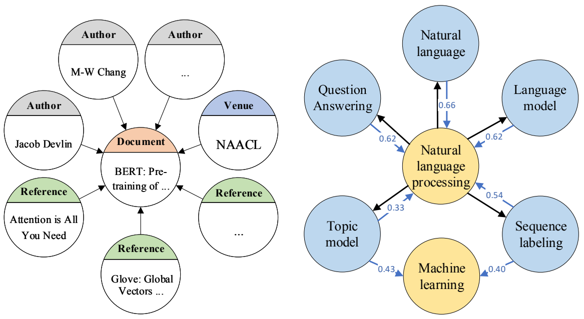
Beyond Text: Incorporating Metadata and Label Structure for Multi-Label Document Classification using Heterogeneous Graphs

Chenchen Ye[1], **Linhai Zhang**[1], **Deyu Zhou**[1,*], **Yulan He**[2 3], **Wu jie**[4]
[1]School of Computer Science and Engineering, Key Laboratory of Computer Network
and Information Integration, Ministry of Education, Southeast University, China
[2]University of Warwick, [3]Alan Turing Institute, UK
[4]Huawei Software Technologies Co., Ltd.
{chenchenye,lzhang472,d.zhou}@seu.edu.cn
yulan.he@warwick.ac.uk, wujie79@huawei.com

Abstract

Multi-label document classification, associating one document instance with a set of relevant labels, is attracting more and more research attention. Existing methods explore the incorporation of information beyond text, such as document metadata or label structure. These approaches however either simply utilize the semantic information of metadata or employ the predefined parent-child label hierarchy, ignoring the heterogeneous graphical structures of metadata and labels, which we believe are crucial for accurate multi-label document classification. Therefore, in this paper, we propose a novel neural network based approach for multi-label document classification, in which two heterogeneous graphs are constructed and learned using heterogeneous graph transformers. One is metadata heterogeneous graph, which models various types of metadata and their topological relations. The other is label heterogeneous graph, which is constructed based on both the labels' hierarchy and their statistical dependencies. Experimental results on two benchmark datasets show the proposed approach outperforms several state-of-the-art baselines.



(a) Metadata Graph.

(b) Label Graph.

Figure 1: An example of a document with metadata and the label relationship in the MAG-CS dataset. In subfigure (b), the arrow in black represents the hierarchy relationship between labels and the arrow in blue represents the statistical dependency relationship between labels.

1 Introduction

With the rapid growth of scientific documents, it is difficult to track related literature manually. For example, there are more than 200,000 publications related to COVID-19 by April 2021. Therefore, it is crucial to automatically assign publications with their corresponding categories. Multi-label document classification (MLDC), associating one document instance with a set of most relevant labels, is attracting more and more research attention.

To tackle the problem, early work focused on learning semantic representations of the input text using some text encoders. For example, Liu et al. (2017) proposed the XML-CNN model using a convolutional neural network and You et al. (2018) proposed the AttentionXML model using a recurrent neural network as text semantic encoder. Recently, Chang et al. (2020) proposed the X-Transformer model, a deep transformer model fine-tuned for MLDC.

Different from the aforementioned approaches, there have been attempts exploring information beyond text for MLDC. On the one hand, the information associated with labels such as label semantics and the relationships between labels are employed. For example, Xiao et al. (2019) generated label-specific document representation using the label semantic information. You et al. (2018) improved the classification performance by constructing a hierarchical label tree. To model label dependency, MLDC is cast as a seq2seq task (Yang et al., 2018). On the other hand, the document metadata is incorporated. For example, to employ the metadata information, the representation of the document and its metadata are learned in the same embedding space (Zhang et al., 2021). The label hierarchy is

[*] Corresponding author.

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 3162–3171
November 7–11, 2021. ©2021 Association for Computational Linguistics

also applied to regularize the output probability of each child label by its parents.

However, existing methods have two limitations: (1) the heterogeneous structure of a document's metadata is ignored. As shown in Figure 1a, for the *BERT* paper (Devlin et al., 2018), a heterogeneous metadata graph can be constructed which consists of multiple authors, multiple references and a publication venue. It can be observed that different types of nodes convey information in different granularity. It is worth noting that labels can also be categorized in different granularity. For example, in Figure 1b, yellow nodes are more coarse-grained compared to blue nodes. The node "NAACL" of the venue type in Figure 1a carries the coarse-grained information relating to the label node "Natural language processing" in Figure 1b, while the node "Glove: Global Vectors for Word Representation" of the reference type in Figure 1a contains the fine-grained information relating to the label node "Word Embedding" in Figure 1b. We believe that modeling such metadata with a heterogeneous graph structure helps to improve the accuracy of final classification. (2) the implicit statistical dependencies between labels are ignored. As shown in Figure 1b, there might exist statistical dependencies between labels. For example, the label "topic model" might have a strong association with "machine learning". But such information is not captured in the label hierarchy.

To tackle the above limitations, in this paper, we propose a novel neural network based approach for multi-label document classification using two heterogeneous graphs. Specifically, a metadata heterogeneous graph with four node types and five edge types is constructed to capture the metadata information and a label heterogeneous graph with two edge types is constructed to capture both the label hierarchy and labels' statistical dependencies. Both graphs are learned using the heterogeneous graph transformer. Moreover, to fully utilize the label information, the label-specific document representation is learned.

