Automatic Recognition and Labeling of Knowledge Points in Learning Test Questions Based on Deep-Walk Image Data Mining

Ying Chang, Qinghua Zhu
Beijing Polytechnic, Beijing 100176, China

***Abstract*—This paper deeply studies and discusses the application of image data mining technology based on the Deep-Walk algorithm in automatic recognition and annotation of knowledge points in learning test questions. With the rapid development of educational informatization, how to effectively mine and label the knowledge points in learning test questions from image data has become an urgent problem to be solved. In this paper, we introduce a novel approach that integrates graph embedding technology with natural language processing techniques. Initially, we leverage the Deep-Walk algorithm to embed the knowledge points present in the test question images, effectively transforming the high-dimensional image data into a low-dimensional vector representation. This transformation meticulously preserves the intricate structural information while meticulously capturing the subtle semantic nuances embedded within the image data. Subsequently, we undertake a thorough semantic analysis of these vectors, seamlessly integrating natural language processing techniques, to facilitate automated recognition with unparalleled precision. This innovative methodology not only elevates the accuracy of knowledge point recognition to new heights but also achieves semantic annotation of these points, thereby furnishing richer, more insightful data support for subsequent intelligent education applications. Through experimental verification, the proposed method has achieved remarkable results on multiple data sets, which proves its feasibility and effectiveness in practical applications. Furthermore, this paper delves into the expansive potential applications of this methodology in the realm of image data mining, encompassing areas such as online education, intelligent tutoring systems, personalized learning frameworks, and numerous other domains. As we look ahead, we aim to refine the algorithm, enhance recognition accuracy, and uncover additional application scenarios, thereby contributing significantly to the intelligent evolution of the education sector.**

***Keywords*—Deep-walk; image data mining; study test questions; knowledge point recognition**

I. Introduction

As human society progresses towards the era of intelligence, education in my country is undergoing a paradigm shift from traditional models to intelligent education. However, the pressing issue of imbalanced and inadequate education development persists. Recently, various government ministries and commissions have enacted a series of pertinent policies [1, 2] to ensure and expedite this educational transformation. These policies aim to harness the power of artificial intelligence, big data, blockchain, and 5G technology to drive innovation and expedite the evolution of digital education. They strive to unlock the latent potential of digital education, explore novel governance approaches, and seize the opportunities presented by the digital revolution for future educational transformation. Additionally, these policies promote the utilization of information technology to innovate teaching methods, develop teaching aids that align with educational needs, and leverage artificial intelligence to provide teachers with comprehensive assistance in tasks such as resource sourcing, homework grading, and online question answering. Ultimately, the goal is to establish an intelligent, efficient, and comprehensive educational analysis system.

In 2006, Hinton published an article titled "Reducing the dimension of data with neural networks" in "Science" magazine, which opened the prelude to the field of deep learning [3, 4]. Since then, research and applications based on deep learning have achieved great success in many fields, such as speech recognition, object visual recognition and target detection. In this disruptive wave of technology, the field of education has also been greatly impacted. Big data and artificial intelligence technologies can be applied to all stages of the field of education, providing support for new learning methods, improving teaching level and teaching quality, and improving Teaching efficiency and teaching effect, reducing the burden on teachers and students. A large number of intelligent education programs have been applied to actual education and teaching, enriching the forms of education and teaching at this stage. With the large-scale development of online education during the epidemic, this form of educational organization based on the Internet has further entered people's lives. Online education actively responds to the challenges brought by the global COVID-19 pandemic to the way education is organized, helping to minimize the impact of the epidemic on normal teaching order. At the same time, online education expands the supply of educational resources, reduces the differences in education levels in different regions, and further realizes educational equity [5, 6].

In the process of carrying out learning and evaluating learning, subject test questions play a vital role. By mining the deep hidden information in online test questions, it can help teachers and students build an intelligent learning environment and reduce the burden. For example, by analyzing the similarity of the hidden information in the test text, it can quickly locate the approximate question type; Using the hidden information of test text to build an intelligent question bank system, further realize a more intelligent and balanced automatic test paper generation model. Among them, the knowledge points labeling of test questions is the basic work of building an intelligent question bank system.

The labeling of knowledge points in test questions refers to the process of labeling the knowledge points used in answering test questions through a certain method. In teaching activities, knowledge points refer to the basic organizational units and transmission units that transmit teaching information, including concepts, definitions and theorems. A knowledge point is a general description of a certain concept, which usually exists as a goal that teachers and students want to achieve together. There are mainly five types of relationships between different knowledge points, namely, hierarchical relationship, dependent relationship, implication relationship, association relationship and dissociative relationship [7, 8]. Hierarchical relationship refers to the form that the upper layer contains the lower layer, and the lower layer belongs to the previous form. These knowledge points are interrelated and usually form a tree structure. According to the tree structure, different knowledge points can be described as a father-son relationship or a brother relationship.

The traditional method of labeling knowledge points of test questions is mainly manual labeling. Usually, front-line teachers with rich teaching and research experience are used as labeling personnel to label knowledge points of test questions. However, artificial knowledge point labeling methods are highly subjective, usually have cognitive biases, and the accuracy is difficult to guarantee. Furthermore, manual labeling fails to leverage the full potential of labeled test question data efficiently, as incremental test question data continues to require time-consuming manual annotation, hindering the pursuit of more efficient solutions. Consequently, amidst the exponential growth of online test question resources, the costs associated with manual knowledge point labeling—both in terms of time and human resources—have skyrocketed, underscoring the pressing need for alternative approaches.

With the application of artificial intelligence technology in various fields, some scholars use deep text mining technology to try to automatically label knowledge points and have achieved certain results. The method based on deep text mining can automatically learn and discover the potential features of test text from the test question data set and can label the appropriate knowledge points for the test questions according to the learned features. Therefore, it can better solve the problem of manual labeling costs caused by the explosive growth of online test question resources, and help automatically build an intelligent test bank system [9, 10].

Most of the existing knowledge point labeling methods of test questions are migrated from general text categorization methods. Test questions refer to a class of question texts used to test teaching effects. They have strong professionalism and certain structure and are different from texts in general fields. Therefore, methods in general fields cannot be simply transferred directly to knowledge point labeling field to complete labeling. Compared with the text in the general field, the test text contains more types of knowledge point labels, the sparsity is higher, and the workload and difficulty of the test question knowledge point labeling task are greater.

