the semantic information of the original data.

The performance of the MIFM model and the selected comparison models across three distinct educational tasks, as well as the T-SNE vector visualization results, affirm the effectiveness of the proposed multimodal information fusion-based exercise characterization model, MIFM. The model excels in heterogeneous feature extraction and information fusion operations, highlighting the significance of fusing heterogeneous data (exercise knowledge concepts and exercise accompanying images) for expressive exercise vectors. The experimental findings indicate that image data in multimodal datasets also contains rich semantic information. By extracting image features and fusing them with text feature characterization, a global characterization vector representing multimodal data is

obtained. The utilization of multimodal characterization vectors with fused heterogeneous information significantly enhances the performance of downstream tasks.

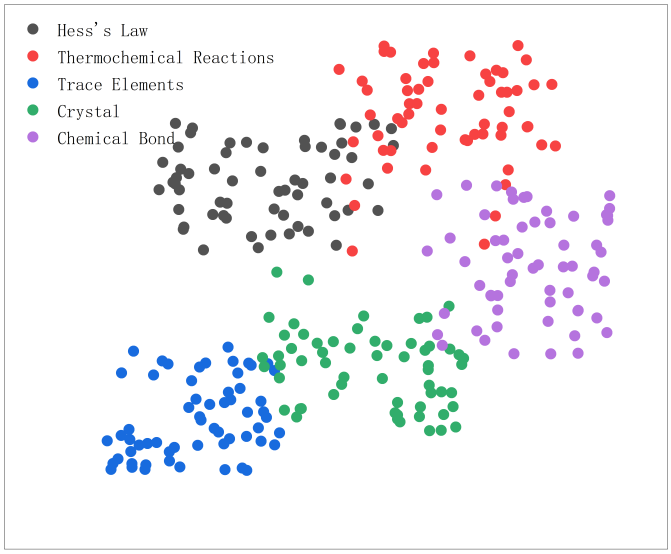

Figure 6. T-SNE Vector Visualization.

4.5. Time Analysis

To substantiate the efficiency of the MIFM model proposed in this paper for extracting features from multimodal data, the experiment conducts training under the experimental environment and model parameters described in the "Experimental Environment and Parameters" section. The experiment utilizes a training dataset comprising approximately 180,000 samples and employed a computing system equipped with 32 GB of RAM, a NVIDIA GeForce RTX3060 graphics card, and an i5-12400f CPU, investing approximately 100 h in the training process to refine the MIFM model.

Subsequently, this model is applied to perform ten multimodal vectorization operations on a subset of 100 samples, each containing both text and image data. The average processing time, from input to output, is recorded at 16.7 s, with an average execution time of 0.0167 s per data for a single multimodal vectorization operation. The efficiency of the MIFM model owes much to the incorporation of the transformer architecture, particularly the multi-head attention mechanism, which enables independent weight computations for each position, facilitating the processing of the entire sequence in a single computation. Furthermore, the dual-stream architecture proposed in this paper further amplifies the model's prowess in parallelly processing multimodal data.

5. Ablation Studies

In this section, we conduct ablation experiments using the multimodal information fusion model MIFM to demonstrate the effectiveness of the model in extracting heterogeneous exercise features. These studies aim to emphasize the importance of exercise knowledge concepts and exercise accompanying images in addition to exercise text data for exercise characterization. Specifically, the performance of MIFM is examined when utilizing exercise information from different modalities as input data. MIFM-T, MIFM-I, and MIFM-K represent the input data of exercise text, exercise accompanying images, and exercise knowledge concepts, respectively. These experiments are performed on three distinct educational tasks to evaluate the performance of different input modalities.