The main contributions of this paper are listed as follows:

- A novel neural network based approach is proposed to utilize the information learned from two heterogeneous graphs. As far as we know, we are first to incorporate both the document metadata and the label structure information using the heterogeneous graphs.
- Experimental results on two benchmark datasets show that the proposed approach outperforms existing state-of-the-art approaches for multi-label document classification.

2 Related Work

Based on the information utilized for model learning, approaches for multi-label document classification can be categorized into two types: using textual information only or additionally incorporating external information.

For approaches solely based on textual information, early attempts (Babbar and Schölkopf, 2017; Jain et al., 2016) employed bag-of-words representations. Recently, deep learning based approaches were proposed to learn better text representations. For example, Liu et al. (2017) proposed to use a convolutional neural network for text encoding. You et al. (2018) proposed a neural network approach with the attention mechanism to capture the most relevant part of the input text for each label. Chang et al. (2020) employed a pre-trained Transformer (Vaswani et al., 2017) to capture textual information for text classification. However, such methods ignore the information beyond text, which we believe is crucial for accurate multi-label document classification.

Other approaches attempted to incorporate external information, which can be further classified into two categories, using the document metadata and using label information. For approaches utilizing metadata, Tang et al. (2015) proposed a neural network approach for sentiment analysis which incorporates user and product meta information by a vector space model. Kim et al. (2019) employed categorical metadata signals as additional features to train a deep neural network classifier. (Zhang et al., 2021) developed an approach called MATCH, which pre-trained the embeddings of text and metadata in the same space, and used the Transformer to capture the relationship between them. Nevertheless, these methods ignored the heterogeneous structure of a document's metadata.

Approaches using label information have considered label semantic information, label hierarchy and label dependency. To incorporate label semantic information, Du et al. (2019) proposed an interactive mechanism that incorporates word-level matching signals into the text classification task. LSAN (Xiao et al., 2019) leveraged label semantic information to determine the semantic connec-

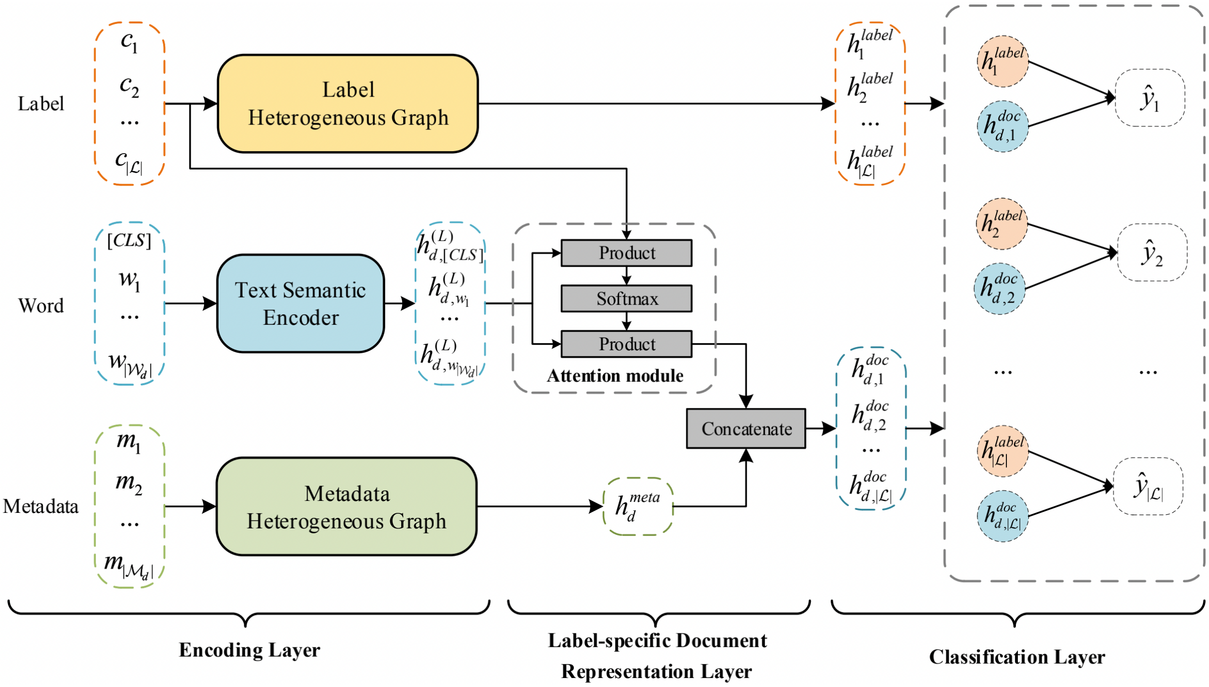


Figure 2: The overall architecture of the proposed model for multi-label document classification. The description of the notations in the figure can be found in Table 1.

tion between each label and the document. Pappas and Henderson (2019) proposed the GILE model, which is a joint input label embedding model for neural text classification. To incorporate label hierarchy, Gopal and Yang (2015) employed a recursive regularization to encourage the similarity between the child classifiers and their parent classifier. Peng et al. (2018) further extended this regularization to a text semantic encoder based on graph convolutional neural networks. Zhang et al. (2018) constructed a label dependency graph to model the label embedding in the label space based on the graph prior. Yang et al. (2018) converted multi-label document classification into a Seq2Seq task, and predicted label sequences with label dependence. Zhang et al. (2021) proposed the MATCH model and employed different ways to regularize the parameters and output probability of each child label by its parents. However, these methods cannot jointly consider label semantic information, label hierarchy, and label dependency.

