Hierarchy-Aware Global Model for Hierarchical Text Classification

Jie Zhou[1,2*], **Chunping Ma**[2], **Dingkun Long**[2], **Guangwei Xu**[2],
Ning Ding[3], **Haoyu Zhang**[4], **Pengjun Xie**[2], **Gongshen Liu**[1†]
[1]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University
[2]Alibaba Group, [3]Tsinghua University, [4]National University of Defense Technology
{sanny02,lgshen}@sjtu.edu.cn,
{kunka.xgw,chengchen.xpj}@taobao.com
{chunping.mcp,dingkun.ldk}@alibaba-inc.com
{dingn18}@mails.tsinghua.edu.cn, {zhanghaoyu10}@nudt.edu.cn

Abstract

Hierarchical text classification is an essential yet challenging subtask of multi-label text classification with a taxonomic hierarchy. Existing methods have difficulties in modeling the hierarchical label structure in a global view. Furthermore, they cannot make full use of the mutual interactions between the text feature space and the label space. In this paper, we formulate the hierarchy as a directed graph and introduce hierarchy-aware structure encoders for modeling label dependencies. Based on the hierarchy encoder, we propose a novel end-to-end hierarchy-aware global model (HiAGM) with two variants. A multi-label attention variant (HiAGM-LA) learns hierarchy-aware label embeddings through the hierarchy encoder and conducts inductive fusion of label-aware text features. A text feature propagation model (HiAGM-TP) is proposed as the deductive variant that directly feeds text features into hierarchy encoders. Compared with previous works, both HiAGM-LA and HiAGM-TP achieve significant and consistent improvements on three benchmark datasets.

1 Introduction

Text classification is widely used in Natural Language Processing (NLP) applications, such as sentimental analysis (Pang and Lee, 2007), information retrieval (Liu et al., 2015), and document categorization (Yang et al., 2016). Hierarchical text classification (HTC) is a particular multi-label text classification (MLC) problem, where the classification result corresponds to one or more nodes of a taxonomic hierarchy. The taxonomic hierarchy is commonly modeled as a tree or a directed acyclic graph, as depicted in Figure 1.

Existing approaches for HTC could be categorized into two groups: local approach and global approach. The first group tends to constructs multiple classification models and then traverse the hierarchy in a top-down manner. Previous local studies (Wehrmann et al., 2018; Shimura et al., 2018; Banerjee et al., 2019) propose to overcome the data imbalance on child nodes by learning from parent one. However, these models contain a large number of parameters and easily lead to exposure bias for the lack of holistic structural information. The global approach treats HTC problem as a flat MLC problem, and uses one single classifier for all classes. Recent global methods introduce various strategies to utilize structural information of top-down paths, such as recursive regularization (Gopal and Yang, 2013), reinforcement learning (Mao et al., 2019) and meta-learning (Wu et al., 2019). There is so far no global method that encodes the holistic label structure for label correlation features. Moreover, these methods still exploit the hierarchy in a shallow manner, thus ignoring the fine-grained label correlation information that has proved to be more fruitful in our work.

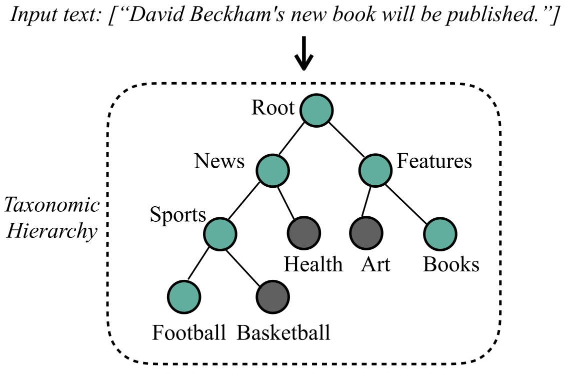

Figure 1: This short sample is tagged with *news*, *sports*, *football*, *features* and *books*. Note that HTC could be either a single-path or a multi-path problem.

In this paper, we formulate the hierarchy as a directed graph and utilize prior probabilities of label dependencies to aggregate node information. A hierarchy-aware global model (HiAGM) is pro-

[*] This work was done during intern at Alibaba Group.
[†] Corresponding author.

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 1106–1117
July 5 - 10, 2020. ©2020 Association for Computational Linguistics

posed to enhance textual information with the label structural features. It comprises a traditional text encoder for extracting textual information and a hierarchy-aware structure encoder for modeling hierarchical label relations. The hierarchy-aware structure encoder could be either a TreeLSTM or a hierarchy-GCN where hierarchical prior knowledge is integrated. Moreover, these two structure encoders are bidirectionally calculated, allowing them to capture label correlation information in both top-down and bottom-up manners. As a result, HiAGM is more robust than previous top-down models and is able to alleviate the problems caused by exposure bias and imbalanced data.

