Applied Intelligence (2024) 54:9363–9380
https://doi.org/10.1007/s10489-024-05666-w

Dual-view graph convolutional network for multi-label text classification

Xiaohong Li[1] · Ben You[1] · Qixuan Peng[1] · Shaojie Feng[1]

Accepted: 30 June 2024 / Published online: 15 July 2024

Abstract
Multi-label text classification refers to assigning multiple relevant category labels to each text, which has been widely applied in the real world. To enhance the performance of multi-label text classification, most existing methods only focus on optimizing document and label representations, assuming accurate label-document similarity is crucial. However, whether the potential relevance between labels and if the problem of the long-tail distribution of labels could be solved are also key factors affecting the performance of multi-label classification. To this end, we propose a multi-label text classification model called DV-MLTC, which is based on a dual-view graph convolutional network to predict multiple labels for text. Specifically, we utilize graph convolutional neural networks to explore the potential correlation between labels in both the global and local views. First, we capture the global consistency of labels on the global label graph based on existing statistical information and generate label paths through a random walk algorithm to reconstruct the label graph. Then, to capture relationships between low-frequency co-occurring labels on the reconstructed graph, we guide the generation of reasonable co-occurring label pairs within the local neighborhood by utilizing the local consistency of labels, which also helps alleviate the long-tail distribution of labels. Finally, we integrate the global and local consistency of labels to address the problem of highly skewed distribution caused by incomplete label co-occurrence patterns in the label co-occurrence graph. The Evaluation shows that our proposed model achieves competitive results compared to existing state-of-the-art methods. Moreover, our model achieves a better balance between efficiency and performance.

Keywords Multi-label classification · Graph convolutional networks · Random walk model · Label co-occurrence · Label graph

1 Introduction

Multi-label text classification (MLTC) is a crucial task in natural language processing that finds applications in various domains, including sentiment analysis [1], patent classification [2], and question answering [3]. The primary objective of MLTC is to assign one or more appropriate categories to a document using a set of predefined categories or labels. In recent years, the MLTC has garnered significant attention and has become an active area of research. However, the increasing number of labels and documents, coupled with the complex interrelationships between labels and documents, pose significant challenges to MLTC. These challenges have prompted researchers to delve deeper into the field of multi-label learning.

Previous research on MLTC focused on developing enhanced document representations. Various methods have been proposed for learning label-specific document representations [4, 5]. Moreover, some studies have used attention mechanisms to capture label-semantic-based representations [6–8] and document-label interaction representations [9–12]. Although these approaches have shown promising results, they have not fully explored the interactions between label-

Ben You contributed equally to this work.

✉ Xiaohong Li
xiaohongli@nwnu.edu.cn

Ben You
2021222268@nwnu.edu.cn

Qixuan Peng
2021222181@nwnu.edu.cn

Shaojie Feng
2022222319@nwnu.edu.cn

[1] College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China

specific semantic components, thereby ignoring the rich label co-occurrence information within documents.

In recent years, label co-occurrence graph-based methods have gained attention for their ability to exploit statistical correlations between labels to construct label co-occurrence graphs [13–17]. In this study, we refer to the view of label co-occurrence graphs built using statistical correlations as label global consistency. We identified two issues regarding label global consistency. First, statistical label correlations may exhibit a long-tailed distribution, with some categories being common and most having only a few relevant documents. Figure 1 shows the long-tailed distribution on RCV1 [18], where only a few labels have a large number of articles, and these head labels also have a high co-occurrence with other labels. Second, the co-occurrence patterns between label pairs obtained from the training data are frequently incomplete. For instance, in the AAPD, the labels "computers and society (cs.CY)" and "Physics and Society (physics.soc-ph)" co-occurred 300 times in training set, while only 6 times in test set (0.009%). This imbalance in the co-occurrence frequency of labels within the data as well as between the training and testing sets, led to a highly skewed distribution [15]. Existing methods based solely on label global consistency model label relationships, which are based on prior statistical information from the training data, fail to address above two challenges effectively.

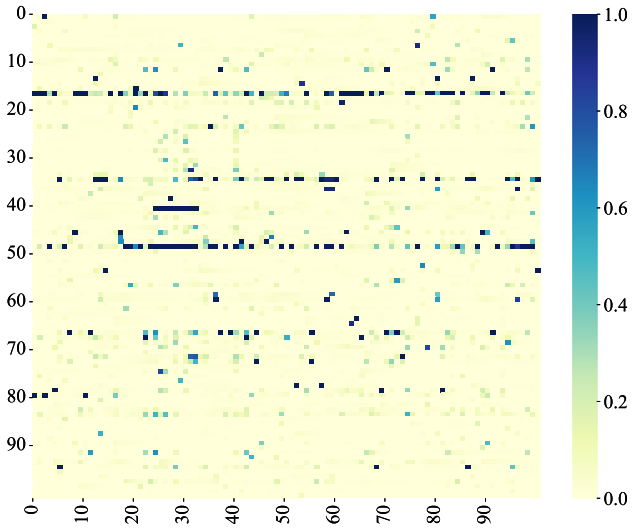


Fig. 1 Long-tailed distribution and label co-occurrence for the RCV1. The co-occurrence matrix undergoes color-coding, wherein the representation is influenced by the conditional probability $p(i|j)$. This probability signifies the likelihood of the presence of a class in the i-th column given the occurrence of a class in the j-th row

To address these challenges, we propose a dual-view graph convolutional network for multi-label text classification (DV-MLTC). The proposed method aims to model label co-occurrences from both global and local perspectives, thereby offering a comprehensive solution. First, to address the long-tailed distribution problem, we introduce a strategy that generates label paths for the local label graph using a random walk. By reconstructing the local label graph based on this strategy, we effectively captured the relationships between low-frequency co-occurring labels. This approach helps alleviate the long-tail distribution issue and enhances the overall performance. Second, to address the highly skewed distribution problem caused by the incompleteness of label co-occurrence patterns in the label co-occurrence graph, we leverage the power of the graph convolutional networks (GCN) [16]. By employing a GCN, we can model rich co-occurrence patterns between labels from both global and local consistency perspectives. Additionally, label local consistency is proposed to measure the rationality of label co-occurrence in local neighborhoods, further improving the accuracy of the model. Furthermore, we incorporated attention flow to extract label-specific semantic components from the document content. This allows us to merge the semantic information of the labels and obtain the initial embedding of the dual-view graph convolution. Finally, we fuse the fine-grained document information with learned label correlations for classification, resulting in a comprehensive and robust classification model.

This paper makes the following contributions:

- We introduced a novel neural network that leverages dual-view convolutions on label co-occurrence graphs for MLTC tasks. Our model combines learned label information from a dual-view graph convolution with label-specific document representations using a dual attention flow. This integration enhanced the overall performance of the model.
- To effectively capture the co-occurrence patterns between labels, we leveraged both global and local label consistencies. Additionally, we employed a dynamic construction approach for the local label graph using a random-walk strategy. This strategy enriches the co-occurrence patterns between labels and significantly improves the performance of multi-label text classification.
- To evaluate the effectiveness of the proposed model, we conducted experiments on three commonly used benchmark datasets. The experimental results demonstrate the competitiveness of our model on these datasets, demonstrating its ability to achieve impressive performance in multi-label text classification tasks.

2 Related work

2.1 Enhancing document-label interaction in MLTC

With the widespread application of neural network methods in document representation, innovative deep-learning approaches have been developed. XML-CNN [4] uses a convolution neural network(CNN) and dynamic pooling to learn text representations for multi-label text classification. A sequence-to-sequence (Seq2Seq) model based on recurrent neuron networks (RNNs) was used to capture the correlations between labels [19–21]. Nevertheless, they treated all words uniformly and failed to discern the informative content within the documents. Considering the negative impact of label sequence order on Seq2Seq models, S2S-LSAM [22] introduces a novel Seq2Seq model with distinct semantic attention mechanisms for labels. This model incorporates label semantics and textual features through the interaction of the label semantic attention mechanism, resulting in fused information comprising both label and textual information. ML-Reasoner [23] utilizes a sequence model as a text feature extractor and incorporates the prediction probabilities from the previous round as an additional input in the model to reflect label correlation. This approach mitigates the reliance on label order. The aforementioned methods do not model the rich co-occurrence relationships among labels. Moreover, these methods struggle to effectively address the long-tail issue associated with labels.

Recently, attention mechanisms have been used in several studies to enhance the interaction between labels and words [24], labels and documents [6, 11, 25–27], and labels and labels [7], in order to learn specific label-specific document representations for classification tasks. Some methods have taken a different approach by incorporating additional sources of knowledge to enhance label-specific document representations [28–30]. These approaches exhibited promising results in MLTC, underscoring the importance of investigating semantic connections. However, they did not thoroughly explore the interactions among label-specific semantic components, which could potentially enhance the prediction of low-frequency labels. In our research, we introduced a label-word attention module and a label-semantic self-attention module. The former extracts important semantics specific to labels from the word-level document information. The latter further helps capture label-level semantic features. Our approach enriches the semantic information of labels by combining these two modules, and this enhanced representation has the potential to improve prediction accuracy, particularly for low-frequency labels.