Our approach is partly inspired by MATCH (Zhang et al., 2021), but with the following differences: (1) the heterogeneous structure of the document's metadata is modeled in the proposed approach, which is ignored in (Zhang et al., 2021); and (2) the proposed approach incorporates the implicit statistical dependencies between labels, which are not considered in (Zhang et al., 2021).

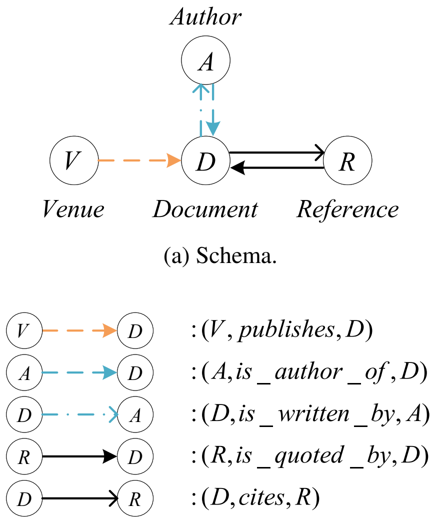
Notation	Description
d	The d-th document
l_i	The i-th label
c_i	Label embedding of l_i
w_j	The j-th word in document text
m_t	The t-th metadata of document
$\hat{y}_i$	Predicted score of l_i
y_i	Ground-truth label of l_i
$\hat{y}_d$	Predicted score for all labels of d
y_d	Ground-truth label for all labels of d
$h^{(L)}_{d,w_j}$	Word representation of w_j in d
h^{meta}_d	Metadata representation of d
h^{label}_i	Label representation of l_i
h^{text}_d	Label-specific text representation of d
h^{doc}_d	Label-specific document representation of d

Table 1: Notations used in the model.

3 Methodology

3.1 Problem Definition

In traditional approaches, the multi-label document classification task is to learn a mapping function $f : \mathcal{W}_d \rightarrow \hat{y}_d$ using only textual information of document d, where $\hat{y}_d$ is the relevant labels of document d and $\mathcal{W}_d$ is the word sequence $w_1, w_2, ..., w_{|\mathcal{W}_d|}$. However, as shown in Figure 1a and 1b, the information beyond text (such as the metadata of documents, the label semantic information, and the label hierarchy) is crucial for accurate document classification.


Author
A
V
D
R
Venue
Document
Reference
V
D
: (V, publishes, D)
A
D
: (A, is _ author _ of, D)
D
A
: (D, is _ written _ by, A)
R
D
: (R, is _ quoted _ by, D)
D
R
: (D, cites, R)

(a) Schema.

(b) Meta relations.

Figure 3: The schema and meta relations of the metadata heterogeneous graph.

Therefore, in this paper, not only the document text but also the information of the metadata and labels is considered. Given a dataset, a label set is denoted as $\mathcal{L} = \{l_1, l_2, ..., l_{|\mathcal{L}|}\}$, and a label hierarchy is represented as a directed acyclic graph $\mathcal{G}^{hierarchy} = (\mathcal{L}, \mathcal{E}_{hierarchy})$, where $\mathcal{E}_{hierarchy}$ is the set of the edges representing the parent-child relationships between labels. The semantic information of each label is represented as an δ-dimension embedding vector $c_i \in \mathbb{R}^{\delta}$. Given a document $d \in \mathcal{D}$ and its word sequence $\mathcal{W}_d$, its associated metadata set $\mathcal{M}_d$ is denoted as:

$$\mathcal{M}_d = (m_1, m_2, ..., m_{|\mathcal{M}_d|} | m_i \in \mathcal{M}) \tag{1}$$

Here, $\mathcal{M}$ is the set of all metadata in the dataset. Take the *BERT* paper in Figure 1a as an example, $\mathcal{W}_d$ is its title and abstract, $\mathcal{M}_d$ is a set of its metadata such as the author of the paper, '*Jacob Devlin*', and the paper it cited, '*Attention is All You Need*'.

The task aims to learn a mapping function $f : (\mathcal{W}_d, \mathcal{M}_d, \mathcal{L}) \rightarrow \hat{y}_d$ that assigns the most relevant labels $\hat{y}_d$ to the document d, while $y_d = \{y_{d,i} \in \{0, 1\}^{|\mathcal{L}|}\}$ is the ground truth labels of the document d.

3.2 Metadata Heterogeneous Graph Construction

In this subsection, we present how to construct a metadata heterogeneous graph (MHG) based on the relationship between documents and their metadata.

The MHG schema is shown in Figure 3a, which contains four types of nodes such as $document$, $author$, $venue$, and $reference$. In addition, there are five types of edges in MHG as shown in Figure 3b. It is worth noting that the $document \rightarrow venue$ relationship is ignored as the node of the venue type is connected with a large number of nodes of the document type, which increases the computational complexity. Taking the MAG-CS dataset as an example, the number of $document \rightarrow venue$ edges is $705,407$, the number of $venue$ nodes is 105, on average, each $venue$ is connected to $6,718$ $document$.