II. Overview of Related Technologies and Theories

A. *Text Preprocessing*

Text preprocessing is a necessary step in text categorization task, and its effect affects the effect of text categorization to a certain extent. Therefore, different preprocessing methods are generally adopted according to the needs of text categorization tasks. The text of test questions generally has strong specialization and structure, so it is not easy to transfer the preprocessing method of general domain text to the field of test question text processing directly. Therefore, this paper mainly carries out the following operations in terms of text preprocessing:

1) Data cleansing: Cleanse task-independent text in the original text, and design regular expressions to match and delete this part of the text.

2) Chinese word segmentation: Chinese word segmentation methods are broadly categorized into two primary approaches: character-based and word-based, based on the granularity of segmentation. Each of these methodologies carries its unique set of advantages and constraints, underscoring the importance of selecting the optimal granularity tailored precisely to the demands of a given task. Entity recognition demands a high degree of accuracy in word segmentation, as the precision of this initial step directly impacts the overall effectiveness of these downstream applications. Therefore, these tasks usually choose character-based word segmentation methods [11, 12]. In order to achieve better performance, such tasks as text categorization, sentiment analysis, and text summarization, which pay more attention to text semantic understanding, usually choose word-based word segmentation methods. The process of knowledge point identification and labeling is shown in Fig. 1.

3) De-stop words: The accuracy of text classification can be improved by removing stop words in the text. Stop words refer to the words that appear frequently in categorized texts but have little effect on helping to improve the classification effect. For example, and the land of in English text, these words can be found in almost every sentence, but they do not provide much help in the semantic understanding of the sentence. Studies have shown that in a small English paragraph, more than 50% of the words are contained in a list of 135 commonly used words, which are generally considered noise words and should be removed during the text preprocessing stage [13]. There are also many such words in Chinese text. For example, stop words such as He, Ruo, can provide very little information for text categorization tasks, but often introduce more noise information. Deleting stop words helps significantly reduce the size of the text feature space, speed up model calculation and improve the accuracy of text categorization [14].

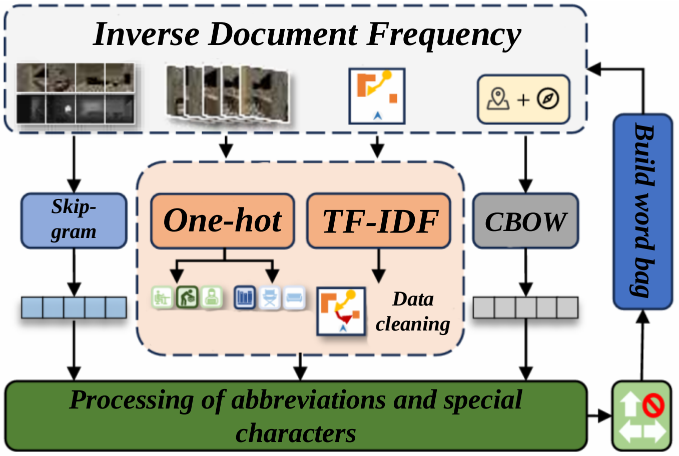
Inverse Document Frequency
Skip-gram
One-hot
TF-IDF
CBOW
Data cleaning
Build word bag
Processing of abbreviations and special characters

Fig. 1. The process of knowledge point identification and labeling.

4) Handling abbreviations and special characters: In the process of daily life and learning, some words are often set as their own abbreviations. The original meaning of these abbreviations is for the convenience of memory, but after abbreviation, part of the semantic information of the original words will be lost, which is not conducive to the learning of classification models [15]. Therefore, it is necessary to convert these abbreviated words in the processing stage to restore their original text expressions. Most text datasets will contain many unnecessary characters, such as punctuation marks and special characters. These punctuation marks and special characters are very important for human wording, sentence breaking, understanding, etc., but these are not helpful to improve the performance of classification algorithms, and will bring a lot of noise to the learning of the model. Choose to remove these punctuation and characters. The node embedding formula in the Deep Walk model is shown in Eq. (1).

$$W(d,t) = TF(d,t) * log\left(\frac{N}{df(t)}\right) \tag{1}$$

Here, N is the number of documents, and df(t) is the number of documents in the corpus that contain the word t. The first term in the equation improves the recall rate, while the second term improves the precision of the word embedding. Although TF-IDF reduces the problems caused by high-frequency words in documents to a certain extent, it also ignores the relationship between words in the text, and directly ignores the semantic information of words.

Word2Vec is a popular word embedding method that captures the relationship between words in context and embeds words into Euclidean space [16]. The Word2Vec method uses two shallow neural networks with continuous word bags (CBOW) and Skip-gram to create a vector for each word in the library. The day scale of the Skip-gram model is to maximize the probability in Eq. (2).

$$\underset{\theta}{argmax} \prod_{w \in T} [\prod_{c \in c(w)} p(c \mid w; \theta)] \tag{2}$$

Fig. 2 shows image data mining and processing process. As Fig. 2 shows, the purpose of the CBOW model and the Skip-gram model are different. While the CBOW model is tasked with finding words based on a sequence of words, the Skip-gram model is tasked with finding the words most likely to appear near a given word based on that word. Word2Vec has greatly advanced the field of natural language understanding by providing a very powerful tool for capturing similarity relationships between words in a corpus [17].

Prior to delving into Word Frequency-Inverse Document Frequency (TF-IDF), it is imperative to grasp the fundamentals of the Bag of Words (BoW) model, which serves as a cornerstone for text representation. The BoW model simplifies text by transforming documents or sentences into a concise list of word frequencies. This list is compiled during the model's construction process, where each unique word in the vocabulary is first encoded into a one-hot encoding vector. For instance, in the given scenario, assuming a vocabulary size of 19, the model would generate a 19-dimensional vector for each word, with a '1' occupying the position corresponding to the word's index in the vocabulary and all other positions set to '0'. This encoding scheme provides a straightforward yet effective way to represent textual data. Then the bag-of-words model combines word frequency as a feature representation of the document and sentence. However, the word bag model only pays attention to the feature of word frequency when collecting and constructing word bags, and ignores the semantic relationship between words, which cannot help the model learn the deep-seated semantic information of the text well [18].