2.2 Label co-occurrence graph in MLTC

To apprehend profound correlations among labels in a graph structure and delve into the semantic interactions between label-specific components in documents, a common approach involves utilizing label graphs based on statistical co-occurrence. MAGNET [14] constructs a label graph based on frequency. DXML [31] establishes an explicit label co-occurrence graph to explore label embeddings in a low-dimensional latent space. LiGCN [32] utilized a pretrained language model as the initial embedding of a label-word heterogeneous graph and achieved outstanding classification performance while paying attention to different word choices. The methods used by LR-GCN [33] and GCN-MTC [15] are similar. They constructed labeled graphs based on data-driven statistical information, and the former performed better than the latter. LDGN [34] adaptively modeled the interactions among labels using dual-graph convolutional neural networks. CFTC [35] first constructed a global label co-occurrence graph and then prevented confounding shortcuts using counterfactual techniques with the help of a human causal graph. S-GCN [36] leverages text, words, and labels to construct a global heterogeneous graph for mining correlations between similar documents. Subsequently, an encoder is trained to extract semantic features from document nodes, followed by utilizing graph convolutional networks to classify the text nodes. TLC-XML [37] initially constructs a label correlation graph using the semantic information of labels and symmetric conditional probabilities. Subsequently, strongly correlated labels are further grouped into the same cluster. Finally, graph convolutional networks are employed to extract the inter-cluster correlations among the label clusters. Nevertheless, each label is assigned to only one cluster, which severely ignores the semantic correlation of labels.

However, the majority of the above methods primarily focus on the label global consistency of label co-occurrence while neglecting the potential label local consistency, which could potentially enhance classification performance. By contrast, our proposed dual-view convolution module is guided by prior knowledge from co-occurrence statistics and posterior information obtained from a dynamic random walk, which can effectively capture comprehensive interactions from different views, understand the potential relationships between labels through global and local modes in the data, and improve their classification performance.

3 Proposed model

As shown in Fig. 2, our model comprises two primary modules: 1) a label-specific document representation based on dual attention flow. This module outlines the process of extracting label-specific semantic components from the word-level information of each document and further extracting label-specific semantic components. 2) Dual-view graph convolutional networks for semantic interactive learning. We

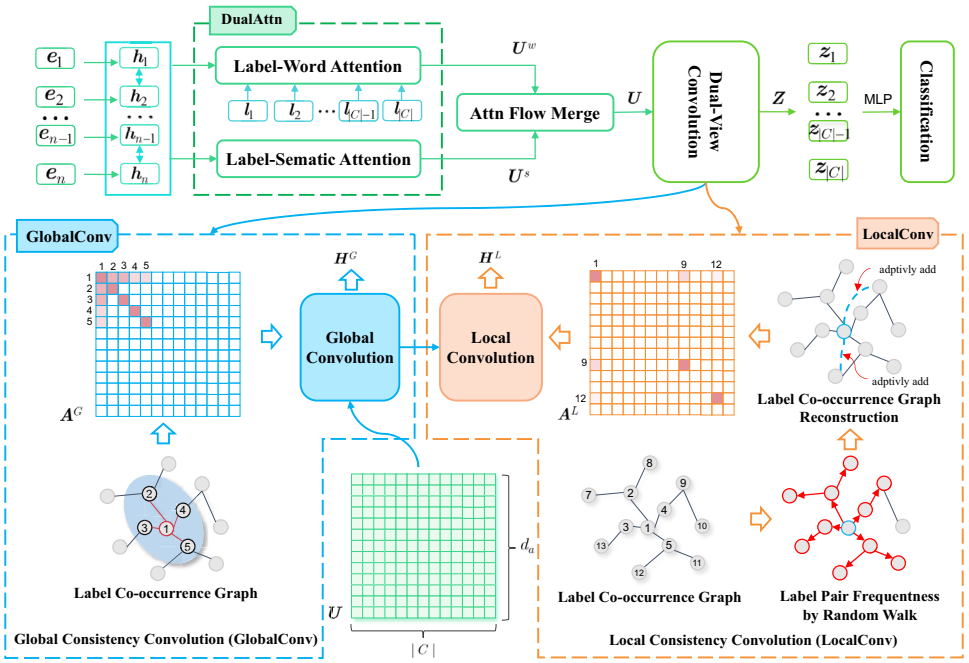


Fig. 2 The model architecture of the DV-MLTC consists of two main components: GlobalConv and LocalConv. In the GlobalConv component, we construct a prior label co-occurrence graph and derive the label co-occurrence matrix A^G. This matrix represents the connections between labels based on their co-occurrence probabilities. Using GlobalConv, we obtain the label embedding matrix H^G under the guidance of the global information (For example, node 1 connects nodes 1,2,3,4,5 by priory probability). In the LocalConv component, we leverage the label co-occurrence graph to compute the local co-occurrence frequency between label pairs using a random walk module. This process involves extending multiple label paths from a starting node. By incorporating this local guidance, we can adaptively add label relationships between pairs that initially had few co-occurrences. This helps mitigate the long-tail distribution problem by reconstructing the label co-occurrence graph. (For instance, in Fig. 2, we observe that node 1 was not originally connected to node 9 and node 12, but the co-occurrence relationship is added through label co-occurrence graph reconstruction.) Finally, the label co-occurrence matrix A^L is passed to the local convolution layer to obtain the matrix H^L

present a detailed description of how this module effectively explores and captures comprehensive interactions from distinct perspectives, guided by prior knowledge of statistical co-occurrences and posterior information obtained from a dynamic random walk. Our dual-view convolution module can effectively explore and capture comprehensive interactions from distinct views guided by prior knowledge of statistical co-occurrences and posterior information obtained from a dynamic random walk.

3.1 Problem definition

In the MLTC problem, we have a document set denoted as $D = \{d_1, d_2, ..., d_{|D|}\}$, and a corresponding label set denoted as $C = \{c_1, c_2, ..., c_{|C|}\}$. Here, $|D|$ represents the number of documents in the document set, and $|C|$ represents the total number of labels. Each document d_i contains n words and is associated with labels $c_i \in C$, where $c_i \in \{0, 1\}^{|C|}$, indicating whether a label is relevant.

To achieve the goal of MLTC, which is assigning the most relevant label to a new document, we define a global label co-occurrences graph $G = (V, E)$ where V represents nodes set and E represents edges set, as in previous work [13, 34, 38]. In this graph, the nodes represent the categories, and the nodes v_i correspond to the labels c_i in the label set C. The edges in the graph represent the statistical co-occurrences between categories. Specifically, we compute the conditional probability for all label pairs in the training set, yielding the global label co-occurrence matrix $A^G \in R^{|C|\times|C|}$ Here, $A^G_{(i,j)} = p(v_j|v_i)$ signifies the conditional probability of a document being categorized as c_j when it belongs to category c_i. Notably G is a directed graph, therefore, $A_{(i,j)}$ may not be equal to $A_{(j,i)}$ owning the conditional probability calculations.

3.2 Label-specific attention networks

Given a document D containing n words, we utilized bidirectional long short-term memory (BiLSTM) to encode word-level semantic information in the document representation. BiLSTM leverages its bidirectional nature to effectively capture contextual information by processing word sequences in both forward and backward directions. This enables a thorough understanding of the document's semantic context.

Upon applying BiLSTM, we obtained two sets of hidden states: forward and backward. These hidden states encapsulate the contextual information of the words within a document. To create a comprehensive word sequence representation, we concatenated the forward and backward hidden states, resulting in the matrix $H \in \mathbb{R}^{n \times d_a}$. d_a denotes the dimensions of the word vectors. By concatenating the forward and backward hidden states, semantic information can be captured in both directions, thereby creating a robust and holistic representation of the word sequence within the document.

3.2.1 Label-word attention

Labels possess distinctive semantics in the context of text classification, concealed within their textual representations or descriptions. To capitalize on this semantic information, labels undergo preprocessing and are symbolized as trainable matrices $L \in R^{|C| \times d_a}$ in the same latent d_a-dimensional space as words. To ascertain determine the semantic relationship between each pair of words and labels, scaled dot-product attention is employed:

$$U^w = softmax(\frac{LH^T}{\sqrt{d_a}})H \tag{1}$$

where L is the query vector, H is the key vector and the value vector. u_i is the i-th row vector of $U^w \in R^{|C| \times d_a}$, denoting the semantic component in the document associated with the label c_i. This representation is based on labeled text, which can be called the Label-Word (LW) attention mechanism.

3.2.2 Label-semantic self-attention

Multiple labels may be assigned to labeled documents, and each document should encompass the contexts most relevant to its corresponding labels. Consequently, each document may comprise multiple components, and the words within a document may contribute differently to each label. To capture these distinct components of each label, a self-attention mechanism is employed. The label-semantic (LS) self-attention score ($Q \in R^{|C| \times n}$) is calculated as follows:

$$\begin{aligned} Q &= softmax(W_2 tanh(W_1 H^T)) \\ U^s &= Q \times H \end{aligned} \tag{2}$$

where $W_1 \in R^{d_b \times d_a}$ and $W_2 \in R^{|C| \times d_b}$ are self-attention parameters that must be trained. d_b is a hyperparameter.

Label-specific semantic components are extracted from text content using a novel approach that incorporates both label-word attention U^w and label-semantic self-attention U^s. By combining these attention flows, we obtain the label-specific document representation $U = U^w + U^s$, which is calculated as the sum of U^w and U^s. Our approach draws inspiration from previous works, such as [25] and [39], which also utilized attention mechanisms. However, the dual-attention flow module distinguishes itself based on two key aspects. First, we focused on the interaction between documents and labels, enabling a more targeted exploration of their relationships. Second, our calculation method is designed to be more straightforward and efficient while still delivering superior performance.

The resulting label-specific document representation U serves as the input for the subsequent module: the dual-view convolutional networks. These networks further process and capture the interactions between the extracted semantic components.

3.3 Dual-view graph convolutional networks

To capture the interactions between label-specific semantic components from multiple perspectives, we employed a dual-view interaction approach. Specifically, we utilize global and local consistency convolutions. In the global consistency convolution, we construct a global label co-occurrence graph and apply GCN to achieve global consistency. This convolution leverages the co-occurrence patterns between labels captured by the global label co-occurrence graph. In the local consistency convolution, we generated a local label co-occurrence graph using a random walk strategy. Subsequently, we employed a GCN to perform local consistency convolution. This convolution focuses on enhancing the co-occurrence patterns between labels based on the local context captured by the local label co-occurrence graph. These convolutions consider distinct interaction views, thereby enhancing the co-occurrence patterns between labels.