To aggregate text features and label structural features, we present two variants of HiAGM, a multi-label attention model HiAGM-LA and a text feature propagation model HiAGM-TP. Both variants extract hierarchy-aware text features based on the structure encoders. HiAGM-LA extracts the inductive label-wise text features while HiAGM-TP generates hybrid information in a deductive manner. Specifically, HiAGM-LA updates the label embedding across the holistic hierarchy and then employs node outputs as the hierarchy-aware label representations. Finally, it conducts multi-label attention for label-aware text features. On the other hand, HiAGM-TP directly utilizes text features as the input of the structure encoder in a serial dataflow. Hence it propagates textual information throughout the overall hierarchy. The hidden state of each node in the entire hierarchy represents the class-specific textual information.

The major contributions of this paper are:

- With the prior hierarchy knowledge, we adopt typical structure encoders for modeling label dependencies in both top-down and bottom-up manners, which has not been investigated for hierarchical text classification.
- We propose a novel end-to-end hierarchy-aware global model (HiAGM). We further present two variants for label-wise text features, a hierarchy-aware multi-label attention model (HiAGM-LA) and a hierarchy-aware text feature propagation model (HiAGM-TP).
- We empirically demonstrate that both variants of HiAGM achieve consistent improvements on various datasets when using different structure encoders. Our best model outperforms the state-of-the-art model by 3.25% of Macro-F1 and 0.66% of Micro-F1 on RCV1-V2.
- We release our code and experimental splits of Web-of-Science and NYTimes for reproducibility. [1]

2 Related Work

Existing works for HTC could be categorized into local and global approaches. Local approaches could be subdivided into local classifier per node (LCN) (Banerjee et al., 2019), local classifier per parent node (LCPN) (Dumais and Chen, 2000), and local classifier per level (LCL)(Shimura et al., 2018; Wehrmann et al., 2018; Kowsari et al., 2017). Banerjee et al. (2019) transfers parameters of the parent model for child models as LCN. Wehrmann et al. (2018) alleviates exposure bias problem by the hybrid of LCL and global optimizations. Peng et al. (2018) decomposes the hierarchy into subgraphs and conducts Text-GCN on n-gram tokens.

The global approach improves flat MLC models with the hierarchy information. Cai and Hofmann (2004) modifies SVM to Hierarchical-SVM by decomposition. Gopal and Yang (2013) proposes a simple recursive regularization of parameters among adjacent classes. Deep learning architectures are also employed in global models, such as sequence-to-sequence (Yang et al., 2018), meta-learning (Wu et al., 2019), reinforcement learning (Mao et al., 2019), and capsule network (Peng et al., 2019). Those models mainly focus on improving decoders based on the constraint of hierarchical paths. In contrast, we propose an effective hierarchy-aware global model, HiAGM, that extracts label-wise text features with hierarchy encoders based on prior hierarchy information.

Moreover, the attention mechanism is introduced in MLC by Mullenbach et al. (2018) for ICD coding. Rios and Kavuluru (2018) trains label representation through basic GraphCNN and conducts mutli-label attention with residual shortcuts. AttentionXML (You et al., 2019) converts MLC to a multi-label attention LCL model by label clusters. Huang et al. (2019) improves HMCN (Wehrmann et al., 2018) with label attention per level. Our HiAGM-LA, however, employs multi-label attention in a single model with a simplified structure encoder, reducing the computational complexity.

Recent works, in semantic analysis (Chen et al., 2017b), semantic role labeling (He et al., 2018) and machine translation (Chen et al., 2017a), shows the improvement on sentence representation of syntax

[1] https://github.com/Alibaba-NLP/HiAGM

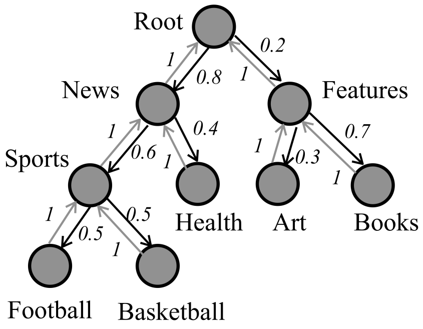

Figure 2: Example of the taxonomic hierarchy. The number indicates the prior probability of label dependencies according to the training corpus.

encoder, such as Tree-Based RNN (Tai et al., 2015; Chen et al., 2017a) and GraphCNN (Marcheggiani and Titov, 2017). We modify those structure encoders for HTC with fine-grained prior knowledge in both top-down and bottom-up manners.