After defining the MHG schema, mathematically, the MHG can be denoted as $\mathcal{G}_{meta} = (\mathcal{V}_{meta}, \mathcal{E}_{meta}, \mathcal{A}_{meta}, \mathcal{R}_{meta})$, where $\mathcal{V}_{meta}$ is the combined set of nodes representing all documents D and nodes representing metadata $\mathcal{M}$, $\mathcal{E}_{meta}$ is the set of edges, $\mathcal{A}_{meta}$ is a set of four node types, and $\mathcal{R}_{meta}$ is a set of five edge types. In the MHG, each node v_{meta} and each edge e_{meta} are associated with their type mapping functions $\tau(v_{meta}) : \mathcal{V}_{meta} \rightarrow \mathcal{A}_{meta}$ and $\phi(e_{meta}) : \mathcal{E}_{meta} \rightarrow \mathcal{R}_{meta}$. Taking Figure 1a as an example, the "NAACL" node belongs to $\mathcal{V}_{meta}$, but its type $venue$ is a node type belonging to $\mathcal{A}_{meta}$.

In MHG, an edge $e_{meta} = (v_i, v_j)$ indicates the relevance between node v_i and v_j, whose weight is determined by

$$\begin{aligned} A_{i,j}^{\phi\left((v_i,v_j)\right)} &= \begin{cases} 1, & \text{if } i \text{ is metadata of } j, \\ 0, & \text{otherwise.} \end{cases} \\ A_{j,i}^{\phi\left((v_j,v_i)\right)} &= \begin{cases} 1, & \text{if } j \text{ has metadata } i, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \tag{2}$$

where $\phi\left((v_i, v_j)\right)$ is the edge type of (v_i, v_j) and A is the adjacency matrix of $\mathcal{G}_{meta}$. For the reverse edge (v_j, v_i), $A_{j,i}^{\phi\left((v_j,v_i)\right)}$ needs to be calculated except for v_j with $venue$ type. For each edge $\phi(e_{meta})$ type, there is a corresponding adjacency matrix $A^{\phi(e_{meta})} \in \mathbb{R}^{|\mathcal{V}_{meta}| \times |\mathcal{V}_{meta}|}$ in the MHG.

3.3 Label Heterogeneous Graph Construction

In this subsection, we introduce how to construct a label heterogeneous graph based on the label hierarchy and statistical dependencies between labels.

Unlike MHG, LHG contains only one type of nodes *label* and two types of edges representing label hierarchy and statistical dependency. Take the "Topic Model" label in Figure 1b as an example, ("NLP", $is\_the\_parent\_label\_of$, "Topic

Model") is an edge representing the label hierarchy, ("Topic Model", *depends_on*, "Machine Learning") is an edge representing the label statistical dependency.

Therefore, we define label heterogeneous graph as $\mathcal{G}_{label} = (\mathcal{V}_{label}, \mathcal{E}_{label}, \mathcal{A}_{label}, \mathcal{R}_{label})$, where $\mathcal{V}_{label}$ is the set of the label nodes, $\mathcal{E}_{label}$ is the set of all edges, $\mathcal{A}_{label}$ is the set containing only the $label$ node type, and $\mathcal{R}_{label}$ is the set consisting of two edge types.

A label hierarchy edge (v_i, v_j) indicates the parent-child relevance of two label node v_i and v_j, whose weight is determined by

$$A^{hierarchy}_{i,j} = \begin{cases} 1, & \text{if } i \text{ is parent of } j, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

where $A^{hierarchy}$ is the adjacency matrix of the label hierarchy in $\mathcal{G}_{label}$.

Following Chen et al. (2019), the probability between two labels is employed to represent the label statistical dependency. For example, as shown in Figure 1b, the label statistical dependency is in the form of conditional probability, i.e., $P(B|A)$ which denotes the probability of the presence of B when A appears. In addition, we use the threshold λ to filter noisy edges in the label statistical dependency edges. Thus, we can obtain the adjacency matrix $A^{dependency}$ of the label statistical dependency in $\mathcal{G}_{label}$ as

$$A^{dependency}_{i,j} = \begin{cases} 1, & \text{if } P(j|i) \geq \lambda, \\ 0, & \text{if } P(j|i) < \lambda. \end{cases} \tag{4}$$

The adjacency matrix of LHG can be defined as $A = \{A^{dependency}, A^{hierarchy}\}$.

3.4 Model Architecture

In this subsection, we introduce the architecture of the proposed model as shown in Figure 2. It consists of three components: (1) an encoding layer, (2) a label-specific document representation layer and (3) a classification layer. The first layer aims to obtain the text representation and the metadata representation of an input document, and also the label representation based on the label heterogeneous graph. The second layer is designed to generate a label-specific document representation by fusing text representation, label semantic representation and metadata representation. Finally, the last layer employs the label-specific document representation and the label representation based on LHG to predict the most relevant labels.

Encoding Layer The encoding layer consists of three parts. The first part is a text semantic encoder built on the multi-layers Transformer (Vaswani et al., 2017) to capture the text semantics information. The second part is a metadata heterogeneous graph where the heterogeneous structure of the document's metadata is learned through the heterogeneous graph transformer (HGT) (Hu et al., 2020). The last part is a label heterogeneous graph through which the label representation is learned via HGT.