3.3.1 GlobalConv

To establish deep relationships between label-specific semantic components guided by statistical label correlations, we employ a global consistency convolution (GlobalConv). We leverage a GCN layer to propagate messages between neighboring labeled nodes, thereby enhancing their representation of these labeled nodes. The layer-by-layer propagation rules are defined as follows:

$$H^G = \sigma(D_1^{-(1/2)} \hat{A}^G D_1^{-(1/2)} U W^G) \tag{3}$$

where A^G in (3) is the global label co-occurrence graph. $\sigma(\cdot)$ represents the LeakyReLU activation function. $\hat{A}^G$ represents the normalized adjacency matrix of A^G. D_1 is the degree matrix of A^G and $W^G \in \mathbb{R}^{d_a \times d_c}$ denotes the transformation matrix that must be learned. GlobalConv uses the initialized components $U \in \mathbb{R}^{|C| \times d_a}$ and A^G as inputs and

ultimately generates $H^G \in \mathbb{R}^{|C|\times d_c}$, where d_c denotes the dimensionality of the final node representation.

GlobalConv primarily performs a 1-hop diffusion process in each layer by leveraging prior statistical relationships present in the dataset. As described in a previous study [40], this process only considers the addition of feature vectors from neighboring nodes to account for the feature relationships between them. However, the statistical label correlations obtained from training data can be incomplete and noisy, and the co-occurrence patterns between label pairs may suffer from long-tailed distributions [15]. Recognizing this limitation motivated us to assign a certain probability to low-frequency co-occurring labels, indicating that they might belong to the same text rather than being directly filtered as noise. We enabled the model to learn more effective propagation and richer co-occurrence patterns by introducing local consistency convolution.

3.3.2 LocalConv

In addition to the graph structure information defined by the adjacency matrix A^G, we utilized positive pointwise mutual information (PPMI) to encode the potential relationship between label pairs. First, we calculated the frequency matrix F using a random walk. Subsequently, we derived the local graph label co-occurrence graph $A^L \in \mathbb{R}^{|C|\times|C|}$ based on F. Finally, we performed a local consistency convolution.

A random walk can be characterized as a Markov chain that delineates the sequence of nodes visited by a random walker [40]. We define a state as $s(m) = v_i$ if a random walker is on node v_i at time m. The transition probability of moving from the current node v_i to one of its neighbors v_j is denoted as $p(s(m+1) = v_j|s(m) = v_i)$. In our problem setting, given a prior label co-occurrence matrix A^L, we assign:

$$p(s(m+1) = v_j|s(m) = v_i) = \frac{A^G_{i,j}}{\sum_j A^G_{i,j}} \tag{4}$$

This assignment ensures that the transition probability is proportional to the label co-occurrence in A^L, thereby incorporating semantic information into the random walk process.

Algorithm 1 outlines the calculation of the frequency matrix F using random walk. This algorithm can be parallelized by simultaneously performing multiple random walks on different parts of a graph.

Following the computation of the frequency matrix F, the i-th row in F corresponds to the row vector $F_{i,:}$, while the j-th column in F corresponds to the column vector $F_{:,j}$. Specifically, $F_{i,:}$ represents the path node context for node v_i, and $F_{:,j}$ represents the path neighbor node context j. Moreover, $F_{i,j}$ denotes the number of co-occurrences of v_i and v_j in all generated paths. A higher value of $F_{i,j}$ indicates a greater frequency of co-occurrence between the two nodes.

Algorithm 1 Calculation method of frequency matrix F.

Require: global label co-occurrence matrix A^G, path length q, number of iteration t
Ensure: frequency matrix F
Initialize matrix F as 0 matrix
for each label node v_i **do**
  Set v_i as the starting point of the path for the random walk
  for 1 to t **do**
    Generate path $S = RandomWalk(A^G, v_i, q)$
    Uniformly sample non-repeating label pairs (v_i, v_j) from S
    for (v_i, v_j) **do**
      $F_{i,j} += 1$; $F_{j,i} += 1$
    end for
  end for
end for

Using the frequency matrix F, we transform it into a PPMI matrix as follows:

$$\begin{aligned} p_{i,j} &= \frac{F_{i,j}}{\sum_{i,j} F_{i,j}} \\ p_{i,*} &= \frac{\sum_j F_{i,j}}{\sum_{i,j} F_{i,j}} \\ p_{*,j} &= \frac{\sum_i F_{i,j}}{\sum_{i,j} F_{i,j}} \end{aligned} \tag{5}$$

We apply (6) to encode the potential relationship between label pairs in F. Here, $p_{i,j}$ represents the estimated probability of node v_i appearing in context context$_j$; $p_{i,}$ denotes the estimated probability of node v_i, and $p_{,j}$ indicates the estimated probability of the context context$_j$. The adjacency matrix based on the label local consistency is computed as follows:

$$A^L_{i,j} = \max\{PMI_{i,j} = \log(\frac{p_{i,j}}{p_{i,*}p_{*,j}}), 0\} \tag{6}$$

where $PMI_{i,j}$ is the pointwise mutual information between node v_i and context context$j$. The PPMI matrix A^L represents the adjacency matrix based on label local consistency, where any negative PMI value is set to zero.

Similar to GlobalConv, we defined an independent single-layer GCN for LocalConv based on A^L. The graph convolutional networks is given by:

$$H^L = \sigma(D_2^{-(1/2)}\hat{A}^L D_2^{-(1/2)} H^G W^L) \tag{7}$$

where $\hat{A}^L$ denotes the normalized label local consistency matrix, D_2 is the degree matrix of A^L, and $W^L \in \mathbb{R}^{d_c\times d_c}$ is a training parameter. Notably, the dynamically reconstructed A^G based on random walk ensures label local consistency, where labels that appear on the same path are reasonably considered to belong to the same text. In addition, as the path length increases within a reasonable range, the importance of the labels becomes more prominent. Moreover, the

non-positive values in the PPMI matrix were automatically filtered out, preventing low-frequency co-occurrence labels such as noise from disturbing the model.

Both H^G and H^L represent graph convolution-based label representations, with the former focusing on the similarity of global labels and the latter emphasizing the co-occurrence plausibility from local perspectives. These representations had different training parameters. In this task, concatenation is employed to integrate them.

$$Z = H^L || H^G \tag{8}$$

The label-specific document representation generated under the guidance of global and local consistency can be described as matrix $Z \in \mathbb{R}^{|C| \times 2d_c}$. We then make label predictions using a trainable linear layer followed by a sigmoid activation function:

$$\hat{Y} = sigmoid(W_3 Z + b_2) \tag{9}$$

where W_3 represents the weights of the linear layer and b_2 is the bias. Let $y \in \mathbb{R}^{|C|}$ denote the true label of a document, where $y_i \in \{0, 1\}^{|C|}$ indicates whether label i is present in the document. The proposed model was trained using multi-label cross-entropy loss as follows:

$$L = \sum_{i=1}^{N} \sum_{j=1}^{C} y_{ij} \log(y_{ij}) + (1 - y_{ij}) \log(1 - \hat{y}_{ij}) \tag{10}$$

In (10), N represents the number of documents, C represents the number of labels, and y_{ij} and $\hat{y}_{ij}$ denote the true and predicted values, respectively, for the j-th label of the i-th document.

4 Experiment

4.1 Datasets and evaluation metrics

We evaluate the proposed model on three benchmark multi-label text classification datasets:

RCV1[1]: RCV1 [18] was collected and manually classified by Reuters, which collected more than 80k news texts and corresponding multiple labels from 1996 to 1997. Moreover, the testing set consisted of a significantly larger number of examples than the training set. This aspect allowed for a comprehensive evaluation of the generalization capability of the proposed model.

[1] http://www.ai.mit.edu/projects/jmlr/papers/volume5/lewis04a/lyrl2004_rcv1v2_README.htm

Table 1 Statistics of the datasets

Datasets	N	M	D	L	$\overline{L}$	$\tilde{L}$
RCV1	804,414	23,149	781,265	101	3.18	729.67
AAPD	55,840	54,840	1000	54	2.41	2444.04
EUR-Lex	171,120	11,585	3,865	3,956	5.32	15.59

where N represents the total number of documents, M is the number of training documents, D is the number of testing documents, L is the number of class labels, $\overline{L}$ is the average number of labels per document, and $\tilde{L}$ is the average number of documents per label

AAPD[2]: AAPD [19] was constructed by gathering the abstracts and their corresponding subjects from a computer science academic website encompassing 55,840 papers.

EUR-Lex[3]: EUR-Lex [41] is an extreme multi-label text classification dataset comprising documents related to European Union law across 3956 subjects. The public version includes 11585 instances for training and 3865 instances for testing.

These datasets were meticulously chosen due to their widespread usage and large scale, allowing us to validate the efficiency of the proposed model. Additionally, to maintain consistency with prior research, we employed the same dataset partitioning as those in earlier studies [25, 34]. These partitions were the original ones provided by the publishers of the datasets. Detailed statistics for the datasets are presented in Table 1.

Following the established conventions of previous studies [24, 25, 33, 34], we employed the accuracy of the top k (P@k) and the normalized discounted cumulative gain of the top k (nDCG@k) as performance evaluation metrics for all three datasets.

The word embeddings in our model were initialized with 300-dimensional GloVe [42] word vectors that were trained on the dataset using the Skip-gram [43] algorithm. The hidden sizes of the Bi-LSTM and GCN layers were set to 300 and 512, respectively. For the AAPD, we established $q = 2$ and $t = 400$. We determined that $q = 3$ and $t = 450$ for RCV1. Finally, for the EUR-Lex, we set $t = 3$ and $t = 600$. We employed the Adam optimization method to minimize cross-entropy loss. The learning rate was initialized to 1e-3, and a cosine-annealing algorithm was applied to gradually reduce the learning rate during training. To ensure a fair comparison with related baselines using the large language model (LLM), we also implemented an LLM-based version of our model. In this version, we used the word sequence token RoBERTa [44] as the output of the label-specific attention

[2] https://git.uwaterloo.ca/jimmylin/Castor-data/tree/master/datasets/AAPD/

[3] http://nlp.cs.aueb.gr/software.html

network module in our model. The model was trained for 15 epochs with a batch size of 64. The best parameter configuration was selected based on the performance of the validation set and evaluate using a testing set.