3 Problem Definition

Hierarchical text classification (HTC), a subtask of text classification, organizes the label space with a predefined taxonomic hierarchy. The hierarchy is predefined based on holistic corpus. The hierarchy groups label subsets according to class relations. The taxonomic hierarchy mainly contains the tree-like structure and the directed acyclic graph (DAG) structure. Note that DAG can be converted into a tree-like structure by distinguishing each label node as a single-path node. Thus, the taxonomic hierarchy can be simplified as a tree-like structure.

As illustrated in Figure 2, we formulate a taxonomic hierarchy as a directed graph $G = (V, \overrightarrow{E}, \overleftarrow{E})$ where V refers to the set of label nodes $V = \{v_1, v_2, \dots, v_C\}$ and C denotes the number of label nodes. $\overrightarrow{E} = \{(v_i, v_j) | i \in V, j \in child(i)\}$ is the top-down hierarchy path and $\overleftarrow{E} = \{(v_j, v_i) | i \in V, j \in child(i)\}$ is the bottom-up hierarchy path. Formally, we define HTC as $H = (X, L)$ with a sequence of text objects $X = (x_1, x_2, \dots, x_N)$ and an aligned sequence of supervised label sets $L = (l_1, l_2, \dots, l_N)$.

As depicted in Figure 1, each sample x_i corresponds to a label set l_i that includes multiple classes. Those corresponding classes belong to either one or more sub-paths in the hierarchy. Note that the sample belongs to the parent node v_i in the condition pertaining to the child node $v_j \in child(i)$.

4 Hierarchy-Aware Global Model

As depicted in Figure 3, we propose a **H**ierarchy-**A**ware **G**lobal **M**odel (HiAGM) that leverages the fine-grained hierarchy information and then aggregates label-wise text features. HiAGM consists of a traditional text encoder for textual information and a hierarchy-aware structure encoder for hierarchical label correlation features.

We present two variants of HiAGM for hybrid information aggregation, a multi-label attention model (HiAGM-LA) and a text feature propagation model (HiAGM-TP). HiAGM-LA updates label representations with the structure encoder and generates label-aware text features with multi-label attention mechanism. HiAGM-TP propagates text representations throughout the holistic hierarchy, thus obtaining label-wise text features with the fusion of label correlations.

4.1 Prior Hierarchy Information

The taxonomic hierarchy describes the hierarchical relations among labels. The major bottleneck of HTC is how to make full use of this established structure. Previous studies directly utilize this hierarchy path in a static method based on a pipeline framework, hierarchical model or label assignment model. In contrast, based on Bayesian statistical inference, HiAGM leverages the prior knowledge of label correlations regarding the predefined hierarchy and corpus. We exploit the prior probability of label dependencies as prior hierarchy knowledge.

Suppose that there is a hierarchy path $e_{i,j}$ between the parent node v_i and child node v_j. This edge feature $f(e_{i,j})$ is represented by the prior probability $P(U_j|U_i)$ and $P(U_i|U_j)$ as:

$$\begin{aligned} P(U_j|U_i) &= \frac{P(U_j \cap U_i)}{P(U_i)} = \frac{P(U_j)}{P(U_i)} = \frac{N_j}{N_i}, \\ P(U_i|U_j) &= \frac{P(U_i \cap U_j)}{P(U_j)} = \frac{P(U_j)}{P(U_j)} = 1.0, \end{aligned} \tag{1}$$

where U_k means the occurrence of v_k and $P(U_j|U_i)$ is the conditional probability of v_j given that v_i occurs. $P(U_j \cap U_i)$ is the probability of $\{v_j, v_i\}$ occurring simultaneously. N_k refers to the number of U_k in the training subset. Note that the hierarchy ensures U_k given that $v_{child(k)}$ occurs. We rescale and normalize the prior probabilities of child nodes $v_{child(k)}$ to sum total to 1.