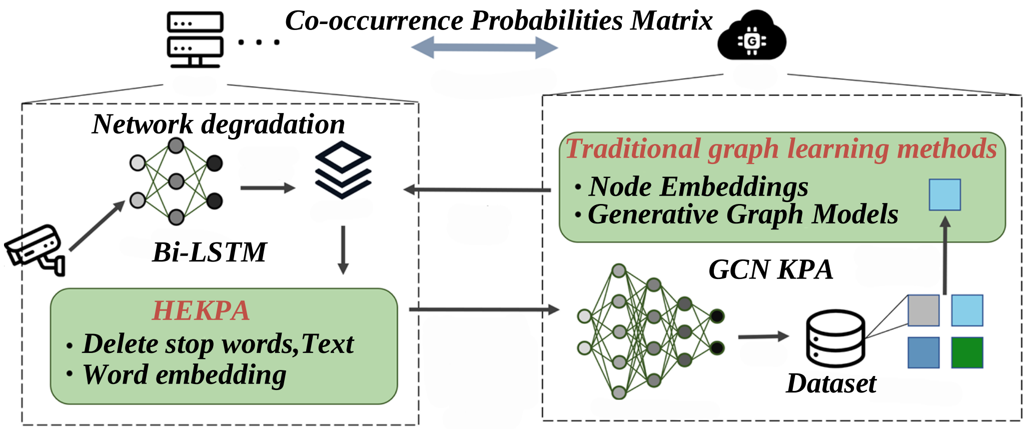

Fig. 2. Image data mining and processing process.

Inverse Document Frequency (IDF) is often used in conjunction with word frequency to reduce the negative impact of frequent words in the corpus. IDF assigns higher weight to high-frequency or low-frequency words in a document: this combination of TF and IDF is called word frequency-inverse document frequency (TF-IDF). Its word embedding is obtained by Eq. (3) as follows:

$$R_{\text{err}} = \sum_{i=1}^{n} \arccos(\frac{\text{tr}(R_{\text{out}_i}^{T} R_{\text{gt}_i}) - 1}{2}) \tag{3}$$

B. Glove

Another powerful and widely used model is Glove, which calculates word embedding by counting the number of global word co-occurrences across large corpora [19]. Before introducing the Glove model, let's introduce the latent semantic analysis algorithm based on singular value decomposition. This algorithm obtains the vector representation of words and documents by performing singular value decomposition on the word-document matrix. The Glove model combines the ideas and methods of latent semantic analysis algorithm and Word2Vec algorithm, because the author believes that both methods have certain defects. While the Latent Semantic Analysis (LSA) algorithm effectively harnesses global statistical information, its performance in word analogy tasks falls short, suggesting that there is room for enhancement in the generated vectors. In contrast, Word2Vec excels in word analogy tasks, albeit with minimal reliance on corpus statistics, underscoring its unique strengths in this domain [20]. The Glove model combines these two features together, and uses global statistical features and local contextual features of the corpus to help generate a vector representation of the text. For this reason, the Glove model introduces a Co-occurrence Probability Matrix (Co-occurrence Probability Matrix) to achieve this goal. First, the concept of a co-occurrence matrix is introduced. In the co-occurrence matrix X, the rows and columns of the matrix are words in the dictionary. Use xi,j to represent the number of times the word j appears in the context of the word i (usually a window size is set to specify the search distance of the context). The meaning of xi is the number of times the word i appears in the corpus. The co-occurrence probability matrix is obtained by counting the above two values, where the probabilities Pi,j are defined as shown in Eq. (4).

$$P_{ij} = \frac{x_{ij}}{x_i} \tag{4}$$

That is, P is the ratio of the number of times the word appears in the context of word i in the corpus to the total number of times the word i appears. Assuming i = ice, j = steam, k = solid, the feature extraction formula in image data mining is shown in Eq. (5).

$$Ratio = \frac{P_{ik}}{P_{jk}} \tag{5}$$

Using Ratio can also well reflect the relationship between i, j, and k (because the co-occurrence probability Ratio conforms to common sense), so the original author assumed that the word vector of i, j, and k generated by the Glove model can fit this Ratio after some calculation. Make the word vector obtained by Glove consistent with the co-occurrence matrix, so as to reflect the co-occurrence relationship between words, that is, the goal of Glove is defined as shown in Eq. (6).

$$f(w_i - w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}} \tag{6}$$

C. Graph Representation Learning

As a classic data structure, Graph is widely used in various fields of natural science. It is generally believed that a Graph is a set of objects (nodes) and interactions (edges) between objects [21]. The nodes in the social network graph are usually used to represent individuals, and the edges inside are used to indicate that there is a certain connection between two people.

Based on graphs, we can analyze, understand and learn complex systems in the real world. In the past dozens of works, many high-quality large-scale graph data have emerged, such as large-scale social network graphs based on social software, knowledge graphs for general fields, Internet network device topology graphs, and so on. The appearance of these large-scale graphs has greatly promoted the development of graph technology, among which the methods based on machine learning are particularly prominent. Machine learning provides many technical means for modeling, analyzing and understanding this part of large-scale and complex graph data, helping people to further explore and discover the theory and knowledge existing in the complex system behind these large-scale graphs.

Before discussing machine learning methods applied to graphs, a formal definition of what exactly means "graph data" is needed. Formally, the graph G=(V, E) consists of a set of nodes V and a set of edges E between these nodes, and represents the edges from node a∈V to node b∈V as (a, b)∈ E. The formula for the ReLU activation function is shown in Eq. (7).

$$\mathrm{E}[D_{\theta}(x_0 + n; \sigma, c) - x_{02}^{2}] \tag{7}$$

A simple way to store a graph is through the adjacency matrix A ∈ RV, where V is the number of nodes. Each row and column of the adjacency matrix represents a specific node; A in adjacency matrix; To represent the-edge from node i to node j, if (i, j)∈E then Ai, j = 1, otherwise Ai, j = 0. The elements Ai, j in the adjacency matrix can also store any real value instead of 0 and 1. At this time, the real value stored by Ai, j is the weight of the edge (i, j)∈E.

Nodes in the graph usually also have their specific attributes or feature information (for example, profiles and pictures of users in social networks), and in most cases, the real-valued matrix F∈Rd is used to represent the nodes. Attributes or features, the order of the nodes in the real-valued matrix is consistent with the order of the rows and rows in the adjacency matrix. The vector representation of the feature is stored in the real-valued matrix, and d is the dimension of the feature vector. The operational formula for the pooling layer and the weight update formula for the fully connected layer are shown in Eq. (8) and Eq. (9).

$$D_w(x; \sigma, c) = wD_1(x; \sigma, c) + (1 - w)D_0(x; \sigma, c) \tag{8}$$

$$\hat{y}_{j,T_n} = \sum_i w_{ij} \hat{y}_{i,T_n} \tag{9}$$

Graph learning algorithms mainly have two stages of development. The first is the traditional graph learning method based on statistics, and the progressive development is the graph representation learning algorithm based on machine learning. Traditional graph learning methods are basically based on the statistical information of nodes and graphs, which requires a lot of feature engineering, so the information is limited. At the same time, the statistical information designed by hand in traditional graph learning methods is not flexible and cannot be adapted in the learning process, so it needs to be redesigned after the task is shifted. With the development of machine learning, a series of methods have emerged that can get rid of manual feature design and learn features in graphs through adaptive methods-graph representation learning. Existing graph representation learning algorithms are mainly divided into three categories: Node Embeddings, GNN, and Generative Graph Models. The research content of this paper mainly involves node embedding and graph neural networks, and does not involve graph generation models. Therefore, the following will focus on the two-graph representation learning algorithms involved in the text.