4.2 Baselines

To demonstrate the efficiency of the proposed model, it was compared with models that achieved state-of-the-art results using selected datasets. For a fair comparison, we only reused the experimental results when selecting baselines instead of reimplementing them to maintain the recommended optimal settings and results. In addition, for models that were not implemented on specific datasets, we reimplemented these models with their source codes and then evaluated them on selected datasets.

Enhancing document-label-based methods

- **XML-CNN** [19]: A sequence generative model that labels correlations as an ordered sequence.
- **AttentionXML** [24]: A model that constructs the label-aware document representation solely based on the document content.
- **LSAN** [25]: Label-aware attention framework based on self-attention and label attention mechanisms.
- **HTTN** [7]: This proposes a head-to-tail network that transfers meta-knowledge from head-labels to tail-labels.
- **MLGN** [26]: A multi-label guided network capable of guiding document representation with multi-label semantic information.

Label graph-based methods

- **DXML** [31]: A deep embedding method that simultaneously models the feature and label space.
- **MAGNET** [14]: A model based on graph attention networks. Capturing the attention-dependent structure between labels using features and correlation matrices was proposed. In addition, the model uses BiLSTM to extract text features.
- **LAHA** [6]: LAHA focuses on using hybrid attention to represent documents with labels. The model comprises three components: a multi-label self-attention mechanism that identifies each word's association with labels, a depiction of label arrangement and document context, and an adaptive fusion method for classification.
- **LDGN** [34]: A dual-graph convolution network that incorporates category information and models adaptive interactions among labels in a reconstructed graph.
- **LiGCN** [32]: A label interpretable graph model that solves the MLTC problem by modeling tokens and labels as nodes in a heterogeneous graph and uses the pretrained language model BERT as a text encoder.
- **LA-MLTC** [39]: A label-aware network built (which we refer to as LA-MLTC) a heterogeneous graph including words and labels to learn the label representation and text representation by metapath2vec.

Table 2 Comparing our model with baselines in terms of P@K and nDCG@k on RCV1

Methods	P@1(%)	P@3(%)	P@5(%)	nDCG@3(%)	nDCG@5(%)
XML-CNN[a] (2018)	95.75	78.63	54.94	89.89	90.77
AttentionXML[a] (2019)	96.41	80.91	56.38	91.88	92.70
LSAN[a] (2019)	96.81	81.89	56.92	92.83	93.43
HTTN[b] (2021)	95.86	78.92	55.27	89.61	90.86
MLGN (2023)	96.67	82.11	57.03	92.23	93.55
DXML[a] (2018)	94.04	78.65	54.38	89.83	90.21
MAGNET (2020)	95.16	79.34	54.26	87.34	88.61
LAHA[c] (2018)	96.95	81.43	56.44	92.69	93.01
LiGCN[d] (2022)	95.61	82.40	56.31	93.40	93.26
LA-MLTC[e] (2021)	<u>97.31</u>	83.11	57.85	93.97	94.59
LDGN[a] (2021)	97.12	82.26	57.29	93.80	95.03
LR-GCN[f] (2023)	97.13	<u>84.29</u>	<u>58.45</u>	**94.98**	<u>95.38</u>
DV-MLTC	97.11	83.68	57.31	94.19	94.77
DV-MLTC$_{RoBERTa}$	**97.94**	**84.83**	**59.01**	<u>94.32</u>	**95.87**

The best performance is highlighted in bold, and the second-best performance is highlighted in underlined text. The following experimental results were extracted: [a] from [34], [b] from [7], [c] from [6], [d] from [32], [e] from [39], and [f] from [33]

Table 3 Comparing our model with baselines in terms of P@K and nDCG@k on AAPD

Methods	P@1(%)	P@3(%)	P@5(%)	nDCG@3(%)	nDCG@5(%)
XML-CNN[a] (2018)	74.38	53.84	37.79	71.12	75.93
AttentionXML[a] (2019)	83.02	58.72	40.56	78.01	82.31
LSAN[a] (2019)	85.28	61.12	41.84	80.84	84.78
HTTN[b] (2021)	83.84	59.92	40.79	79.27	82.67
MLGN (2023)	84.78	60.01	42.37	80.11	83.45
DXML[a] (2018)	80.54	56.30	39.16	77.23	80.99
MAGNET (2020)	82.53	60.71	40.19	80.37	81.03
LAHA[c] (2018)	84.48	60.72	41.19	80.11	83.70
LiGCN[d] (2022)	82.50	61.26	41.38	80.39	83.83
LA-MLTC[e] (2021)	85.03	61.46	41.80	80.94	84.90
LDGN[a] (2021)	86.24	61.95	<u>42.29</u>	<u>83.32</u>	<u>86.85</u>
LR-GCN[f] (2023)	<u>86.50</u>	<u>62.43</u>	41.66	82.52	85.48
DV-MLTC	85.19	61.52	40.06	83.21	85.15
DV-MLTC$_{RoBERTa}$	**86.83**	**62.87**	**42.41**	**83.45**	**87.03**

The best performance is highlighted in bold, and the second-best performance is highlighted in underlined text. The following experimental results were extracted: [a] from [34], [b] from [7], [c] from [6], [d] from [32], [e] from [39], and [f] from [33]

- **LR-GCN** [33]: A multi-label text classification model combining a pre-trained language model and a GCN.

4.3 Performance comparison of different methods

The performances of the different models on the three datasets are listed in Tables 2, 3, and 4 in terms of P@k and nDCG@k, respectively. For each row, the best result is highlighted in bold, and the second-best result is underlined.

As shown in Tables 2, 3, and 4, the proposed DV-MLTC model outperforms previous works on all three datasets. Specifically, the DV-MLTC-enhanced version of Roberta achieves better or more competitive performance on all metrics and significantly improves the previous baseline best scores compared to those with the shared source code. For example, on EUR-Lex, DV-MLTC improves P@1 and nDCG@3 from 82.59% to 83.61% and from 72.15% to 74.62%, respectively. Compared with the best baseline LR-GCN on RCV1 and AAPD, our proposed model still performs better or is competitive on all metrics.

Furthermore, by observing the results in Tables 2, 3, and 4, we can see that methods that do not incorporate label correlation to improve the learning process of textual representations demonstrate inferior performance. Specifically, on AAPD, AttentionXML elevated the P@1 value of DXML from 80.54% to 83.02%, marking an increase of approximately 3.08%. It is plausible that while DXML seeks to represent information in the label space using deep embedding techniqus, AttentionXML can concentrate on the more semantically relevant document sections for each label. Nev-

Table 4 Comparing our model with baselines in terms of P@K and nDCG@k on EUR-Lex

Methods	P@1(%)	P@3(%)	P@5(%)	nDCG@3(%)	nDCG@5(%)
XML-CNN[a] (2018)	70.40	54.98	44.86	58.62	53.10
AttentionXML[a] (2019)	67.34	52.52	47.72	56.21	50.78
LSAN[a] (2019)	79.17	64.99	53.67	68.32	62.47
HTTN[b] (2021)	81.14	67.62	56.38	70.89	64.42
MLGN (2023)	68.65	53.17	48.92	57.34	51.28
DXML[a] (2018)	75.63	60.13	48.65	63.96	53.60
LAHA[c] (2018)	78.34	64.62	53.08	68.15	62.27
LDGN[d] (2021)	81.03	67.79	56.36	71.81	66.09
LR-GCN (2023)	<u>82.59</u>	<u>68.25</u>	<u>58.34</u>	<u>72.15</u>	<u>66.87</u>
DV-MLTC	81.01	66.98	56.73	71.02	66.14
DV-MLTC$_{RoBERTa}$	**83.61**	**70.14**	**59.40**	**74.62**	**68.11**

The best performance is highlighted in bold, and the second-best performance is highlighted in underlined text. The following experimental results were extracted: [a] from [34], [b] from [7], [c] from [6], and [d] from [32]

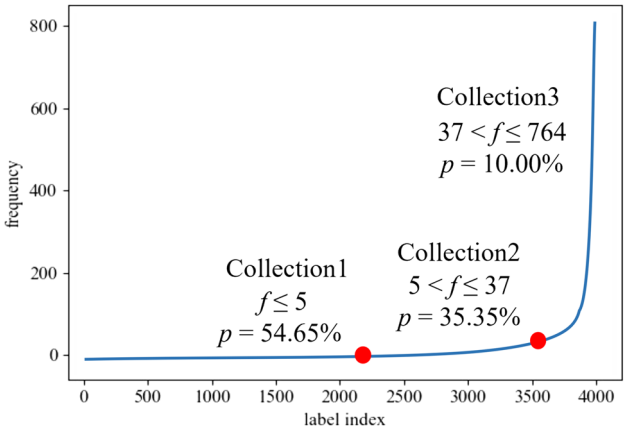


Fig. 3 The label distribution of EUR-Lex. x-axis represents the label index sorted by frequency in the training set. f represents the label frequency, and p represents the proportion of labels in collection to the entire label set

ertheless, AttentionXML solely focuses on encoding text content in the presentation layer without considering label information, thus restricting its capacity to adjust contextual representations through interactions.

The better performance of LSAN compared to other previous approaches for exploring document-label relationships, such as HTTN and MLGN, may be attributed to its multi-view learning space mechanism and the fact that LSAN considers semantic correlations between text and labels simultaneously. The multi-view learning mechanism helps stabilize adaptive fusion through the attention mechanism, which learns the text representation specific to the labels.