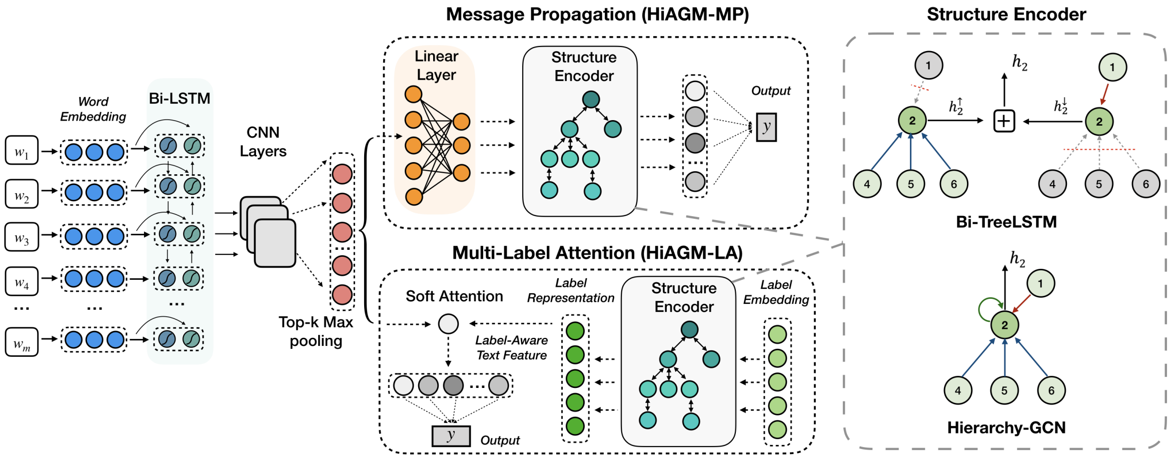

Figure 3: The overall structure of our hierarchy-aware global model. HiAGM consists of a text encoder and a hierarchy-aware encoder. The dataflows of structure encoders are illustrated in the grey dashed box. Two variants, as HiAGM-LA and HiAGM-TP, are presented in black dashed boxes, respectively.

4.2 Hierarchy-Aware Structure Encoder

Tree-LSTM and graph convolutional neural networks (GCN) are widely used as structure encoders for aggregating node information in NLP (Tai et al., 2015; Chen et al., 2017a; He et al., 2018; Rios and Kavuluru, 2018). As depicted in Figure 3, HiAGM models fine-grained hierarchy information based on the hierarchy-aware structure encoder. Based on the prior hierarchy information, we improve typical structure encoders for the directed hierarchy graph. Specifically, the top-down dataflow employs the prior hierarchy information as $f_c(e_{i,j}) = \frac{N_j}{N_i}$ while the bottom-up one adopts $f_p(e_{i,j}) = 1.0$.

Bidirectional Tree-LSTM Tree-LSTM could be utilized as our structure encoder. The implementation of Tree-LSTM is similar to syntax encoders(Tai et al., 2015; Zhang et al., 2016; Li et al., 2018). The predefined hierarchy is identical to all samples, which allows the mini-batch training method for this recursive computational module. The node transformation is as follows:

$$\begin{aligned}
\boldsymbol{i}_k &= \sigma(\boldsymbol{W}_{(i)}\,\boldsymbol{v}_k + \boldsymbol{U}_{(i)}\,\widetilde{\boldsymbol{h}}_k + \boldsymbol{b}_{(i)}),\\
\boldsymbol{f}_{k,j} &= \sigma(\boldsymbol{W}_{(f)}\,\boldsymbol{v}_k + \boldsymbol{U}_{(f)}\,\boldsymbol{h}_j + \boldsymbol{b}_{(f)}),\\
\boldsymbol{o}_k &= \sigma(\boldsymbol{W}_{(o)}\,\boldsymbol{v}_k + \boldsymbol{U}_{(o)}\,\widetilde{\boldsymbol{h}}_k + \boldsymbol{b}_{(o)}),\\
\boldsymbol{u}_k &= tanh(\boldsymbol{W}^{(u)}\,\boldsymbol{v}_k + \boldsymbol{U}^{(u)}\,\widetilde{\boldsymbol{h}}_k + \boldsymbol{b}^{(u)}),\\
\boldsymbol{c}_k &= \boldsymbol{i}_k \odot \boldsymbol{u}_k + \sum\nolimits_j \boldsymbol{f}_{k,j} \odot \boldsymbol{c}_j,\\
\boldsymbol{h}_k &= \boldsymbol{o}_k \odot tanh(\boldsymbol{c}_k),
\end{aligned} \tag{2}$$

where $\boldsymbol{h}_k$ and $\boldsymbol{c}_k$ represent the hidden state and memory cell state of node k respectively.

To induce label correlations, HiAGM employs a bidirectional Tree-LSTM by the fusion of a child-sum and a top-down module:

$$\begin{aligned}
\widetilde{\boldsymbol{h}}_k^{\uparrow} &= \sum\nolimits_{j\in child(k)} f_p(e_{k,j})\,\boldsymbol{h}_j^{\uparrow},\\
\widetilde{\boldsymbol{h}}_k^{\downarrow} &= f_c(e_{k,p})\,\boldsymbol{h}_p^{\downarrow},\\
\boldsymbol{h}_k^{bi} &= \boldsymbol{h}_k^{\uparrow} \oplus \boldsymbol{h}_k^{\downarrow},
\end{aligned} \tag{3}$$

where $\boldsymbol{h}_k^{\uparrow}$ and $\boldsymbol{h}_k^{\downarrow}$ are separately calculated in the bottom-up and top-down manner as $\boldsymbol{h}_k = \text{TreeLSTM}(\widetilde{\boldsymbol{h}_k})$. $\oplus$ indicates the concatenation of hidden states. The final hidden state of node k is the hierarchical node representation $\boldsymbol{h}_k^{bi}$.