The results presented in Tables 5–7 indicate that MIFM-T and MIFM-TI outperform the other input types across all three educational tasks, while MIFM-I and MIFM-K exhibit an inferior performance. Notably, MIFM-ALL, which incorporates all three exercise data inputs, achieves the best results across all tasks. These findings suggest that the text of the exercise contains the primary semantic information, whereas the knowledge concept and

accompanying image only provide partial or incomplete semantic information. Therefore, when unimodal data are used as model input, the resulting exercise characterization vector fails to fully capture the semantic information of the exercise. Only by combining data from different modalities as input can the model improve the task performance compared to using unimodal data alone. The ablation experiment underscores the significance of heterogeneous exercise features for exercise characterization and demonstrates the effectiveness of fusing multimodal data and knowledge information to enhance educational task performance.

Table 5. Ablation Study for Knowledge Mapping. Bold data is the best result under the same criterion.

	ACC	***Precision***	***Recall***	***F1***
$MIFM-T$	0.4582	0.7532	0.7213	0.7378
$MIFM-I$	0.1726	0.1912	0.2126	0.2001
$MIFM-K$	0.2013	0.2063	0.2142	0.2107
$MIFM-TI$	0.6873	0.7605	0.7352	0.7476
$MIFM-TK$	0.5158	0.7513	0.7079	0.7289
$MIFM-IK$	0.1986	0.2146	0.2523	0.2319
$MIFM-ALL$	**0.7235**	**0.8164**	**0.7583**	**0.8065**

Table 6. Ablation Study for Student Performance Prediction. Bold data is the best result under the same criterion.

	MAE	***RMSE***	***ACC***	***AUC***
$MIFM-T$	0.4536	0.4752	0.7528	0.5658
$MIFM-I$	0.4621	0.4821	0.6892	0.5485
$MIFM-K$	0.4660	0.4723	0.7013	0.5502
$MIFM-TI$	0.4221	0.4603	0.7664	0.6121
$MIFM-TK$	0.4375	0.4672	0.7592	0.5613
$MIFM-IK$	0.4487	0.4716	0.7121	0.5572
$MIFM-ALL$	**0.3512**	**0.4521**	**0.7736**	**0.6257**

Table 7. Ablation Study for Difficulty Estimation. Bold data is the best result under the same criterion.

	MAE	***RMSE***	***PCC***
$MIFM-T$	0.2216	0.2834	0.3341
$MIFM-I$	0.2351	0.2883	0.2042
$MIFM-K$	0.2406	0.2763	0.2215
$MIFM-TI$	0.2116	0.2720	0.3631
$MIFM-TK$	0.2195	0.2648	0.3586
$MIFM-IK$	0.2374	0.2761	0.2875
$MIFM-ALL$	**0.2076**	**0.2632**	**0.4683**

6. Conclusions

This paper focuses on generating a unified characterization vector that fuses heterogeneous features from input data with multimodal features. To accomplish this, a multimodal information fusion-based exercise characterization model called MIFM is proposed. MIFM effectively extracts and fuses features from multimodal data, allowing for the extraction of corresponding data features and the preservation of semantic information. A series of experiments is conducted to validate the importance of multimodal feature extraction and fusion, as well as the effectiveness of the multimodal exercise characterization model, MIFM.

While MIFM demonstrates a commendable performance across all three educational tasks, there remains room for improvement. In the context of heterogeneous feature fusion, this paper employs a cross-modal attention mechanism, which could lead to incomplete fusion when heterogeneous information lacks precise correspondence and high-quality exercise data. Subsequent research endeavors may involve the development of more

advanced methods for heterogeneous feature fusion, with the aim of effectively amalgamating data from diverse modalities. This paper predominantly focuses on three types of knowledge information: exercise text, exercise knowledge concepts, and accompanying exercise images. Future investigations may encompass exercise answers, students' exercise practicing records, and audio information for explaining exercises, thus enabling a more comprehensive expression of exercises and ensuring that the generated test representation vectors align more closely with the original semantic content of the exercises.