We observed that LR-GCN performed best on RCV1 in terms of the nDCG@3. This can be explained by initializing text embedding using the pretrained language model Roberta, which can efficiently extract fine-grained document information. In contrast, our model uses a simple BiLSTM architecture to represent the input text and achieves optimal or near-optimal results. In addition, we used Roberta's version of word embedding to obtain the same word embeddings as the LR-GCN. The results of AAPD and EUR-Lex demonstrate the effectiveness of our dual-view graph convolutional networks module, with DV-MLTC$_{RoBERTa}$ achieving the best results compared to the competing models.

LDGN [34] demonstrated competitive performance on all datasets, which may be attributed to its adaptive interaction component, benefiting from a large number of adaptive parameters. Inspired by LDGN, we propose an adaptive reconstruction of the graph based on random wandering. However, the LDGN adaptive module operates as a black box, and its parameter guidance lacks explicit transparency. By contrast, our dual-graph module allows parameter sharing and provides natural interpretability. This allowed us to conduct further research on our model, particularly on parameter tuning and its implications.

We also observed that the methods that utilized labeled graphs outperformed the document-label based methods overall, which highlights the advantage of MLTC methods with graphs, as most of them incorporate rich interaction information to improve multilabel text prediction. The exception is the LAHA based on simple label co-occurrence, which we hypothesize captures only the representation of labels from the label co-occurrence graph without further exploring the deep relationships between labels.

4.4 Comparison on sparse dataset

To evaluate the performance of DV-MLTC on long-tailed labels, we categorized the labels in EUR-Lex into three groups based on their frequency of occurrence, following the approach in [6, 25]. Figure 3 illustrates the distribution of label frequencies on EUR-Lex, where f represents the label frequency. Approximately 55% of the labels appeared one to five times, constituting the first label collection (Collection1). The labels that appeared 5-37 times were assigned to Collection2, accounting for 35.35% of the entire label set. The remaining 10% of frequent labels formed the final collection (Collection3). Clearly, Collection 1 presents greater difficulty compared to the other two collections due to the lack of training data. Obviously, Collection 1 is much more difficult than the other two collections owing to the lack of training data.

Figure 4 shows the prediction results in terms of P@1, P@3, and P@5 obtained using AttentionXML, DXML, and DV-MLTC, respectively. The three methods improved from Collection1 to Collection3, which is reasonable because each label contained an increasing number of documents

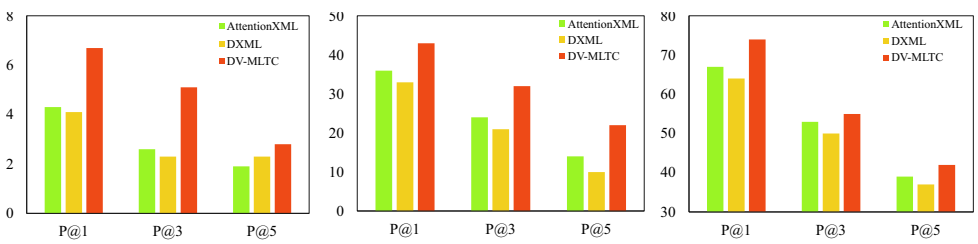


Fig. 4 AttentionXML, DXML and DV-MTC for three collections on EUR-Lex in terms of P@k

from Collection1 to Collection3. DV-MLTC significantly improves the predictive performance of Collection1. Particularly, DV-MLTC achieved an average gain of over 55.83%, 96.22%, and 47.36% for AttentionXML on the three metrics of Collection1, and 63.41%, 121.73%, and 44.37% for DXML, respectively. This result demonstrates the superiority of the proposed model for multi-label text data with tail labels.

4.5 Ablation experiments

A series of ablation experiments was performed to assess the importance and necessity of each module. We performed ablation experiments on all three datasets and divided the experiments into two groups: **Group1** and **Group2**.

Group1 experiments focus on the modules related to dual-graph convolution. The ablation components tested in this group were as follows:

1. $w/o\ LW$: our model without LW attention
2. $w/o\ LS$: our model without LS attention
3. $w/o\ dual\ attn$: our model without dual attention

Group2 experiments focus on modules related to dual attention. The ablation modules tested in this group were as follows:

1. $w/o\ GlobalConv$: our model without GlobalConv
2. $w/o\ LocalConv$: our model without LocalConv
3. $w/o\ DualConv$: our model without dual-attention
4. $sharing\ para$: GlobalConv and LocalConv share the parameters of the GCN layer

From the results shown in Fig. 5 of the ablation experiments conducted on AAPD and RCV1, several observations about **Group1** can be made: Dual Attention Flow Module: $w/o\ LW$ and $w/o\ LS$ outperformed $w/odualattn$, with large margins of 3.03% and 2.21% on AAPD, indicating that both attention flows enhance the model and are indispensable. In other words, both the label-word attention and label-semantic self-attention modules contribute to the performance of proposed model. Label attention considers the interaction between the label and word information and captures the contribution of words to labels. Self-attention, on the other hand, focuses on the semantic information of the labels themselves.

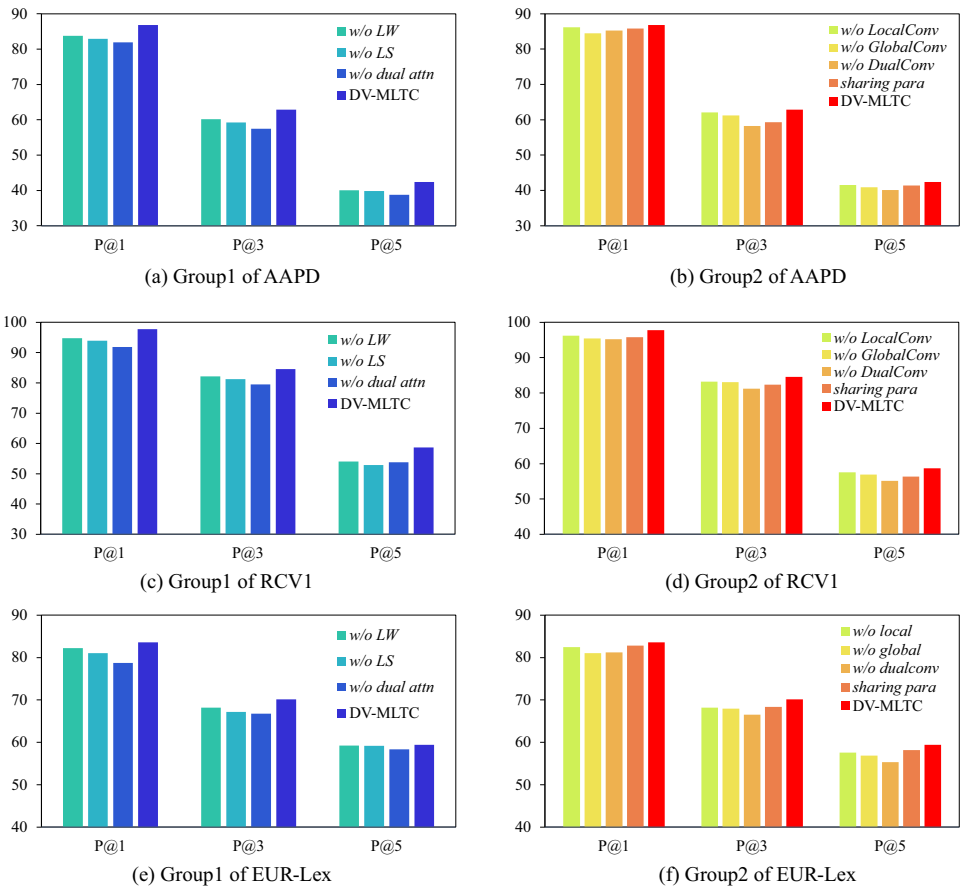


Fig. 5 Ablation Experiment. (a) and (b) are the experimental results of Groups 1 and 2, respectively, for AAPD. (c) and (d) are the experimental results of Groups 1 and 2, respectively, for RCV1. (e) and (f) are the experimental results of Groups 1 and 2, respectively, for EUR-Lex

Conclusions about **Group2**: (1) Dual-View Convolutional Modules: The experiments $w/o\ LocalConv$ and $w/o\ GlobalConv$ outperform $w/o\ DualConv$, such as on RCV1, with better results of 1.98% and 1.82% on P@3. This indicates that exploring either global label or local consistency can effectively capture the semantic interactions between label-specific components. The superiority of $w/o\ LocalConv$ over $w/o\ GlobalConv$ suggests that models with global consistency convolution have a significant impact on classification improvement, indicating their ability to capture semantic dependencies effectively. (2) $w/o\ GlobalConv$ improves $w/o\ DualConv$: The experiment $w/o\ GlobalConv$ improves the performance of the model based on the dual attention flow, indicating that incorporating the new label co-occurrence relationship generated through a random walk and mutual information can benefit the model's performance. (3) $Sharing\ Parameters$: The experiment involving parameter sharing between the global convolution and local convolution shows slightly lower performance compared to the complete model. This suggests that the two sets of GCNs, which model label correlation from different perspectives and interactions, benefit from separate parameter operations rather than sharing. (4) Overall Model: The complete model, which combines dual attention flow and dual-view convolutions while separating the parameters, achieves the best performance. These results demonstrate the efficacy of the suggested modules and their contributions to the overall performance of the model in capturing label dependencies and semantic interactions.