Hierarchy-GCN GCN (Kipf and Welling, 2017) is proposed to enhance node representations based on the local graph structural information. Some NLP studies have improved Text-GCNs for rich word representations upon the syntactic structure and word correlation(Marcheggiani and Titov, 2017; Vashishth et al., 2019; Yao et al., 2019; Peng et al., 2018). We introduce a simple hierarchy-GCN for the hierarchy structure, thus gaining our aforementioned fine-grained hierarchy information.

Hierarchy-GCN aggregates dataflows within the top-down, bottom-up, and self-loop edges. In the hierarchy graph, each directed edge represents a pair-wise label correlation feature. Thus, those dataflows should conduct node transformations with edge-wise linear transformations. However, edge-wise transformations shall lead to over-parameterized edge-wise weight matrixes. Our Hierarchy-GCN simplifies this transformation with a weighted adjacent matrix. This weighted adjacent matrix represents the hierarchical prior probability. Formally, Hierarchy-GCN encodes the hidden state of node k based on its associated neighbourhood $N(k) = \{n_k, child(k), parent(k)\}$ as:

$$\begin{aligned} \boldsymbol{u}_{k,j} &= a_{k,j}\boldsymbol{v}_j + \boldsymbol{b}_l^k, \\ \boldsymbol{g}_{k,j} &= \sigma(\boldsymbol{W}_g^{d(j,k)}\boldsymbol{v}_k + \boldsymbol{b}_g^k), \\ \boldsymbol{h}_k &= \mathrm{ReLU}(\sum\nolimits_{j\in N(k)} \boldsymbol{g}_{k,j} \odot \boldsymbol{u}_{k,j}), \end{aligned} \tag{4}$$

where $\boldsymbol{W}_g^{d(k,j)} \in \mathbb{R}^{dim}$, $\boldsymbol{b}_l \in \mathbb{R}^{N\times dim}$, and $\boldsymbol{b}_g \in \mathbb{R}^N$. $d(j,k)$ indicates the hierarchical direction from node j to node k, including top-down, bottom-up, and self-loop edges. Note that $a_{k,j} \in \mathbb{R}$ denotes the hierarchy probability $f_{d(k,j)}(e_{kj})$, where the self-loop edge employs $a_{k,k} = 1$, top-down edges use $f_c(e_{j,k}) = \frac{N_k}{N_j}$, and bottom-up edges use $f_p(e_{j,k}) = 1$. The holistic edge feature matrix $\boldsymbol{F} = \{a_{0,0}, a_{0,1}, \ldots, a_{C-1,C-1}\}$ indicates the weighted adjacent matrix of the directed hierarchy graph. Finally, the output hidden state $\boldsymbol{h}_k$ of node k denotes its label representation corresponding to the hierarchy structural information.

4.3 Hybrid Information Aggregation

Previous global models classify labels upon the original textual information and improve the decoder with predefined hierarchy paths. In contrast, we construct a novel end-to-end hierarchy-aware global model (HiAGM) for the mutual interaction of text features and label correlations. It combines a traditional text classification model with a hierarchy encoder, thus obtaining label-wise text features. HiAGM is extended to two variants, a parallel model for an inductive fusion (HiAGM-LA) and a serial model for a deductive fusion (HiAGM-TP).

Given a document $x = (\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_s)$, the sequence of token embedding is firstly fed into a bidirectional GRU layer to extract text contextual feature. Then, multiple CNNs are used for generating n-gram features. The concatenation of n-gram features is filtered by a top-k max-pooling layer to extract key information. Finally, by reshaping, we can obtain the continuous text representation $S = (\boldsymbol{s}_1, \ldots, \boldsymbol{s}_n)$ where $\boldsymbol{s}_i \in \mathbb{R}^{d_c}$ and d_c indicates the output dimension of the CNN layer. $n = n_k \times n_c$ refers to the multiplication of top-k number and the number of CNNs.

Hierarchy-Aware Multi-Label Attention The first variant of HiAGM is proposed based on multi-label attention, called as HiAGM-LA. Attention mechanism is usually utilized as the memory unit in text classification (Yang et al., 2016; Du et al., 2019). Recent LCL studies (Huang et al., 2019; You et al., 2019) construct one multi-label attention-based model per level so as to avoid optimizing label embedding among different levels.