The input to the Text Semantic Encoder is a word sequence of a document, prepended by a [CLS] token as typically done in BERT (Devlin et al., 2018). That is, for a document d, the input to the Text Semantic Encoder, denoted as $h_d^{(0)}$, is:

$$h_d^{(0)} = [\mathrm{e}_{[CLS]}; \mathrm{e}_{w_1}; \mathrm{e}_{w_2}; ...; \mathrm{e}_{w_{|\mathcal{W}_d|}}]. \tag{5}$$

Here, $h_d^{(0)} \in \mathbb{R}^{(1+|\mathcal{W}_d|)\times\delta}$, where δ is the dimension of the word representation space and $\mathrm{e}_{w_1} \in \mathbb{R}^k$ is the word embedding of the token w_1. One Transformer layer can be described by

$$\begin{aligned} z_i &= Norm(\mathrm{e}_i + MHA(\mathrm{e}_i, h_d^{(0)})), \\ h_i &= Norm(z_i + FFN(z_i)). \end{aligned} \tag{6}$$

where $Norm(.)$ is the normalization layer, $MHA(.)$ is multi-head attention operator, and $FFN(.)$ is the position-wise feed-forward network (Vaswani et al., 2017). We can stack L Transformer layers to model text semantics, where the l-th layer $h_d^{(l)}$ is also the input of the $(l+1)$-th layer. Therefore, we can obtain the word representation $h_d^{(L)} \in \mathbb{R}^{(1+|\mathcal{W}_d|)\times\delta}$ and text representation $h_{d,[CLS]}^{(L)} \in \mathbb{R}^{\delta}$ of document text through the text semantic encoder.

In order to model the heterogeneous graph structure of metadata and label, the heterogeneous graph transformer (HGT) (Hu et al., 2020) is employed to build two heterogeneous graph neural networks. HGT consists of three components: Heterogeneous Mutual Attention $ATT(.)$, Heterogeneous Message Passing $MSG(.)$ and Target-Specific Aggregation $AGG(.)$. When t is the target node, the layer-wise propagation rule of the HGT at layer $l-1 \in [0, L]$ is defined as:

$$\begin{aligned} h^{(l)}[t] &= AGG\Big(t, \tilde{h}^{(l)}[t]\Big) + h^{(l-1)}[t] \\ \tilde{h}^{(l)}[t] &= \underset{\forall s \in N(t)}{\oplus}\Big(ATT(s,e,t) \odot MSG(s,e,t)\Big) \end{aligned} \tag{7}$$

Here e is the edge between s and t, and $N(t)$ is all neighbor nodes of t. $h^{(l)}[t] \in \mathbb{R}^{\delta}$ is the representation of the target node t at the l-th layer. In the metadata heterogeneous graph neural network, we set the document node as the target node, and use the text representation $h_{d,[CLS]} \in \mathbb{R}^{\delta}$ as the embedding of the document d node. We can obtain the metadata representation of document d by:

$$h_d^{meta} = MHG(\mathcal{G}_{meta,d}, h_{d,[CLS]}^{(L)}) \tag{8}$$

Here $\mathcal{G}_{meta,d}$ is a sub-graph composed of the document node d and its neighbor nodes.

In the label heterogeneous graph neural network, we set the label nodes as target nodes, and use label embedding C to initialize the embedding of label nodes. We can obtain the label representation $h^{label} \in \mathbb{R}^{|\mathcal{L}| \times \delta}$:

$$h^{label} = LHG(\mathcal{G}_{label}, C) \tag{9}$$

Different from the metadata heterogeneous graph neural network, to obtain the representation of the labels, $\mathcal{G}_{label}$ is set as full-graph.

Label-specific Document Representation Layer The label-specific document representation layer aims to obtain the label-related representation based on the document text and metadata. To obtain the label-specific text representation $h_d^{text} \in \mathbb{R}^{|\mathcal{L}| \times \delta}$ using the attention mechanism, we construct the query vector $Q_{label} \in \mathbb{R}^{|\mathcal{L}| \times \delta}$ using label semantics embedding C and construct the key vector K_d and the value vector V_d using the representation of the document d, $h_d^{(L)}$:

$$\begin{aligned} Q_{label} &= tanh(Linear_q(C)) \\ K_d &= tanh(Linear_k(h_d^{(L)})) \\ V_d &= tanh(Linear_v(h_d^{(L)})) \\ Att_d^{label} &= Softmax(Q_{label} \odot K_d^T) \\ h_d^{text} &= tanh(Att_d^{label} \odot V_d) \end{aligned} \tag{10}$$

Here, $Linear(.)$ is a linear transform layer, $tanh(.)$ is the activation function, $Att_d^{label} \in \mathbb{R}^{|\mathcal{L}| \times (1+|\mathcal{W}_d|)}$ is the attention score between the labels and words, and $\odot$ is the dot product operation. Then, we concatenate the text representation $h_{d,i}^{text}$ along the i-th label and the document d's metadata representation h_d^{meta}:

$$h_{d,i}^{doc} = tanh(Linear(h_{d,i}^{text} \oplus h_d^{meta})) \tag{11}$$

Here, $h_d^{doc} \in \mathbb{R}^{|\mathcal{L}| \times \delta}$ is the label-specific document d representation with the metadata.

	MAG-CS	PubMed
#Training Docs	564,340	718,837
#Validation Docs	70,534	89,855
#Testing Docs	70,533	89,854
#Labels	15,809	17,963
#Labels / Doc	5.60	7.78
Vocabulary Size	425,345	776,975
#Words / Doc	126.33	198.97
#Authors	818,927	2,201,919
#Venues	105	150
#Document-Author Edges	2,274,546	5,989,142
#Document-Venue Edges	705,407	898,546
#Document-Document Edges	1,518,466	4,455,702
#Edges in Taxonomy	27,288	22,842
#Layers of Taxonomy	6	15

Table 2: Dataset statistics.