III. A Knowledge Point Annotation Model Based on Mixed Label Embedding

A. *Test Question Text Data*

Test questions refer to a class of question texts used to test the teaching effect, so test text is more professional than daily text, and the format of test text is usually relatively fixed, for multiple-choice questions, fill-in-the-blank questions, or answer questions. format. Generally speaking, test text data has the following two characteristics:

1) Professionalism: The text of test questions demands unwavering professionalism, stemming from their intended purpose. The descriptions within these questions must adhere to stringent standards of accuracy and clarity, with no room for ambiguity or vague expressions. When juxtaposed with everyday language, the text of test questions typically encompasses a greater abundance of subject-specific proper nouns, thereby presenting a unique challenge that to some degree complicates the migration of general domain models into this specialized context. The normalization formula and the threshold processing formula in the preprocessing are shown in Eq. (10) and Eq. (11).

$$q_i(v) = \pi(K_i R_i^T (v - t_i)) \tag{10}$$

$$c(p) = \sum_{i=1}^{N-1} I_i(p) w_i(p) \tag{11}$$

2) Structural: Most of the common test text can be divided into a certain type of question, and each type of question has its fixed format, which will lead to more meaningless symbols in the test questions, thus introducing noise to model learning, so it is necessary to remove these meaningless symbols. However, according to the specific task requirements, specific rules can also be used to extract the structure of the test questions, so that it can be used as additional information to help the model learn. Because the test text has strong specialization and a certain structure, it is not easy to transfer the text preprocessing methods in the general field to the test text for text preprocessing without modification. Fig. 3 shows the distribution map of the raw image dataset.

3) Data cleansing: The text of the test questions in the dataset used in this paper is obtained by crawling from Baidu Question Bank through crawlers. The original text in the dataset will contain some text that is irrelevant to the task. For this type of text, this paper designs regular expressions to match and delete this part of the text.

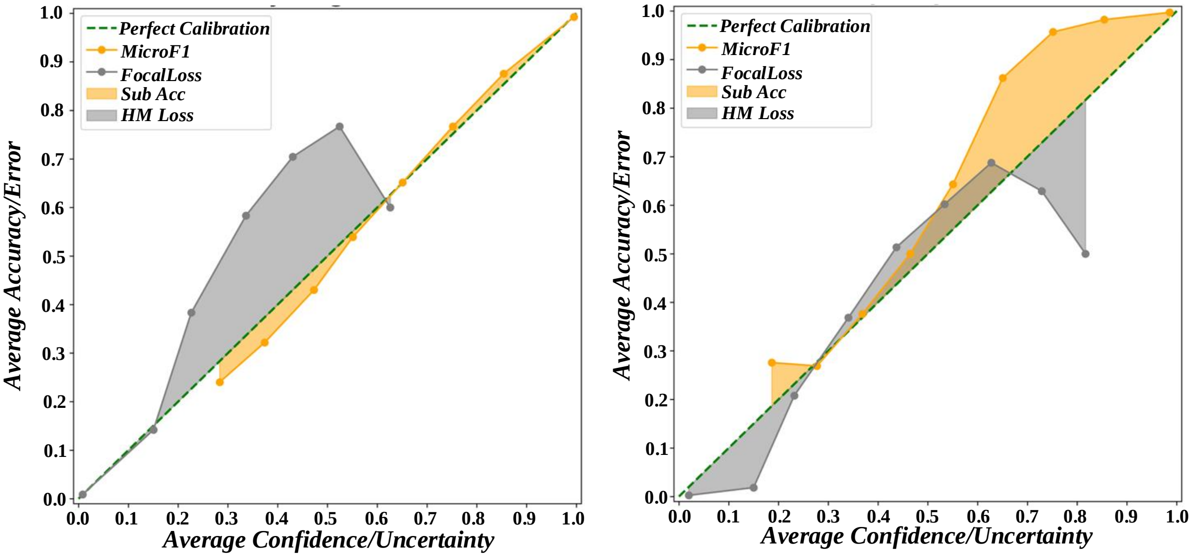

Fig. 3. Distribution map of the raw image dataset.

4) Chinese word segmentation: This paper selects word-based segmentation method to segment the text of the test questions, hoping that the model can better learn the deep-seated semantic information of the text. At present, the research of general Chinese text word segmentation has been relatively mature. There are many word segmentation tools to choose, such as NLPIR, LTP, THULAC, jieba, etc. This paper chooses jieba as a word segmentation tool to obtain word segmentation results. Due to the strong specialization of the test text, the test text usually contains a large number of proper nouns, but the original dictionaries in the existing jieba are designed for general fields, and there will be many mistakes in using the existing jieba directly to segment the test text. Therefore, this paper introduces subject dictionaries to help improve the accuracy of word segmentation and reduce the occurrence of word segmentation errors.

5) De-stop words: One of the basic methods to improve the accuracy of text categorization is to remove the stop words in the text. Stop words refer to the words that appear frequently in categorized texts but have little effect on helping to improve the classification effect. Stop words can provide very little information for text classification tasks, but they often introduce more noise information. Except stop words help to significantly reduce the size of text feature space, help to speed up model calculation and improve the accuracy of text classification. This article includes a list of 859 stop words, which contains most of the Chinese stop words, such as He, Ruo, Yu, Xi, etc.

6) Handling abbreviations and special characters: The inclusion of abbreviations and special characters did not contribute favorably to the model's ability to discern the syntactic and semantic nuances of test questions. Notably, in fields like biology, certain abbreviations of proper nouns are prevalent, designed primarily for mnemonic purposes. However, these abbreviations inherently entail the loss of a portion of the original words' semantic information, which hinders the model's learning process, as it struggles to grasp the full contextual meaning. Therefore, in the text processing stage, it is necessary to convert these abbreviated words to restore their original text expression. In the test text dataset, there are many special characters in addition to the commonly used punctuation marks. These special characters will affect the model's extraction of semantic information to a certain extent, which is not conducive to model learning. This paper will replace these special characters with blank characters in the text preprocessing stage. The gradient calculation formula and the similarity measure formula are shown in Eq. (12) and Eq. (13).

$$\left|B^{l}\right| = K^{2} * F \tag{12}$$

$$L_{ek} = \frac{1}{|P|}\sum_{p\in P}(\|\nabla sdf(p)\|_{2} - 1)^{2} \tag{13}$$

B. Model Construction

Inspired by the dictionary retrieval method, this paper proposes a two-stage automatic labeling model of knowledge points in test questions with mixed label embedding. The model is divided into a classification stage and a labeling stage. In the classification stage, the model classifies the test text into the second level of knowledge points in the knowledge point hierarchy diagram. In the labeling stage, the model obtains a label according to the results of the classification stage and further combines the node embedding and text embedding of the label according to the label-to-label knowledge points [22].