Our HiAGM-LA is similar to those baselines but simplifies multi-label attention LCL models to a global model. Based on our hierarchy encoders, HiAGM-LA could overcome the problem of convergence for label embedding across various levels. Label representations are enhanced with bidirectional hierarchical information. This local structural information makes it feasible to learn label features across different levels in a single model. Formally, suppose that the trainable label embedding of node k is randomly initialized as $\boldsymbol{L}_k \in \mathbb{R}^{d_l}$. The initial label embedding $\boldsymbol{L}_k$ is directly fed into structure encoders as the input vector of aligned label node $\boldsymbol{x}_k$. Then, the output hidden state $\boldsymbol{h} \in \mathbb{R}^{C\times d_c}$ represents as the hierarchy-aware label features. Given text representation $\boldsymbol{S} \in \mathbb{R}^{n\times d_c}$, HiAGM-LA calculates the label-wise attention value α_{ki} as:

$$\alpha_{kj} = \frac{e^{\boldsymbol{s}_j \boldsymbol{h}_k^T}}{\sum_{j=1}^{n} e^{\boldsymbol{s}_j \boldsymbol{h}_k^T}}, \boldsymbol{v}_k = \sum_{i=1}^{n} \alpha_{ki}\, \boldsymbol{s}_i, \tag{5}$$

Note that α_{ki} indicates how informative the i-th text feature vector is for the k-th label. We can get the inductive label-aligned text features $\boldsymbol{V} \in \mathbb{R}^{C\times d_c}$ based on multi-label attention. Then it would be fed into the classifier for prediction. Furthermore, we could directly use the hidden state of hierarchy encoders as the pretrained label representations so that HiAGM-LA could be even lighter in the inference process.

Hierarchical text feature propagation Graph neural networks are capable of message passing (Gilmer et al., 2017; Duvenaud et al., 2015), learning both local node correlations and overall graph structure. To avoid the noise from heterogeneous fusion, the second variant obtains label-wise text features based on a deductive method. It directly takes text features $\boldsymbol{S}$ as the node inputs and updates textual information through the hierarchy-aware structure encoder. This variant mainly conducts the propagation of text features, called as HiAGM-TP. Formally, node inputs $\boldsymbol{V}$ are reshaped from text features by a single linear transformation:

$$\boldsymbol{V} = \boldsymbol{M}\,\boldsymbol{S}, \tag{6}$$

where the trainable weight matrix $\boldsymbol{M} \in \mathbb{R}^{(n \times d_c) \times (C \times d_v)}$ transforms text features $\boldsymbol{S} \in \mathbb{R}^{n \times d_c}$ to node inputs $\boldsymbol{V} \in \mathbb{R}^{C \times d_v}$.

Given the predefined structure, each sample would update its textual information throughout the same holistic taxonomic hierarchy. In a mini-batch learning manner, the initial node representation $\boldsymbol{V}$ is fed into the hierarchy encoder. The output hidden state $\boldsymbol{h}$ denotes deductive hierarchy-aware text features as the input of the final classifier. Compared with HiAGM-LA, the transformation of HiAGM-TP is conducted on textual information without the fusion of label embedding. Thus, the structure encoder would be activated in both training and inference procedures for passing textual messages across the hierarchy. It could converge much easier but has slightly higher computational complexity than HiAGM-LA.

4.4 Classification

We flatten the hierarchy by taking all nodes as leaf nodes for multi-label classification, no matter it is a leaf node or an internal node. The final hierarchy-aware features are fed into a fully connected layer for prediction. HiAGM is complementary with recursive regularization(Gopal and Yang, 2013) as $L_r = \sum_{i \in C} \sum_{j \in child(i)} \frac{1}{2} ||\boldsymbol{w}_i - \boldsymbol{w}_j||^2$ for the parameters of the final fully connected layer. For multi-label classification, HiAGM uses a binary cross-entropy loss function:$L_c = -\sum_{i=1}^{N} \sum_{j=1}^{C} [y_{ij} log(y'_{ij}) + (1 - y_{ij}) log(1 - y'_{ij})]$ where y_{ij} and y'_{ij} are the ground truth and sigmoid score for the j-th label of the i-th sample. Thus, the final loss function is $L_m = L_c + \lambda \cdot L_r$.

5 Experiment

In this section, we introduce our experiments with datasets, evaluation metrics, implementation details, comparison, ablation study, and analysis of experimental results.

5.1 Experiment Setup

We experiment our proposed architecture on RCV1-V2, Web-of-Science (WOS) and NYTimes (NYT) datasets for comparison and ablation study.