Author Contributions: Conceptualization, J.S.; methodology, J.S.; software, H.C.; validation, K.X.; data curation, H.C.; writing—original draft preparation, H.C.; writing—review and editing, C.L.; supervision, K.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundations of China under grant No. 62272364.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Acknowledgments: The authors would like to thank the Assistant Editor of this article and anonymous reviewers for their valuable suggestions and comments.

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

References

1. Thakur, N. A large-scale dataset of Twitter chatter about online learning during the current COVID-19 Omicron wave. *Data* **2022**, *7*, 109. [CrossRef]
2. Boca, G.D. Factors influencing students' behavior and attitude towards online education during COVID-19. *Sustainability* **2021**, *13*, 7469. [CrossRef]
3. Chatterjee, A.; Gupta, U.; Chinnakotla, M.K.; Srikanth, R.; Galley, M.; Agrawal, P. Understanding emotions in text using deep learning and big data. *Comput. Hum. Behav.* **2019**, *93*, 309–317. [CrossRef]
4. Baltrušaitis, T.; Ahuja, C.; Morency, L.P. Multimodal machine learning: A survey and taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 423–443. [CrossRef]
5. Qin, Z.; Zhao, Y. Correlation analysis of mathematical knowledge points based on word co-occurrence and clustering. In Proceedings of the 2020 International Conference on Computers, Information Processing and Advanced Education (CIPAE), Ottawa, ON, Canada, 16–18 October 2020; pp. 47–52. [CrossRef]
6. Liu, Y.; Yi, X.; Chen, R.; Zhai, Z.; Gu, J. Feature extraction based on information gain and sequential pattern for English question classification. *IET Softw.* **2018**, *12*, 520–526. [CrossRef]
7. Huo, Y.; Wong, D.F.; Ni, L.M.; Chao, L.S.; Zhang, J. Knowledge modeling via contextualized representations for LSTM-based personalized exercise recommendation. *Inf. Sci.* **2020**, *523*, 266–278. [CrossRef]
8. Wang, L.; Sun, Y.; Zhu, Z. Knowledge points extraction of junior high school english exercises based on SVM method. In Proceedings of the 2018 2nd International Conference on E-Education, E-Business and E-Technology, Beijing, China, 5–7 July 2018; pp. 43–47. [CrossRef]
9. Shahmirzadi, O.; Lugowski, A.; Younge, K. Text similarity in vector space models: A comparative study. In Proceedings of the 2019 18th IEEE International Conference on Machine Learning and Applications (ICMLA), Boca Raton, FL, USA, 16–19 December 2019; pp. 659–666. [CrossRef]
10. Zhang, L.; Wang, J.; Zhang, Y.; Fu, A.; Xu, D. Research on the multi-source information deduplication method based on named entity recognition. In Proceedings of the 2022 8th International Conference on Big Data and Information Analytics (BigDIA), Guiyang, China, 11–12 August 2022; pp. 479–484. [CrossRef]
11. Huang, Z.; Liu, Q.; Chen, E.; Zhao, H.; Gao, M.; Wei, S.; Su, Y.; Hu, G. Question Difficulty Prediction for READING Problems in Standard Tests. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017; Volume 31, pp. 1352–1359. [CrossRef]
12. Hermann, K.M.; Kocisky, T.; Grefenstette, E.; Espeholt, L.; Kay, W.; Suleyman, M.; Blunsom, P. Teaching machines to read and comprehend. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 1693–1701.
13. Liu, Q.; Huang, Z.; Huang, Z.; Liu, C.; Chen, E.; Su, Y.; Hu, G. Finding similar exercises in online education systems. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; pp. 1821–1830. [CrossRef]