The co-occurrence relationship and the hierarchical relationship between knowledge points can be used as a priori knowledge to guide the model to label. The current method of automatic labeling of knowledge points has the problem of sparse label space. HEKPA guides the model to label knowledge points. The primary consideration is how to obtain the structural relationship.

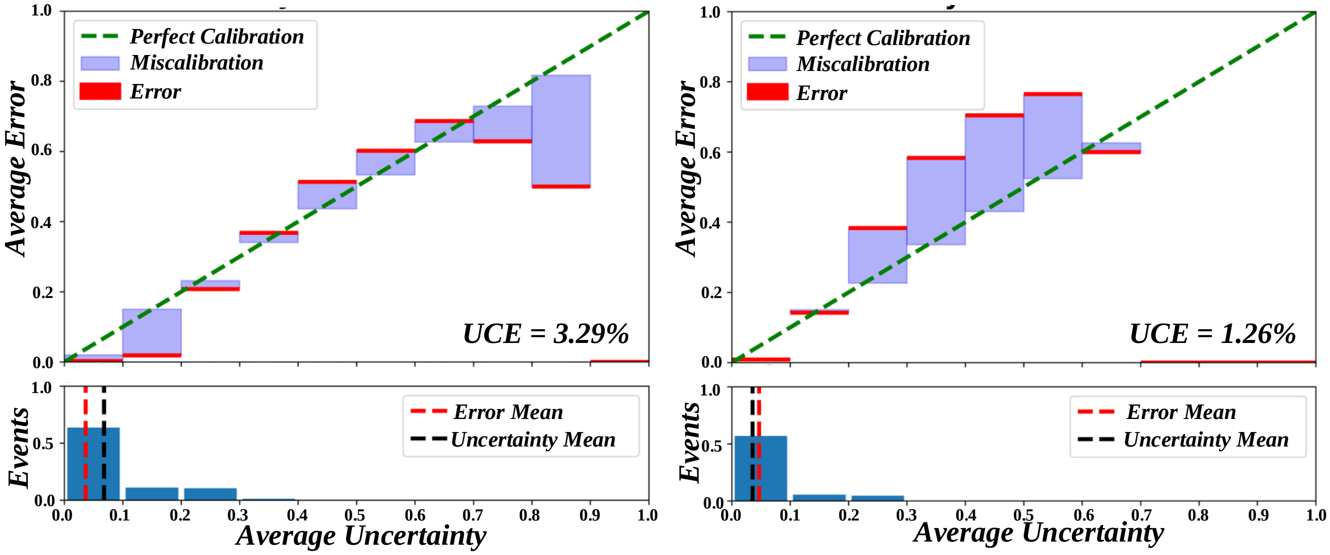

Fig. 4. Comparison diagram of the image feature extraction effect.

Fig. 4 shows comparison diagram of the image feature extraction effect. In Deep-Walk, the random walk algorithm is used to obtain the co-occurrence relationship of nodes in the graph. The results of the random walk with the vertex vi as the starting point are expressed as Wvi, = Wv1, Wv2, …, Wvi, where l is the predetermined random walk step size. After the random walk processes are obtained by using the random walk algorithm, these random walk processes are input into the Word2Vec algorithm as sentence sequences to learn the embedded representation of each node.

IV. A Knowledge Point Labeling Model Based on Graph Convolutional Neural Network

A. Model Construction

To overcome the limitations of shallow node embedding, this chapter presents an end-to-end model, GCN KPA, based on graph convolutional neural networks. This model captures the relationships between knowledge point labels. It features a feature extraction layer using Bi-LSTM to extract text features and a labeling layer that incorporates knowledge point label information using a GNN Network to accomplish labeling tasks.

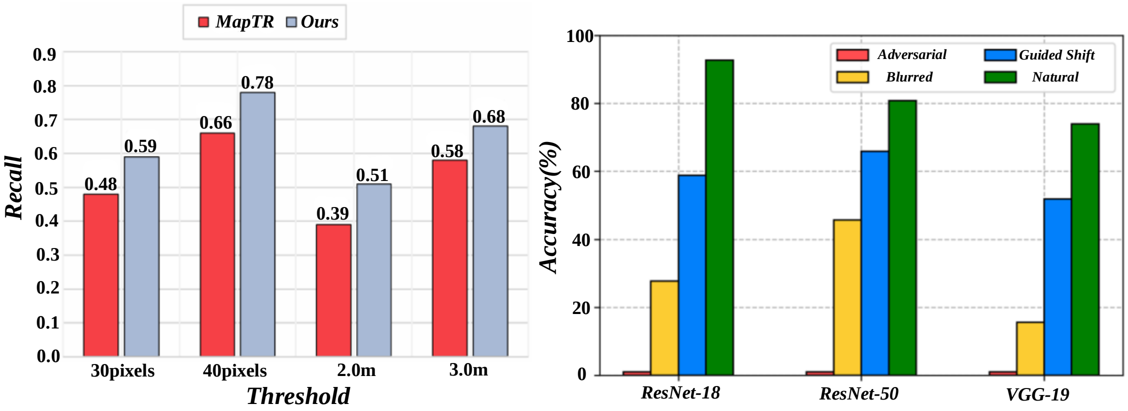

Fig. 5. Deep-walk node embedding visualization graph.