Classification Layer The classification layer aims to assign the most relevant labels $\hat{y}_d$ to the document d. We dot product the label representation h^{label} with the label-specific document representation h_d^{doc}, and then use the $sigmoid$ activation function to obtain the multi-label prediction:

$$\hat{y}_d = sigmoid(h_d^{doc} \odot h^{label}) \tag{12}$$

Finally, the cross-entropy loss is used:

$$\begin{aligned} \mathcal{J}_{BCE} = - \sum_{d \in \mathcal{D}} \sum_{l \in \mathcal{L}} \Big(& y_{d,l} log(\hat{y}_{d,l}) \\ & + (1 - y_{d,l}) log(1 - \hat{y}_{d,l}) \Big) \end{aligned} \tag{13}$$

4 Experiments

In this section, we evaluate the proposed model on two large-scale datasets for extreme multi-label document classification (with the number of labels more than 15,000).

4.1 Experiments Setting

Datasets Two large-scale datasets[1] constructed by Zhang et al. (2021) are employed:

- **MAG-CS**: focusing on the computer science domain based on the Microsoft Academic Graph (MAG). Papers published in 105 top CS conferences from 1990 to 2020 are collected.
- **PubMed**: containing papers published in 150 top journals in medicine from 2010 to 2020. Each paper is tagged with related MeSH

[1] https://drive.google.com/file/d/1pn9WhPxIR4J7Wgm5_AJLgNHvTaMexDcC

Dataset	Method	P@1=nDCG@1	P@3	P@5	nDCG@3	nDCG@5
MAG-CS	XML-CNN	0.8656 ± 0.0006	0.7028 ± 0.0010	0.5756 ± 0.0010	0.7842 ± 0.0009	0.7407 ± 0.0009
	MeSHProbeNet	0.8738 ± 0.0016	0.7219 ± 0.0059	0.5927 ± 0.0075	0.8020 ± 0.0048	0.7588 ± 0.0067
	AttentionXML	0.9035 ± 0.0009	0.7682 ± 0.0017	0.6441 ± 0.0020	0.8489 ± 0.0016	0.8145 ± 0.0020
	Star-Transformer	0.8569 ± 0.0011	0.7089 ± 0.0010	0.5853 ± 0.0011	0.7876 ± 0.0008	0.7486 ± 0.0011
	BERTXML	0.9011 ± 0.0027	0.7532 ± 0.0015	0.6238 ± 0.0020	0.8355 ± 0.0025	0.7954 ± 0.0024
	Transformer	0.8805 ± 0.0007	0.7327 ± 0.0006	0.6024 ± 0.0010	0.8129 ± 0.0008	0.7703 ± 0.0010
	MATCH	0.9190 ± 0.0012	0.7763 ± 0.0023	0.6457 ± 0.0030	0.8610 ± 0.0022	0.8223 ± 0.0030
	Ours w/o LHG	0.9160 ± 0.0009	0.7867 ± 0.0005	0.6659 ± 0.0005	0.8684 ± 0.0007	0.8387 ± 0.0008
	Ours w/o MHG	0.9088 ± 0.0021	0.7815 ± 0.0033	0.6620 ± 0.0026	0.8596 ± 0.0033	0.8296 ± 0.0030
	Ours	**0.9312 ± 0.0025**	**0.8022 ± 0.0013**	**0.6797 ± 0.0015**	**0.8847 ± 0.0017**	**0.8546 ± 0.0019**
PubMed	XML-CNN	0.9084 ± 0.0004	0.7182 ± 0.0007	0.5857 ± 0.0004	0.7790 ± 0.0007	0.7075 ± 0.0005
	MeSHProbeNet	0.9135 ± 0.0021	0.7224 ± 0.0066	0.5878 ± 0.0070	0.7836 ± 0.0057	0.7109 ± 0.0065
	AttentionXML	0.9125 ± 0.0003	0.7414 ± 0.0017	0.6169 ± 0.0016	0.7979 ± 0.0013	0.7341 ± 0.0013
	Star-Transformer	0.8962 ± 0.0023	0.6990 ± 0.0014	0.5641 ± 0.0008	0.7612 ± 0.0015	0.6869 ± 0.0011
	BERTXML	0.9144 ± 0.0014	0.7362 ± 0.0046	0.6032 ± 0.0050	0.7949 ± 0.0038	0.7247 ± 0.0045
	Transformer	0.8971 ± 0.0050	0.7299 ± 0.0029	0.6003 ± 0.0018	0.7867 ± 0.0034	0.7178 ± 0.0027
	MATCH	0.9168 ± 0.0013	0.7511 ± 0.0029	0.6199 ± 0.0029	0.8072 ± 0.0027	0.7395 ± 0.0029
	Ours w/o LHG	0.9165 ± 0.0010	0.7556 ± 0.0013	0.6345 ± 0.0015	0.8101 ± 0.0013	0.7497 ± 0.0014
	Ours w/o MHG	0.9230 ± 0.0015	0.7662 ± 0.0010	0.6440 ± 0.0004	0.8201 ± 0.0011	0.7598 ± 0.0005
	Ours	**0.9352 ± 0.0012**	**0.7900 ± 0.0012**	**0.6662 ± 0.0013**	**0.8420 ± 0.0012**	**0.7822 ± 0.0014**

Table 3: Experimental Results on MAG-CS and PubMed datasets.

terms[2], which are viewed as labels in the MLDC task.