We visualized the label co-occurrence graph matrices A^G and A^L on the AAPD, as shown in Fig. 6. From the visualization, we can observe that the global label co-occurrence graph matrix A^G exhibits a long-tail distribution, where there are many edges with very few co-occurrences. This distribution was based on prior statistics from a corpus. However, these low-frequency edges may be considered noise data, and they can lead to overfitting and negatively affect classification performance. The variant without the A^L matrix ($w/o\ LocalConv$) did not perform optimally. This is because A^G alone, which builds a co-occurrence graph based on statistical co-occurrence, cannot provide sufficient semantic confidence between the label pairs. The dynamic edge adjustment performed by A^L through a random walk leads to a softer performance in the visualization graph. It assigns a certain edge weight to low-frequency co-occurring label pairs, thereby allowing them to overcome the influence of low-frequency noise. This adjustment is beneficial for the GCN because it strengthens the interactions between node pairs in the network. As for A^L, the A^L with an iteration number of 1000 tends to exhibit more smoothness compared to the A^L with an iteration number of 450. Over-smoothing makes it difficult to distinguish the differences in label co-occurrence, potentially degrading the classification performance. Our proposed model integrates A^G and A^L using GlobalConv and LocalConv, respectively, and leverages both statistical co-occurrence information and dynamic edge adjustment based on random walks, leading to improved classification results.

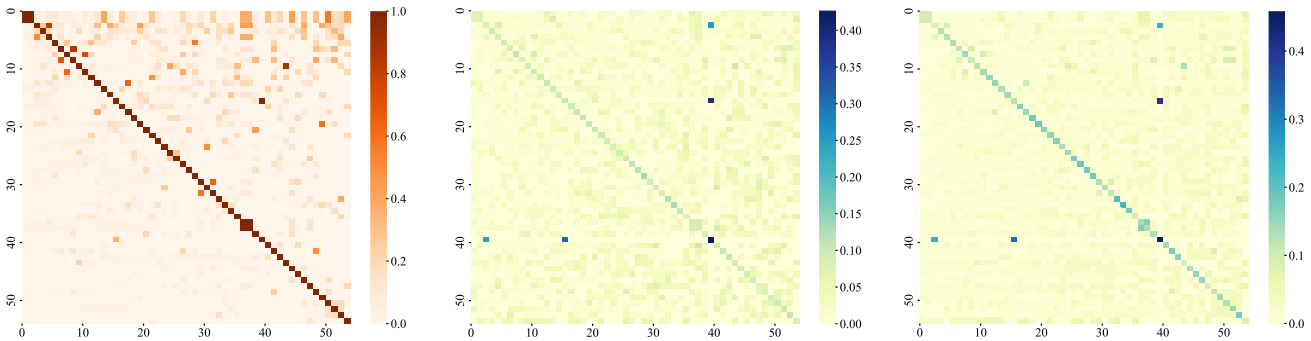
Overall, the visualization of the weight matrices confirmed the effectiveness of incorporating both A^G and A^L in capturing label dependencies and enhancing the performance of the classification model.

4.6 Parametric analysis

We performed relevant experiments on our model using the AAPD. We used the base version of the model in the parametric analysis.

4.6.1 Effect of iteration number *t* on classification

We investigated the effect of the number of iterations, denoted as t, on the classification performance. The number of iterations determined the number of label paths generated by node resampling. By controlling the other parameters and varying the value of t, the impact on the classification per-



(a) A^G (b) A^L ($t = 450$) (c) A^L ($t = 1000$)

Fig. 6 Label graph visualization on AAPD. From left to right: global consistency label graph; local label co-occurrence graph (t=450); local label co-occurrence graph (t=1000)

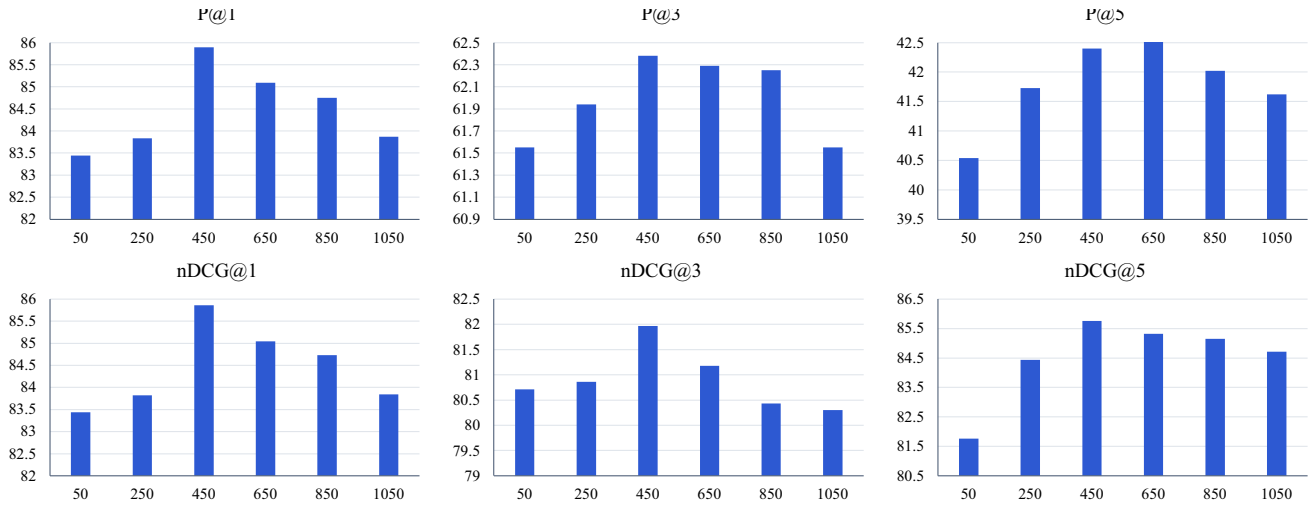


Fig. 7 Test performance (%) under varying t on AAPD

formance was analyzed, as shown in Fig. 7. The experimental results show that when the number of iterations was small, the performance improvement of the model was insignificant. This is because the local label graph that captures local label dependencies fails to effectively capture the key and tail labels. Consequently, the role of all the local label graphs becomes similar to that of a global graph, leading to limited performance gains. As the number of iterations increased, specifically reaching a certain scale (e.g., 450), the local dynamics strongly enhanced the interaction between the key and tail graphs. By leveraging the powerful information diffusion ability of the GCN, the model achieved improved classification performance. This indicates that a sufficient number of iterations allows the local dynamics to capture crucial graph dependencies, resulting in enhanced classification accuracy.

Increasing the number of iterations beyond the optimal value did not significantly affect the model's performance. This suggests that once the key and tail graph nodes are effectively captured and the interaction between graphs is strengthened, further increasing the number of iterations has little effect on the model. In summary, the experimental results demonstrate that the number of iterations, t, plays a crucial role in capturing graph dependencies through local dynamics. Finding the optimal value of t allows the model to effectively enhance graph interactions and improve classification performance.

4.6.2 Effect of path length q on classification

The path length parameter q plays a crucial role in the classification performance of our model, particularly in the LocalConv module. It determines the farthest distance that the random walk can traverse based on probability, with labels on the same path considered to belong to the same document. In our experiments on AAPD, we investigated the impact of q on the classification accuracy while maintaining t at an optimal value of 450. The results shown in Fig. 8 indicate that the choice of q significantly affects the performance of the model, which is consistent with our expectations. Within a reasonable range (e.g., 2 or 3), the model achieved the best classification results, suggesting that the label paths adaptively generated by the model have significant benefits. However, when q exceeds a certain threshold (e.g., 3), the performance of the model begins to decline slightly. We speculate that excessively long label paths result in excessively consistent co-occurrence relationships between nodes during the iterative process. This exacerbates the problem of over-smoothing, ultimately interfering with the discriminative power of the labels in the model. Nevertheless, by integrating the LocalConv and GlobalConv modules, our model maintains its robustness and achieves optimal performance. This highlights the effectiveness and resilience of our approach in capturing label dependencies and enhancing the classification outcomes.

4.6.3 Effect of label-ratio

To assess the sensitivity and performance of the proposed model under different training data proportions, we conducted experiments using various ratios of training data. We also compared our model with other competitive approaches, namely XML-CNN [4], AttentionXML [24], LSAN [25], and LR-GCN [33], while maintaining their respective settings, as described in their papers. In the case of LSAN, we utilized Word2vec for word embeddings because of the absence of pretrained embeddings in its source code. Figure 9 shows the evaluation results for different data scales

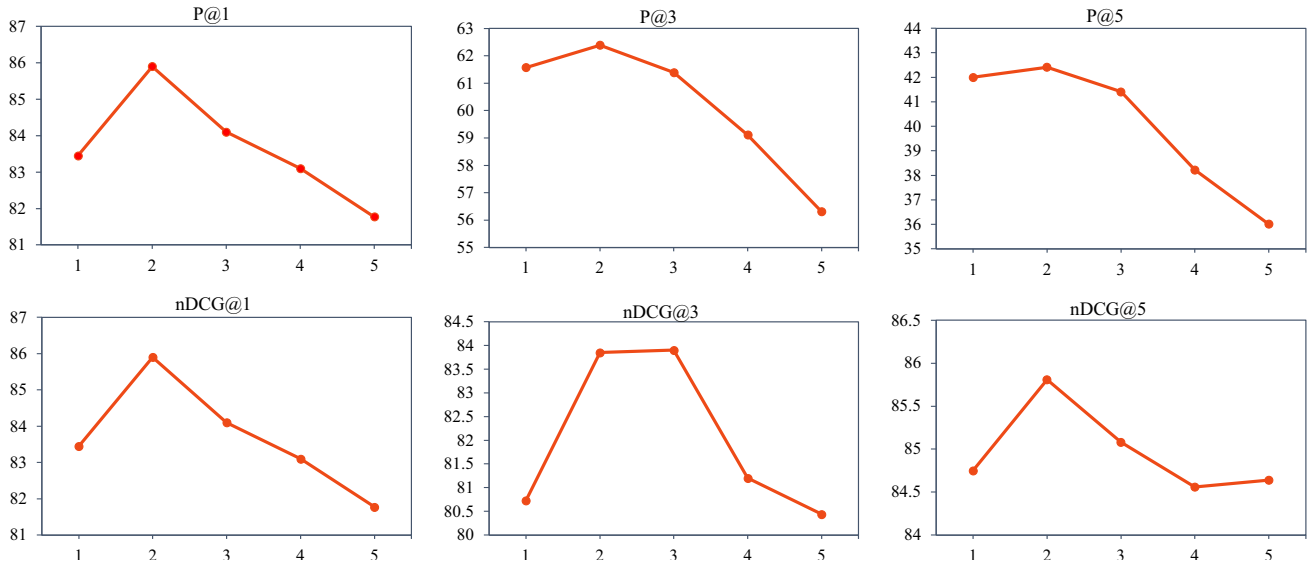


Fig. 8 Test performance (%) under varying q on AAPD

with proportions of 0.05, 0.10, 0.25, 0.50, and 0.75. It is evident from the results that our model consistently achieves competitive performance compared to the baselines. Notably, our model outperformed the baseline models, particularly at low data percentages (< 0.25). We conjecture that this may be attributed to our dual-view convolution module, in which local convolutions yield richer graph co-occurrence patterns, particularly in the case of few labels. This finding demonstrates that our model is robust and insensitive to the training data ratio. Therefore, it can effectively handle scenarios where only a limited number of training samples are available, making it applicable to real-world situations.