Fig. 5 displays the Deep-Walk Node embedding visualization. GCN-KPA utilizes Bi-LSTM to derive the text representation vector of test questions. Word embedding sequence X, pre-trained by a language model, is processed by Bi-LSTM to extract bidirectional text features. Eq. (14) and Eq. (15) depict random rotation and learning rate decay, respectively.

$$\vec{h}_i = \overline{LSTM}(\vec{h}_{i-1}, x_i) \tag{14}$$

$$L_{mfc} = \frac{1}{K}\sum_{k=0}^{K-1}(1 - NCC_k) \tag{15}$$

After acquiring latent semantic hidden states of the text in both directions, they are concatenated to form the final hidden representation of each word. Eq. (16) and Eq. (17) exhibit the dropout layer's mechanism and image classification accuracy calculation, respectively.

$$\hat{C} = \sum_{i=1}^{M} T_i \alpha_i c_i \tag{16}$$

$$T_i = \prod_{j=1}^{i-1}(1 - \alpha_j) \tag{17}$$

B. Callout Layer

The structural relationship between labels can help reduce the sparsity of label space and help guide the model to label knowledge points in test questions [23]. In this chapter, we design a classifier based on graph convolutional neural network, which usually has two parts: the eigenmatrix F ∈ Rq representing the nodes of the graph and the adjacency matrix A ∈Rq representing the edges of the graph.

Mula for the confusion matrix and the evaluation index is shown in Eq. (18).

$$G^{(l+1)} = \mathrm{ReLu}(\hat{A}G^{(l)}W^{(l)}) \tag{18}$$

However, only using a simple graph convolutional neural network as the classifier layer of the model will lead to a slower parameter update in the initialization stage, a lower learning rate of the model at the initial stage, and a model that cannot be learned through the back propagation algorithm for a period of time. GCN KPA further introduces Skip Connection to accelerate faster model initialization, the specific definition of the jump connection connection is shown in Eq. (19). The formula of the clustering algorithm in image data mining is shown in Eq. (20).

$$G_s^{(l+1)} = G^{(l+1)} + G^{(l)} \tag{19}$$

$$o = h \odot G_s \tag{20}$$

Fig. 6 shows classification of the model performance evaluation Fig. In a graph convolutional neural network. The calculation for

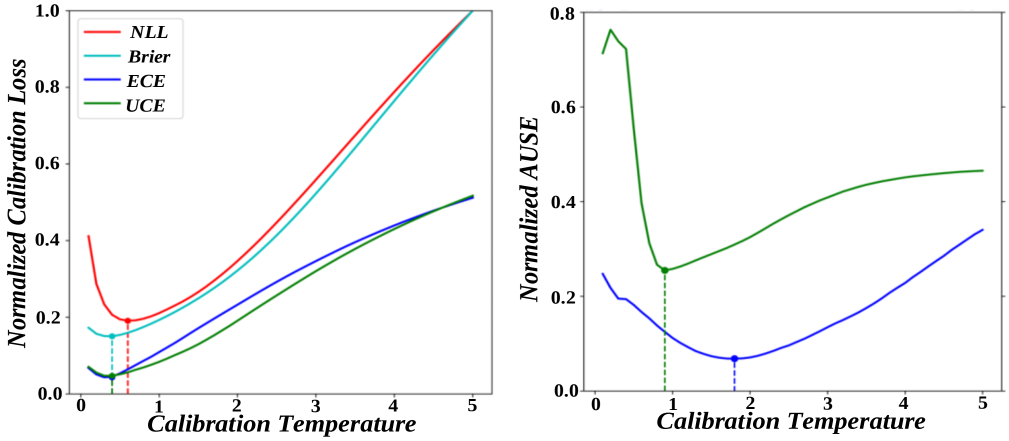

Fig. 6. Classification of the model performance evaluation.

C. Ablation Experiment

Among the three word embedding methods, TF-IDF, Glove and Word2Vec, using Word2Vec to obtain the word embedding representation sequence of text has the greatest improvement to the model, so this chapter defaults to using Word2Vec as the word embedding layer of the model to obtain the word embedding sequence of text [24]. In this section, three sets of ablation experiments were designed:

1) Feature extraction layer ablation experiment: The performance of three feature extraction layers in GCN KPA was studied and compared, with results shown in Fig. 7. Among them, Bi-LSTM performed best, followed by Text CNN, and MLP performed worst. Bi-LSTM extracts latent semantic features of text bidirectionally, hence chosen as the feature extraction layer in this model to enhance annotation accuracy. Fig. 7 displays ablation experiment results.

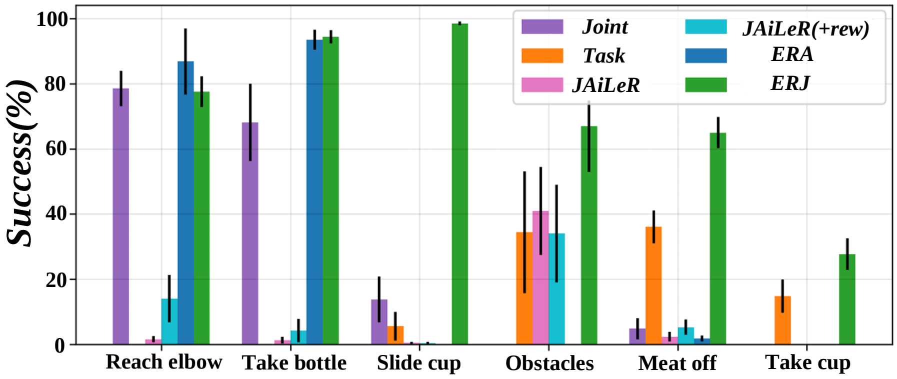

Fig. 7. Results of the ablation experiments of the feature extraction layer.

2) Classifier ablation experiment: Compare the effects of the GCNKPA separator with separate GCN separator and separate FC classifier, results are shown in Table I.

It can be seen from Table I that the model has been greatly improved, which shows that point labels introduced through the graph neural network can be very good [25]. Improve the labeling effect of the model. At the same time, the introduction of Skip Connection further improves the annotation effect of the model. And as shown in Fig. 8, introduce Skip Connection to accelerate model initialization.

TABLE I. COMPARISON OF THE CLASSIFIER EFFECTS

Classifier	Micro F1	Macro F1	H.M Loss	Sub Acc
FC	0.8640	0.7558	0.0116	0.5124
GCN	0.8817	0.8074	0.0100	0.5234
GCN KPA	0.8853	0.8206	0.0097	0.5339

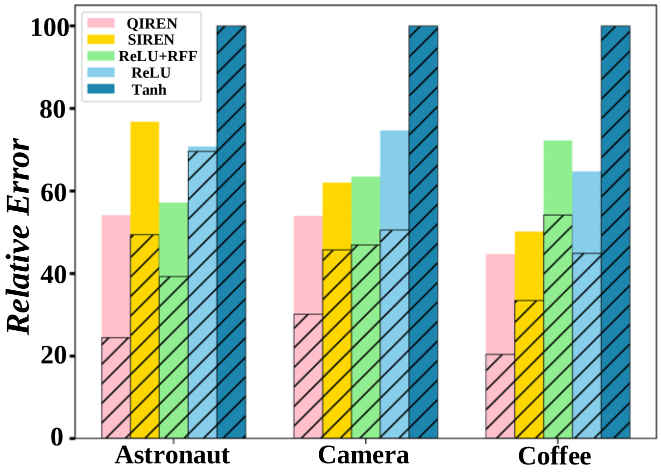

Fig. 8. The annotation effect diagram of the model.