14. Zhao, D.; Liu, Y. A Multimodal Model for College English Teaching Using Text and Image Feature Extraction. *Comput. Intell. Neurosci.* **2022**, *2022*, 3601545. [CrossRef]
15. Ochoa, X.; Chiluiza, K.; Méndez, G.; Luzardo, G.; Guamán, B.; Castells, J. Expertise estimation based on simple multimodal features. In Proceedings of the 15th ACM on International Conference on Multimodal Interaction, Sydney, Australia, 9–13 December 2013; pp. 583–590. [CrossRef]
16. Penuel, W.R. Research–practice partnerships as a strategy for promoting equitable science teaching and learning through leveraging everyday science. *Sci. Educ.* **2017**, *101*, 520–525. [CrossRef]
17. Jalilifard, A.; Caridá, V.F.; Mansano, A.F.; Cristo, R.S.; da Fonseca, F.P.C. Semantic sensitive TF-IDF to determine word relevance in documents. In *Advances in Computing and Network Communications: Proceedings of CoCoNet 2020*; Springer: Berlin/Heidelberg, Germany, 2021; Volume 2, pp. 327–337. [CrossRef]
18. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781. [CrossRef]
19. Graves, A. Generating sequences with recurrent neural networks. *arXiv* **2013**, arXiv:1308.0850. [CrossRef]
20. Wang, R.; Li, Z.; Cao, J.; Chen, T.; Wang, L. Convolutional recurrent neural networks for text classification. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–6. [CrossRef]
21. 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*, 6000–6010.
22. Li, Y.; Qian, Y.; Yu, Y.; Qin, X.; Zhang, C.; Liu, Y.; Yao, K.; Han, J.; Liu, J.; Ding, E. Structext: Structured text understanding with multi-modal transformers. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event, 20 October 2021; pp. 1912–1920. [CrossRef]
23. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708. [CrossRef]
24. Bank, D.; Koenigstein, N.; Giryes, R. Autoencoders. *arXiv* **2020**, arXiv:2003.05991. https://doi.org/10.48550/arXiv.2003.05991.
25. Noh, H.; Hong, S.; Han, B. Learning deconvolution network for semantic segmentation. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1520–1528. [CrossRef]
26. Cheng, H.T.; Koc, L.; Harmsen, J.; Shaked, T.; Chandra, T.; Aradhye, H.; Anderson, G.; Corrado, G.; Chai, W.; Ispir, M.; et al. Wide & deep learning for recommender systems. In Proceedings of the 1st Workshop on Deep Learning for Recommender Systems, Boston, MA, USA, 15 September 2016; pp. 7–10. [CrossRef]
27. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543. [CrossRef]
28. Graves, A.; Schmidhuber, J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw.* **2005**, *18*, 602–610. [CrossRef]
29. Huang, P.Y.; Chang, X.; Hauptmann, A. Multi-head attention with diversity for learning grounded multilingual multimodal representations. *arXiv* **2019**, arXiv:1910.00058. [CrossRef]
30. Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A.; Hovy, E. Hierarchical attention networks for document classification. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, USA, 12–17 June 2016; pp. 1480–1489. [CrossRef]
31. Su, Y.; Liu, Q.; Liu, Q.; Huang, Z.; Yin, Y.; Chen, E.; Ding, C.; Wei, S.; Hu, G. Exercise-enhanced sequential modeling for student performance prediction. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32, pp. 2435–2443. [CrossRef]
32. Devassy, B.M.; George, S. Dimensionality reduction and visualisation of hyperspectral ink data using t-SNE. *Forensic Sci. Int.* **2020**, *311*, 110194. [CrossRef] [PubMed]
33. Sarzynska-Wawer, J.; Wawer, A.; Pawlak, A.; Szymanowska, J.; Stefaniak, I.; Jarkiewicz, M.; Okruszek, L. Detecting formal thought disorder by deep contextualized word representations. *Psychiatry Res.* **2021**, *304*, 114135. [CrossRef] [PubMed]
34. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805. [CrossRef]
35. Ma, L.; Lu, Z.; Li, H. Learning to answer questions from image using convolutional neural network. In Proceedings of the AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; Volume 30, pp. 3567–3573. [CrossRef]