For both datasets, metadata information (includes disambiguated authors, venues, and references) is collected from MAG. The text information of each document is its title and abstract. Table 2 shows the statistics of the two datasets.

Baselines The following extreme multi-label document classification methods and transformer-based models are chosen as the baselines.

- **XML-CNN** (Liu et al., 2017) is an extreme multi-label document classification model built on convolutional neural networks.
- **MeSHProbeNet** (Xun et al., 2019) solves the problem of multi-label document classification with recurrent neural networks and multiple MeSH "probes".
- **AttentionXML** (You et al., 2018) is an extreme multi-label document classification model that is built on RNN with a label-aware attention layer.
- **Transformer** (Vaswani et al., 2017) is composed of multiple self-attention layers.
- **Star-Transformer** (Guo et al., 2019) sparsifies the fully connected attention in the Transformer to a star-shaped structure.
- **BERTXML** (Xun et al., 2020) is a model inspired by BERT (Devlin et al., 2018) with a multi-layer Transformer and multiple *[CLS]* symbols.
- **MATCH**[3] (Zhang et al., 2021) is a multi-label document classification method with metadata-aware Transformer and label hierarchy.

Evaluation Metrics Two widely used metrics, precision at top k ($P@k$) and Normalized Discounted Cumulative Gains at top k ($nDCG@k$), are used to evaluate the model performance [4].

$$
\begin{aligned}
P@k &= \frac{1}{k}\sum_{i=1}^{k} y_{d,rank(i)}.\\
DCG@k &= \sum_{i=1}^{k} \frac{y_{d,rank(i)}}{log(i+1)}, \qquad (14)\\
nDCG@k &= \frac{DCG@k}{\sum_{i=1}^{min(k,||y_d||_0)} \frac{1}{log(i+1)}}.
\end{aligned}
$$

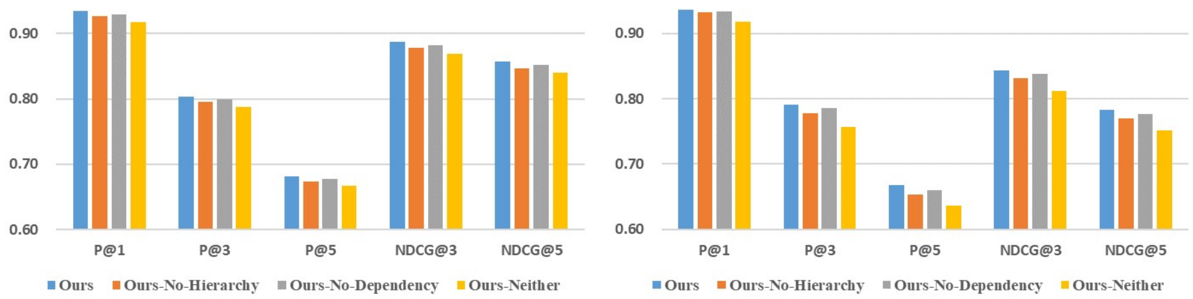
Here, $y_d \in \{0,1\}^{|\mathcal{L}|}$ is the ground truth label vector of the document d, $rank(i)$ is the index of the i-th highest predicted label .

Parameter Setting For all methods, the embedding dimension k is set to 100, and *GloVe.6B.100d* (Pennington et al., 2014) is used to initialize word embeddings. For our method, we set the hidden vector dimension $\delta = 100$, the number of the text

[2] https://www.nlm.nih.gov/mesh/meshhome.html

[3] https://github.com/yuzhimanhua/MATCH

[4] https://github.com/yuzhimanhua/MATCH/blob/master/deepxml/evaluation.py



(a) MAG-CS

(b) PubMed

Figure 4: Ablation analysis of Comprehensive Label Information.

encode layers $L = 4$, the threshold of label dependency $\lambda = 0.3$, and the number of attention heads 2. The training is performed using Adam (Kingma and Ba, 2014) with a batch size of 200 and a learning rate of 1e-3, and the maximum training epochs is 20. The compared methods use their default parameter setting.

4.2 Results

Table 3 shows the performance comparison of the proposed approach with other baselines. Experiments are conducted three times with the mean and standard deviations reported. According to Eq.14, it is easy to show that $P@1 = nDCG@1$ if each document has at least one true label.

It can observed from Table 3 that: (1) the proposed model is significantly better than all compared models on both datasets. (2) the proposed model is also superior to the two ablation models *Ours without LHG* and *Ours without MHG* on both datasets. It shows that the label heterogeneous graph and the metadata heterogeneous graph are effective for the MLDC task. (3) The label-aware models (such as AttentionXML, MATCH and Ours) significantly outperform other models for MLDC on both datasets. It shows that label-awareness is essential, and modeling label information can further improve the classification performance.

It is worth noting that the performance improvement by incorporating the label heterogeneous graph is more significant on PubMed. In specific, on MAG-CS, the proposed model has an average absolute improvement of 1.5% on five metrics in comparison with *ours w/o LHG*, while on PubMed, the improvement is 2.9%. It might attribute to the different label dependencies in the two datasets. When the label dependency threshold is $\lambda = 0.3$, the label dependency edge number on MAG-CS is $67,620$, while in the PubMed dataset, the number is $88,390$.