4.7 Complexity analysis

Notice that the time complexity of the model primarily arises from F in the Algorithm 1. The time complexity is $O(ctq^2)$. Moreover, considering that the parameters t and q are set as small integers in experiments, F can be rapidly computed. Additionally, the algorithm can be parallelized by conducting multiple random walks simultaneously on different parts of a graph. Therefore, the time complexity of the model was deemed acceptable.

Compared with other graph-based models such as MAGNET, LDGN, and LR-GCN, which have shown excellent results in comparative experiments, our model achieves a favorable balance between complexity and efficiency. One of the main contributors to the time complexity of the MAGNET is its graph attention networks. Assume that number of nodes is c, the number of edges is e, and the dimensions before and after feature transformation are d and d', respectively. The time complexity of MAGNET can be expressed as $O(cdd') + O(ed')$. Owing to the potentially large number of edges (e) and relatively large dimensions (d and d'), the event complexity of MAGNET was relatively high. Similarly, for the LDGN, the computational complexity primarily arises from the dynamic reconstruction graph with a time complexity of $O(cdd')$. In a laboratory setting, where dimensions d and d' are relatively large, the event complexity of the LDGN is also high. In comparison, the suboptimal model LR-GCN does not involve redundant multiplication calculations, resulting in a slightly better time complexity than our model. However, as mentioned previously, the time complexity of

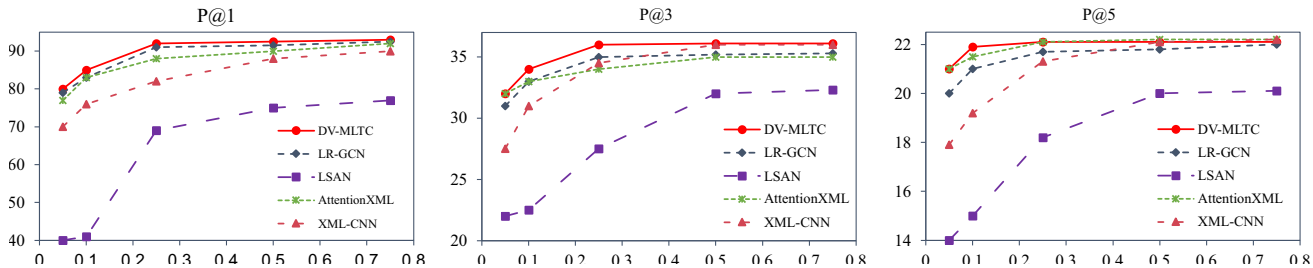


Fig. 9 Test performance (%) with different label ratios on AAPD

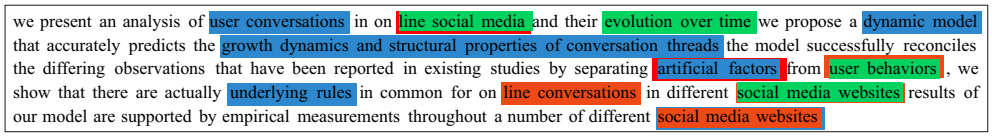


Fig. 10 Visualization of label attention weights. The attention weights of "**physics.soc**" for words are shaded in green, and the attention scores of "**cs.CY**" and "**cs.SI**" are shaded in blue and red respectively

our model remains acceptable. Hence, our model achieves a satisfactory balance between complexity and efficiency. In summary, although other graph-based models may outperform our model in certain comparative experiments, the advantageous balance between complexity and efficiency of our model makes it a valuable choice for practical applications.

4.8 Case studies and visualizations

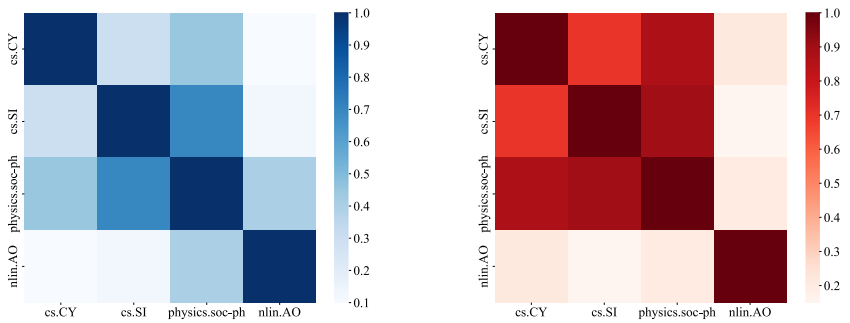
To further verify the effectiveness of our label attention module and dual graph neural networks in DV-MLTC, we present a typical case and visualize the similarity scores between the attention weights of document words and label-specific components using t-sne [45]. We show a testting instance from the original AAPD dataset which belongs to three categories: "physics and society" (**physics.soc-ph**), "computers and society" (**cs.CY**), and "social and information networks" (**cs.SI**).

4.8.1 Label attention visualization

Figure 10 shows the label attention, revealing how different labels focus on specific parts of the document text. Each label assigns importance to its set of words for classification. For instance, in the "**physics.soc-ph**" category, words like "user behaviors" and "evolution over time" were highlighted, capturing key concepts in physics within a social context. In the "**cs.CY**" category, words such as "user conversations", "dynamic model", "growth dynamics and structural properties," and "underlying rules" were emphasized, indicating a focus on computers and society. In the "**cs.SI**" category, attention was given to words such as "artificial factors", "line conversations", and "social media websites". By examining the specific words that receive attention in each category, we gain insights into the semantics and distinguishing aspects of these categories. These visualizations intuitively demonstrate the effectiveness of the model in capturing relevant information in document text for accurate labeling.

4.8.2 Label co-occurrence graph visualization

Figure 11 visualizes the label graph, showing the roles of GlobalConv and LocalConv in capturing the label co-occurrence patterns. The heat maps in Fig. 11 represent the label co-occurrence matrices A^G and A^L. In Fig. 11(a), the heat map shows A^G based on GlobalConv. However, GlobalConv failed to accurately discern the relationships between the labels in this specific test case. Notably, the co-occurrence of "computers and society (**cs.cy**)" and "adaptation and self-organizing systems (**nlin.AO**)" was not considered significant. This limitation arises from relying solely on global statistical information, which may overlook label correlations in individual instances. Conversely, Fig. 11(b) displays A^L based on LocalConv. This highlights the crucial role of LocalConv in establishing local connections between the labels. Even for label pairs with low co-occurrence, such as "computers and society (**cs.CY**)" and "physics and society (**physics.soc-ph**)", LocalConv assigns a label correlation. Multiple label paths generated by LocalConv generalize



(a) Global label co-occurrence visualization

(b) Local label co-occurrence visualization

Fig. 11 Case of label co-occurrence graph visualization

label relationships based on model sampling, independent of human influence. Consequently, LocalConv captures finer label associations, providing a comprehensive understanding of label co-occurrence patterns. In summary, the visualization of the label graph demonstrates how LocalConv effectively supplements the label correlations that GlobalConv alone cannot capture.

5 Conclusion and future tasks

In this study, we propose a novel dual-view convolutional neural network for multi-label text classification. Our approach systematically addresses graph relationships within co-occurrences by employing global and local consistency perspectives. The global consistency convolution utilizes GCNs to model the statistical relationships among graphs based on correlation. For local consistency convolution, we strategically generate graph paths through random walks, reconstruct local graphs, and enrich the co-occurrence patterns. The initial word embeddings were generated via a dual attention flow. Extensive experiments revealed superior performance on RCV1 and EUR-Lex and competitive results on AAPD, highlighting a favorable complexity-efficiency balance. Our approach is effective in enhancing classification performance and mitigating long-tailed issues. Future enhancements include constructing dynamics for sample subsets to reduce computational overhead and further exploring the leveraging of additional graph information for multi-graph text classification.

Acknowledgements This work was supported in part by National Natural Science Foundation of China (No. 61862058), Natural Science Foundation of Gansu Province (No. 20JR5RA518, 21JR7RA114). Industrial Support Project of Gansu Colleges (No. 2022CYZC11).

Author Contributions X.L and B.Y: Conceptualization, Methodology, Formal analysis, Software, Investigation, Validation, Resources, Writing—original draft, review and editing, Visualization. Q.P and S.F: Resources, Writing—review and editing, Supervision.

Availability of Data and Materials The datasets analyzed during the current study were all derived from the following public domain resources. [AAPD: https://git.uwaterloo.ca/jimmylin/Castor-data/tree/master/datasets/AAPD/; RCV1: http://www.ai.mit.edu/projects/jmlr/papers/volume5/lewis04a/lyrl2004_rcv1v2_README.htm; EUR-Lex: http://nlp.cs.aueb.gr/software.html].