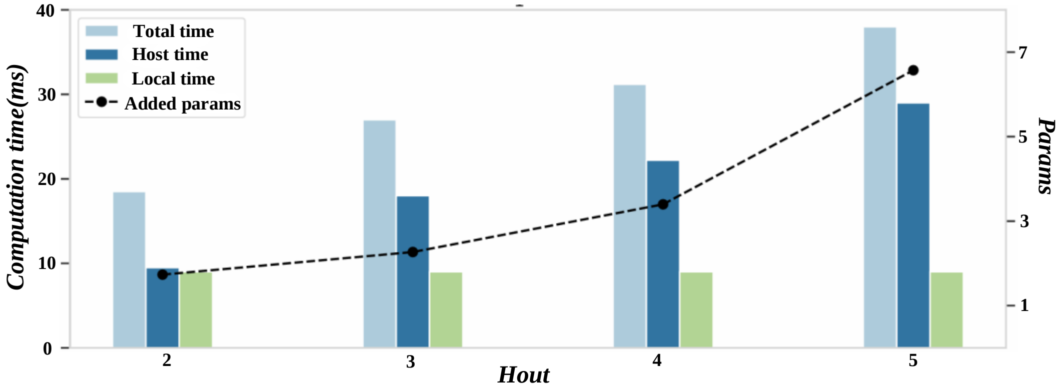

Fig. 9. Comparison diagram of knowledge point identification accuracy and annotation consistency.

Evidently, GCN KPA has demonstrated superior performance across all four evaluation metrics, with a particularly noteworthy enhancement in the MacroF1 indicator, surpassing the gains observed in the other three indicators. This underscores GCN KPA's proficiency in capturing the intricate relationships between labels, enabling it to harness this label information to guide the model's labeling process. Consequently, it mitigates the sparsity of the label space, resulting in a significant improvement in the efficacy of knowledge point labeling [26]. Fig. 9 shows comparison diagram of knowledge point identification accuracy and annotation consistency.

3) Comparison of different Loss functions: The introduction of the Focal Loss function reduces the impact of the unbalanced distribution of labels in the dataset on model learning to a certain extent, and observes the impact of different loss functions on model learning by comparing it with the BCE Loss function commonly used in multi-label text categorization. Detailed experimental results are shown in Table II.

As can be seen from Table II, after replacing BCELoss with Focal Loss, there is an improvement in the three evaluation indicators, but a decrease of 3% in Sub Acc, the most stringent evaluation indicator. This is consistent with the starting point of the Focal loss function design. Focal loss function gives different weights to different samples, reduces the weights of easy-to-classify samples, and allows categories with fewer samples to have higher weights. This makes the model pay more attention to point labels, which is reflected in the experimental results that the improvement rate of MacroF1 is greater than that of MicroF1. At the same time, because the model pays more attention to the part of the label with a small number of samples, the model is more aggressive than before, which leads to the poor performance of the model on the evaluation index Sub Acc. But from the overall experimental results, the introduction of Focal loss makes the model improve most of the evaluation indicators, and can better deal with data sets with unbalanced sample numbers, and better learn knowledge point labels with small sample numbers. Fig. 10 shows performance comparison of the model on different dataset.

TABLE II. EXPERIMENTAL RESULTS FOR THE DIFFERENT LOSS FUNCTIONS

Loss function	Micro F1	Macro F1	HM Loss	Sub Acc
BCE Loss	0.8779	0.8011	0.0106	0.5606
Focal Loss	0.8853	0.8206	0.0097	0.5339

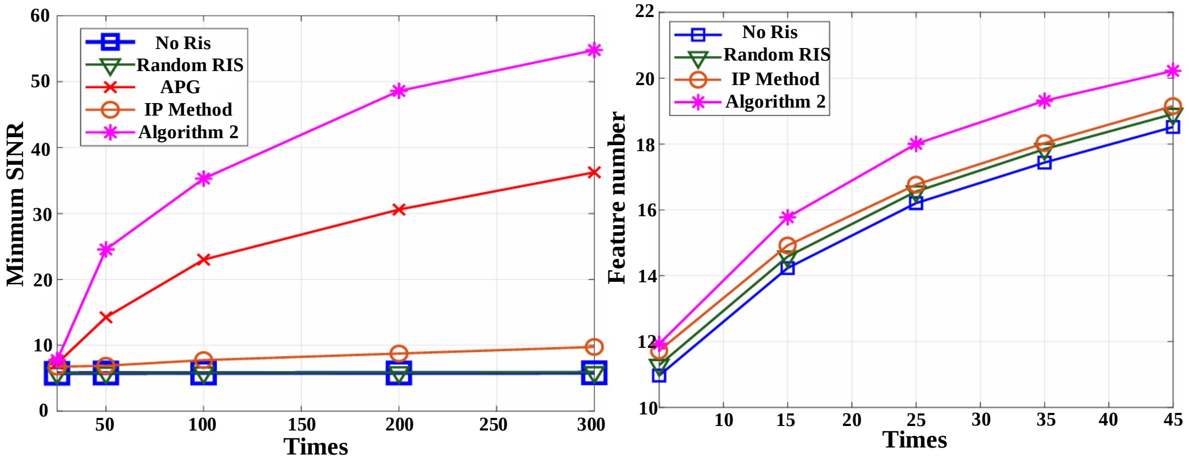

Fig. 10. Performance comparison of the model on different dataset.

V. CONCLUSION AND FUTURE WORK

Aiming at the research of automatic recognition and labeling of knowledge points in learning test questions based on Deep-Walk image data mining, a new method is proposed in this paper. This method combines graph embedding technology and advanced concepts in the field of natural language processing, and realizes the effective recognition and labeling of knowledge points in test questions in image data mining. By deeply exploring the application of Deep-Walk algorithm in graph data embedding, we successfully combine the visual features of images with the knowledge points of test questions, and realize the in-depth analysis and semantic understanding of image data.

The main conclusions of this study are as follows: First, Deep-Walk algorithm shows strong potential in the field of image data mining. It captures the topological structure of images through random walks, and then generates low-dimensional dense vector representations, which provides an effective means for the recognition of knowledge points in test questions. Secondly, combining the visual features of image data with the knowledge points of test questions cannot only improve the accuracy of recognition, but also enhance the semantic richness of labeling, providing strong support for intelligent applications in the field of education.