Datasets RCV1-V2 (Lewis et al., 2004) and NYT (Sandhaus, 2008) are both news categorization corpora while WOS (Kowsari et al., 2017) includes abstracts of published papers from Web of Science. Those typical text classification datasets are all annotated with the ground truth of hierarchical taxonomic labels. We use the benchmark split of RCV1-V2 and select a small partial training subset for validation. WOS dataset is randomly splitted into training, validation and test subsets. In NYT, we randomly select and split subsets from original raw data. We also remove samples with no label or only a single one-level label. Note that WOS is for single-path HTC while NYT and RCV1-V2 include multi-path taxonomic tags. The statistics of datasets is shown in Table 1.

Dataset	$\|L\|$	Depth	Avg($\|L_i\|$)	Train	Val	Test
RCV1	103	4	3.24	20,833	2,316	781,265
WOS	141	2	2.0	30,070	7,518	9,397
NYT	166	8	7.6	23,345	5,834	7,292

Table 1: Data Statistics: $|L|$ is the number of classes. Avg($|L_i|$) is the average number of classes per sample. Depth indicates the maximum level of hierarchy.

Evaluation Metrics We measure the experimental results with standard evaluation metrics (Gopal and Yang, 2013), including Micro-F1 and Macro-F1. Micro-F1 takes the overall precision and recall of all the instances into account while Macro-F1 equals to the average F1-score of labels. So Micro-F1 gives more weight to frequent labels, while Macro-F1 equally weights all labels.

Implementation Details We use a one-layer bi-GRU with 64 hidden units and 3 parallel CNN layers with filter region size of $\{2, 3, 4\}$. The vocabulary is created by the most frequent words with the maximum size of 60,000. We use 300-dimensional pretrained word embedding from GloVe[2] (Pennington et al., 2014) and randomly initialize the out-of-vocabulary words above the minimum count of 2. The key information pertaining to text classification could be extracted from the beginning statements. Thus, we set the maximum length of token inputs as 256. The fixed threshold for tagging is chosen as 0.5. Dropout is employed in the embedding layer and MLP layer with the rate of 0.5 while in the bi-GRU layer and node transformation with the rate of 0.1 and 0.05 respectively. Additionally, for HiAGM-LA, the label embedding is initialized by Kaiming uniform (He et al., 2015) while the other model parameters are initialized by Xavier uniform (Glorot and Bengio, 2010). We use the Adam optimizer in a mini-batch size of 64 with learning rate

[2] https://nlp.stanford.edu/projects/glove

Model	Micro	Macro
Local Models		
HR-DGCNN-3 (Peng et al., 2018)	76.18	43.34
HMCN (Mao et al., 2019)	80.80	54.60
HFT(M) (Shimura et al., 2018)	80.29	51.40
Htrans (Banerjee et al., 2019)	80.51	58.49
Global Models		
SGM [4] (Yang et al., 2018)	77.30	47.49
HE-AGCRCNN (Peng et al., 2019)	77.80	51.30
HiLAP-RL (Mao et al., 2019)	83.30	60.10
Baselines		
TextRCNN	81.57	59.25
TextRCNN+LabelAttention	81.88	59.85
HiAGM-LA		
TreeLSTM	$82.54^{\dagger\ddagger}$	$61.90^{\dagger\ddagger}$
GCN	$82.21^{\dagger\ddagger}$	$61.65^{\dagger\ddagger}$
GCN w/o Rec	$82.26^{\dagger\ddagger}$	$61.85^{\dagger\ddagger}$
HiAGM-TP		
TreeLSTM	$83.20^{\dagger}$	$62.32^{\dagger}$
GCN	**83.96**†	**63.35**†
GCN w/o Rec	$83.95^{\dagger}$	$63.23^{\dagger}$

Table 2: Comparison to previous models on RCV1-V2. Note that the prior probability matrix in HiAGM-TP is fine-tuned during training while the one in HiAGM-LA is fixed. *w/o Rec* denotes training without recursive regularization. "†" and "‡" indicate statistically significant difference ($p<0.01$) from TextRCNN and TextRCNN+LabelAttention respectively.

$\alpha = 1 \times 10^{-4}$, momentum parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1 \times 10^{-6}$. The penalty coefficient of recursive regularization is set as 1×10^{-6}. Our model evaluates the test subset with the best model on the validation subset.

5.2 Comparison

In Table 2, we compare the performance of HiAGM to traditional MLC models and the state-of-the-art HTC studies on RCV1-V2. With the recursive regularization for the last MLP layer, those conventional text classification models also obtain competitive performance. As for our proposed architecture, both HiAGM-LA and HiAGM-TP outperform most state-of-the-art results of global and local studies, esspecially in Macro-F1. It shows the strong advancement of our hierarchy encoders on HTC. HiAGM-LA achieves the performance of 61.90% Macro-F1 score and 82.54% Micro-F1 score while HiAGM-TP obtains the best performance of 63.35% Macro-F1 score and 83.96% Micro-F1 score.