4.3 Effect of Comprehensive Label Info

In both datasets, the label hierarchy is available and we construct the label statistical dependencies by calculating the conditional probability between the labels. To explore the effectiveness of each type of relationships, we conduct an ablation analysis. Three ablation versions of the proposed model are constructed: *No-Hierarchy*, *No-Dependency*, *Neither*. For *No-Hierarchy*, we remove the edges of the label hierarchical relationship from the label heterogeneous graph. We construct the model variants, *No-Dependency* and *Neither*, in a similar way by removing edges of their corresponding types.

The performance comparisons of the proposed model with its three ablations versions is shown in Figure 4. It can be observed that: (1) The proposed model outperforms *No-Hierarchy*, *No-Dependency*, *Neither* in all metrics, indicating that both types of label relationships play a vital role in MLDC task. (2) Among the three ablation versions, *Neither* has the worst performance which shows that the label hierarchical relationship and the label dependency relationship are complementary. It can also be observed that the method with label hierarchy achieves larger improvement. It is because that the label hierarchy information comes from the rigorous label taxonomy, while the label dependency information comes from the label probability statistics.

5 Conclusions

We propose a novel neural network approach for multi-label document classification, in which two heterogeneous graphs are incorporated and learned using heterogeneous graph transformers. One is the metadata heterogeneous graph, which models various type of metadata and their topological relations. The other is the label heterogeneous graph, which is constructed based on the labels' hierarchy and statistical dependency. Experiments on two datasets show the superior performance of the proposed approach over existing approaches. In addition, results of the ablation experiments show the effectiveness of incorporating both the metadata heterogeneous graph and the label heterogeneous graph for multi-label document classification.

Acknowledgements

We would like to thank anonymous reviewers for their valuable comments and helpful suggestions. This work was funded by the National Natural Science Foundation of China (61772132), and the UK Engineering and Physical Sciences Research Council (grant no. EP/T017112/1, EP/V048597/1). YH is supported by a Turing AI Fellowship funded by the UK Research and Innovation (grant no. EP/V020579/1).

References

Rohit Babbar and Bernhard Schölkopf. 2017. Dismec: Distributed sparse machines for extreme multi-label classification. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 721–729.

Wei-Cheng Chang, Hsiang-Fu Yu, Kai Zhong, Yiming Yang, and Inderjit S Dhillon. 2020. Taming pretrained transformers for extreme multi-label text classification. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3163–3171.

Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. 2019. Multi-label image recognition with graph convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5177–5186.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Cunxiao Du, Zhaozheng Chen, Fuli Feng, Lei Zhu, Tian Gan, and Liqiang Nie. 2019. Explicit interaction model towards text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6359–6366.

Siddharth Gopal and Yiming Yang. 2015. Hierarchical bayesian inference and recursive regularization for large-scale classification. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 9(3):1–23.

Qipeng Guo, Xipeng Qiu, Pengfei Liu, Yunfan Shao, Xiangyang Xue, and Zheng Zhang. 2019. Star-transformer. *arXiv preprint arXiv:1902.09113*.

Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. 2020. Heterogeneous graph transformer. In *Proceedings of The Web Conference 2020*, pages 2704–2710.

Himanshu Jain, Yashoteja Prabhu, and Manik Varma. 2016. Extreme multi-label loss functions for recommendation, tagging, ranking & other missing label applications. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 935–944.

Jihyeok Kim, Reinald Kim Amplayo, Kyungjae Lee, Sua Sung, Minji Seo, and Seung-won Hwang. 2019. Categorical metadata representation for customized text classification. *Transactions of the Association for Computational Linguistics*, 7:201–215.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Jingzhou Liu, Wei-Cheng Chang, Yuexin Wu, and Yiming Yang. 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*, pages 115–124.

Nikolaos Pappas and James Henderson. 2019. Gile: A generalized input-label embedding for text classification. *Transactions of the Association for Computational Linguistics*, 7:139–155.

Hao Peng, Jianxin Li, Yu He, Yaopeng Liu, Mengjiao Bao, Lihong Wang, Yangqiu Song, and Qiang Yang. 2018. Large-scale hierarchical text classification with recursively regularized deep graph-cnn. In *Proceedings of the 2018 world wide web conference*, pages 1063–1072.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Duyu Tang, Bing Qin, and Ting Liu. 2015. Learning semantic representations of users and products for document level sentiment classification. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1014–1023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Lin Xiao, Xin Huang, Boli Chen, and Liping Jing. 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)*, pages 466–475.

Guangxu Xun, Kishlay Jha, Jianhui Sun, and Aidong Zhang. 2020. Correlation networks for extreme multi-label text classification. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1074–1082.

Guangxu Xun, Kishlay Jha, Ye Yuan, Yaqing Wang, and Aidong Zhang. 2019. Meshprobenet: a self-attentive probe net for mesh indexing. *Bioinformatics*, 35(19):3794–3802.

Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018. Sgm: sequence generation model for multi-label classification. *arXiv preprint arXiv:1806.04822*.

Ronghui You, Suyang Dai, Zihan Zhang, Hiroshi Mamitsuka, and Shanfeng Zhu. 2018. Attentionxml: Extreme multi-label text classification with multi-label attention based recurrent neural networks. *arXiv preprint arXiv:1811.01727*.

Wenjie Zhang, Junchi Yan, Xiangfeng Wang, and Hongyuan Zha. 2018. Deep extreme multi-label learning. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, pages 100–107.

Yu Zhang, Zhihong Shen, Yuxiao Dong, Kuansan Wang, and Jiawei Han. 2021. Match: Metadata-aware text classification in a large hierarchy. *arXiv preprint arXiv:2102.07349*.