Anticipating the future, as image data mining technology continues to evolve, the Deep-Walk-based approach for knowledge point recognition and annotation is poised to find broader applications across diverse fields. Concurrently, we remain committed to delving deeper into the realm of algorithms and models, with the aim of enhancing recognition accuracy and refining the granularity of labeling, thereby pushing the boundaries of this technology further. In addition, we will also focus on the fusion of image data and other types of data, as well as cross-domain knowledge migration and sharing, laying a solid foundation for the realization of a wider range of intelligent educational applications.

To sum up, the research on automatic recognition and labeling of knowledge points in image data mining learning test questions based on Deep-Walk has achieved remarkable results, which provides a new idea and method for the intelligent development of education.

REFERENCES

[1] Bian, C., & Lu, S. Personalized recommendation of entertainment robots in fine arts education based on human–computer interaction and data mining. Entertainment Computing, vol. 51, pp. 100740, 2024.

[2] Cap, Q. H., Fukuda, A., Kagiwada, S., Uga, H., Iwasaki, N., & Iyatomi, H. Towards robust plant disease diagnosis with hard-sample re-mining strategy. Computers and Electronics in Agriculture, vol. 215, pp. 108375, 2023.

[3] Cerezo, R., Lara, J.-A., Azevedo, R., & Romero, C. reviewing the differences between learning analytics and educational data mining: Towards educational data science. Computers in Human Behavior, vol. 154, pp. 108155, 2024.

[4] Duque, J., Godinho, A., Moreira, J., & Vasconcelos, J. Data Science with Data Mining and Machine Learning A design science research approach. Procedia Computer Science, vol. 237, pp. 245–252, 2024.

[5] Gonzalez, L. F. P., Pivel, M. A. G., & Ruiz, D. D. A. Improving bathymetric images exploration: A data mining approach. Computers & Geosciences, vol. 54, pp. 142–147, 2013.

[6] Guo, Z., Yang, G., Wang, D., & Zhang, D. A data augmentation framework by mining structured features for fake face image detection. Computer Vision and Image Understanding, vol. 226, pp. 103587, 2023.

[7] Jiang, W., Yu, D., Xie, Z., Li, Y., Yuan, Z., & Lu, H. Trimap-guided feature mining and fusion network for natural image matting. Computer Vision and Image Understanding, vol. 230, pp. 103645, 2023.

[8] Jindal, K., & Kumar, R. A Note on “Data mining based noise diagnosis and fuzzy filter design for image processing”. Computers & Electrical Engineering, vol. 49, pp. 50–51, 2016.

[9] Kasat, N. R., & Thepade, S. D. Novel Content Based Image Classification Method Using LBG Vector Quantization Method with Bayes and Lazy Family Data Mining Classifiers. Procedia Computer Science, vol. 79, pp. 483–489, 2016.

[10] Lin, G., Wei, W., Kang, X., Liao, K., & Zhang, E. Deep graph layer information mining convolutional network. Pattern Recognition, vol. 154, pp. 110593, 2024.

[11] Liu, Y., Hu, S., Zhang, H., Dong, Q., & Liu, W. Intelligent mining methodology of product field failure data by fusing deep learning and association rules for after-sales service text. Engineering Applications of Artificial Intelligence, vol. 133, pp. 108303, 2024.

[12] Marshoodulla, S. Z., & Saha, G. A survey of data mining methodologies in the environment of IoT and its variants. Journal of Network and Computer Applications, vol. 228, pp. 103907, 2024.

[13] Raj, M. P., & Saini, J. R. A Novel Comparison of Charotar Region Wheat Variety Classification Techniques using Purely Tree-based Data Mining Algorithms. Procedia Computer Science, vol. 235, pp. 568–577, 2024.

[14] Tang, W. (2Application of support vector machine system introducing multiple submodels in data mining. Systems and Soft Computing, vol. 6, pp. 200096, 2024.

[15] Wang, C., Wang, G., Zhang, Q., Guo, P., Liu, W., & Wang, X. Eliminating and mining strategies for open-world object proposal. Neurocomputing, vol. 599, pp. 128026, 2024.

[16] Wang, Y., Wu, G., Chen, G. (Sheng), & Chai, T. Data mining based noise diagnosis and fuzzy filter design for image processing. Computers & Electrical Engineering, vol. 40(7), pp. 2038–2049, 2014.

[17] Xu, C., Lin, R., Cai, J., & Wang, S. Deep image clustering by fusing contrastive learning and neighbor relation mining. Knowledge-Based Systems, vol. 238, pp. 107967, 2022.

[18] Yang, Y., Lin, H., Guo, Z., & Jiang, J. A data mining approach for heavy rainfall forecasting based on satellite image sequence analysis. Computers & Geosciences, vol. 3(1),pp. 20–30, 2007.

[19] Zhang, J., & Dong, L. RETRACTED: Image Monitoring and Management of Hot Tourism Destination Based on Data Mining Technology in Big Data Environment. Microprocessors and Microsystems, vol. 80, pp. 103515, 2021.

[20] Zhang, J., Gruenwald, L., & Gertz, M. VDM-RS: A visual data mining system for exploring and classifying remotely sensed images. Computers & Geosciences, vol. 35(9), pp. 1827–1836, 2009.

[21] Zhang, K., Chen, K., & Fan, B.Massive picture retrieval system based on big data image mining. Future Generation Computer Systems, vol. 121, pp. 54–58, 2021.

[22] Zhang, X., Gu, N., Chang, J., Ye, H., Lin, C., & Shen, J.Mining discriminative spatial cues for aerial image quality assessment towards big data. Signal Processing: Image Communication, vol. 80, pp. 115646, 2020.

[23] Zhang, Z., Ning, L., Liu, Z., Yang, Q., & Ding, W. Mining and reasoning of data uncertainty-induced imprecision in deep image classification. Information Fusion, vol. 96, pp. 202–213, 2023.

[24] Dol, S. M., & Jawandhiya, P. M. Classification Technique and its Combination with Clustering and Association Rule Mining in Educational Data Mining—A survey. Engineering Applications of Artificial Intelligence, vol. 122, pp. 106071.

[25] El-Gharib, N. M., & Amyot, D. Robotic process automation using process mining—A systematic literature review. Data & Knowledge Engineering, vol. 148, pp. 102229, 2023.

[26] Wang, Z., Zhang, F., Ren, M., & Gao, D. A new multifractal-based deep learning model for text mining. Information Processing & Management, vol. 61(1), pp. 103561, 2024.