Declaration

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

Consent to Participate The authors declare that they agree to participate.

Consent for Publication The authors declare that they agree to publish.

References

1. Huang B, Guo R, Zhu Y, Fang Z, Zeng G, Liu J, Wang Y, Fujita H, Shi Z (2022) Aspect-level sentiment analysis with aspect-specific context position information. Knowl-Based Syst 243:108473. https://doi.org/10.1016/j.knosys.2022.108473
2. Tang P, Jiang M, Xia BN, Pitera JW, Welser J, Chawla NV (2020) Multi-label patent categorization with non-local attention-based graph convolutional network. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 34, pp 9024–9031. https://ojs.aaai.org/index.php/AAAI/article/view/6435
3. Liu W, Wang H, Shen X, Tsang IW (2022) The emerging trends of multi-label learning. IEEE Trans Pattern Anal Mach Intell 44(11):7955–7974. https://doi.org/10.1109/TPAMI.2021.3119334
4. Liu J, Chang W-C, Wu Y, Yang Y (2017) Deep learning for extreme multi-label text classification. In: Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval. SIGIR '17, pp 115–124. Association for computing machinery. https://doi.org/10.1145/3077136.3080834
5. Wu H, Qin S, Nie R, Cao J, Gorbachev S (2021) Effective collaborative representation learning for multilabel text categorization. IEEE Trans Neural Netw Learn Syst 33(10):5200–5214
6. Huang X, Chen B, Xiao L, Yu J, Jing L (2022) Label-aware document representation via hybrid attention for extreme multi-label text classification. Neural Process Lett 54(5):3601–3617
7. Xiao L, Zhang X, Jing L, Huang C, Song M (2021) Does head label help for long-tailed multi-label text classification. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp 14103–14111
8. Zong D, Sun S (2023) Bgnn-xml: bilateral graph neural networks for extreme multi-label text classification. IEEE Trans Knowl Data Eng 35(7):6698–6709
9. Zhang Q-W, Zhang X, Yan Z, Liu R, Cao Y, Zhang M-L (2021) Correlation-guided representation for multi-label text classification. In: IJCAI, pp 3363–3369
10. Ionescu RT, Butnaru A (2019) Vector of locally-aggregated word embeddings (vlawe): a novel document-level representation. In: Proceedings of the 2019 conference of the north american chapter of the association for computational linguistics: human language technologies, vol 1 (Long and Short Papers), pp 363–369. https://doi.org/10.18653/v1/N19-1033. https://aclanthology.org/N19-1033
11. Liu M, Liu L, Cao J, Du Q (2022) Co-attention network with label embedding for text classification. Neurocomputing 471:61–69
12. Wang J, Chen Z, Qin Y, He D, Lin F (2023) Multi-aspect co-attentional collaborative filtering for extreme multi-label text classification. Knowl-Based Syst 260:110110. https://doi.org/10.1016/j.knosys.2022.110110
13. Chen Z-M, Wei X-S, Wang P, Guo Y (2019) Multi-label image recognition with graph convolutional networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 5177–5186
14. Pal A, Selvakumar M, Sankarasubbu M (2020) Magnet: multi-label text classification using attention-based graph neural network. In: Proceedings of the 12th international conference on agents and artificial intelligence 1, vol 2, pp 494–505. https://doi.org/10.5220/0008940304940505
15. Vu H, Nguyen M, Nguyen V, Tien M, Nguyen V (2022) Label correlation based graph convolutional network for multi-label text classification. In: 2022 International joint conference on neural

networks (IJCNN), pp 01–08. https://ieeexplore.ieee.org/abstract/document/9892542
16. Kipf TN, Welling M (2017) Semi-supervised classification with graph convolutional networks. In: International conference on learning representations (ICLR)
17. Liang Z, Guo J, Qiu W, Huang Z, Li S (2024) When graph convolution meets double attention: online privacy disclosure detection with multi-label text classification. Data Min Knowl Discov 1–22
18. Lewis DD, Yang Y, Russell-Rose T, Li F (2004) Rcv1: a new benchmark collection for text categorization research. J Mach Learn Res 5:361–397
19. Yang P, Sun X, Li W, Ma S, Wu W, Wang H (2018) Sgm: sequence generation model for multi-label classification. In: Proceedings of the 27th international conference on computational linguistics, pp 3915–3926. https://aclanthology.org/C18-1330
20. Yang P, Luo F, Ma S, Lin J, Sun X (2019) A deep reinforced sequence-to-set model for multi-label classification. In: Proceedings of the 57th annual meeting of the association for computational linguistics, pp 5252–5258. https://aclanthology.org/P19-1518
21. Liao W, Wang Y, Yin Y, Zhang X, Ma P (2020) Improved sequence generation model for multi-label classification via cnn and initialized fully connection. Neurocomputing 382:188–195
22. Zhang X, Tan X, Luo Z, Zhao J (2023) Multi-label sequence generating model via label semantic attention mechanism. Int J Mach Learn Cybern 14(5):1711–1723
23. Wang R, Ridley R, Qu W, Dai X et al (2021) A novel reasoning mechanism for multi-label text classification. Inf Process Manage 58(2):102441
24. You R, Zhang Z, Wang Z, Dai S, Mamitsuka H, Zhu S (2019) Attentionxml: label tree-based attention-aware deep model for high-performance extreme multi-label text classification. In: Advances in neural information processing systems, vol 32, pp 5820–5830
25. Xiao L, Huang X, Chen B, Jing L (2019) Label-specific document representation for multi-label text classification. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), pp 466–475. Association for Computational Linguistics. https://aclanthology.org/D19-1044
26. Liu Q, Chen J, Chen F, Fang K, An P, Zhang Y, Du S (2023) Mlgn: a multi-label guided network for improving text classification. IEEE Access 11:80392–80402. https://doi.org/10.1109/ACCESS.2023.3299566
27. Qin S, Wu H, Zhou L, Li J, Du G (2023) Learning metric space with distillation for large-scale multi-label text classification. Neural Comput Appl 35(15):11445–11458
28. Wang Q, Zhu J, Shu H, Asamoah KO, Shi J, Zhou C (2023) Gudn: a novel guide network with label reinforcement strategy for extreme multi-label text classification. J King Saud Univ Comput Inf Sci 35(4):161–171
29. Xu P, Xiao L, Liu B, Lu S, Jing L, Yu J (2023) Label-specific feature augmentation for long-tailed multi-label text classification. In: Proceedings of the AAAI conference on artificial intelligence, vol. 37, pp 10602–10610
30. Xiao L, Xu P, Song M, Liu H, Jing L, Zhang X (2023) Triple alliance prototype orthotist network for long-tailed multi-label text classification. IEEE/ACM Trans Audio Speech Lang Process 31:2616–2628. https://doi.org/10.1109/TASLP.2023.3265860
31. Zhang W, Yan J, Wang X, Zha H (2018) Deep extreme multi-label learning. In: Proceedings of the 2018 ACM on international conference on multimedia retrieval, pp 100–107. https://doi.org/10.1145/3206025.3206030
32. Li I, Feng A, Wu H, Li T, Suzumura T, Dong R (2022) LiGCN: label-interpretable graph convolutional networks for multi-label text classification. In: Proceedings of the 2nd workshop on deep learning on graphs for natural language processing (DLG4NLP 2022), pp 60–70. Association for Computational Linguistics. https://aclanthology.org/2022.dlg4nlp-1.7
33. Vu H, Nguyen M, Nguyen V, Pham M, Nguyen V, Nguyen V (2023) Label-representative graph convolutional network for multi-label text classification. Appl Intell 53(12):14759–14774. https://doi.org/10.1007/s10489-022-04106-x
34. Ma Q, Yuan C, Zhou W, Hu S (2021) Label-specific dual graph neural network for multi-label text classification. In: Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (vol 1: Long Papers), pp 3855–3864. Association for computational linguistics
35. Fan C, Chen W, Tian J, Li Y, He H, Jin Y (2023) Accurate use of label dependency in multi-label text classification through the lens of causality. Appl Intell 1–17
36. Zeng D, Zha E, Kuang J, Shen Y (2024) Multi-label text classification based on semantic-sensitive graph convolutional network. Knowl-Based Syst 284:111303
37. Zhao F, Ai Q, Li X, Wang W, Gao Q, Liu Y (2024) Tlc-xml: transformer with label correlation for extreme multi-label text classification. Neural Process Lett 56(1):25
38. Huang Y, Giledereli B, Köksal A, Özgür A, Ozkirimli E (2021) Balancing methods for multi-label text classification with long-tailed class distribution. In: Proceedings of the 2021 conference on empirical methods in natural language processing, pp 8153–8161. Association for computational linguistics
39. Guo H, Li X, Zhang L, Liu J, Chen W (2021) Label-aware text representation for multi-label text classification. In: ICASSP 2021-2021 IEEE International conference on acoustics, speech and signal processing (ICASSP), pp 7728–7732. https://doi.org/10.1109/ICASSP39728.2021.9413921
40. Zhuang C, Ma Q (2018) Dual graph convolutional networks for graph-based semi-supervised classification. In: Proceedings international world wide web conferences steering committee, pp 499–508. https://doi.org/10.1145/3178876.3186116
41. Loza Mencía E, Fürnkranz J (2008) Efficient pairwise multilabel classification for large-scale problems in the legal domain. In: Joint European conference on machine learning and knowledge discovery in databases, pp 50–65
42. Pennington J, Socher R, Manning CD (2014) Glove: global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pp 1532–1543
43. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. Adv Neural Inf Process Syst 26
44. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V (2019) Roberta: a robustly optimized bert pretraining approach. arXiv:1907.11692
45. Maaten L, Hinton G (2008) Visualizing data using t-sne. J Mach Learn Res 9(11)

Xiaohong Li is an associate professor at Northwest Normal University in China. Her current research interests include machine learning and intelligent information processing, with a focus on text mining, recommendation systems, and web data analysis.