Model	HiAGM-LA			HiAGM-TP		
	Micro	Macro	Time	Micro	Macro	Time
TreeLSTM	82.54	61.90	1.0 ×	83.24	62.60	3.2×
GCN	82.21	61.65	**0.9**×	**83.92**	**63.01**	1.1×

Table 3: Comparison of the HiAGM variants on RCV1-V2 with fixed prior probability. Note that *Time* denotes the time cost of one epoch during inference compared to TreeLSTM-based HiAGM-LA. Statistically significant difference ($p<0.01$) compared to the best one.

To clarify the improvement of our proposed HiAGM, we also experiment without recursive regularization. Compared with the state-of-the-art recent work (HiLAP) (Mao et al., 2019), our HiAGM-LA and HiAGM-TP without recursive regularization also achieve competitive improvement by 1.75% and 3.13% in terms of Macro-F1. It demonstrates that the recursive regularization is complementary but not necessary with our proposed architecture.

According to Table 4, HiAGM achieves consistent improvement on the performance of HTC among RCV1-V2, WOS and NYT datasets. It indicates the strong improvement of the label-wise text feature on HTC task. The results present that our proposed global model HiAGM has the advanced capability of enhancing text features for HTC.

All in all, HiAGM strongly improves the performance on the benchmark dataset RCV1-V2 and the other two classical text classification datasets. Especially, it obtains better results on Macro-F1 score. It indicates that HiAGM has a strong ability to tackle data-sparse classes deep in the hierarchy.

5.3 Analysis

Hybrid Information Aggregation According to Table 2, both variants outperform the baseline models and previous studies. It denotes that the enhanced text feature is beneficial for HTC. We clarify the ablation study of two variants and structure encoders in Table 3. Both HiAGM-LA and HiAGM-TP are trained with fixed prior probability. With the help of the recursive computation process, bidirectional Tree-LSTM achieves better performance on learning hierarchy-aware label embedding. However, it additionally leads to lower computational efficiency when compared to Hierarchy-GCN. Regarding HiAGM-TP, hierarchy-GCN shows its better performance and efficiency than bidirectional Tree-LSTM.

These two variants have various advantages, respectively. To be specific, HiAGM-TP has better performance than HiAGM-LA in both Bi-

[4] The result is reproduced with benchmark split upon the released project of SGM.

Model	RCV1-V2		RCV1-V2-R		WOS		NYT	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
Global Text Classification Baseline								
TextRNN	81.10	51.09	87.78	70.42	77.94	69.65	70.29	53.06
TextCNN	79.37	55.45	84.97	68.06	82.00	76.18	70.11	56.84
TextRCNN	81.57	59.25	88.32	72.23	83.55	76.99	70.83	56.18
HiAGM-LA								
GCN	82.21	61.65	88.49	73.14	84.61	79.37	72.35	58.67
TreeLSTM	82.54	61.90	88.47	72.81	84.82	79.51	72.50	58.86
HiAGM-TP								
GCN	**83.96**	**63.35**	88.64	74.00	**85.82**	**80.28**	**74.97**	**60.83**
TreeLSTM	83.20	62.32	**88.86**	**74.16**	85.18	79.95	74.43	60.76

Table 4: Experimental results of our proposed HiAGM-LA and HiAGM-TP on various datasets. Note that RCV1-V2-R refers to the version that transpose original subset of the train and test set. All models are trained with the constraint of recursive regularization. HiAGM-LA is trained with fixed prior probability while HiAGM-TP is trained with trainable one.

TreeLSTM and Hierarchy-GCN encoders. The multi-label attention variant, HiAGM-LA, would somehow induce noises from the randomly initialized label embedding. Otherwise, HiAGM-TP aggregates the fusion of local structural information and text feature maps, without the negative impact of label embedding.

As for efficiency, HiAGM-LA is more computationally efficient than HiAGM-TP, especially in the inference process. The label representation from hierarchy encoders could be utilized as pretrained label embedding for multi-label attention during inference. Thus, HiAGM-LA omits the hierarchy-aware structure encoder module after training.

We recommend HiAGM-TP for high performance while we also suggest HiAGM-LA for empirically good performance and faster inference.

GCN Layers The impact of GCN layers is also an important issue for HiAGM. As illustrated in Figure 4, the one-layer structure encoder consistently performs best in both HiAGM-LA and HiAGM-TP. It indicates that the correlation between non-adjacent nodes is not essential for HTC but somehow noisy for hierarchical information aggregation. This empirical conclusion is consistent with the implementation of recursive regularization (Peng et al., 2018; Gopal and Yang, 2013)and transfer learning (Banerjee et al., 2019; Shimura et al., 2018) between adjacent labels or levels.

Prior Probability According to the aforementioned comparisons, our simplified structure encoders with prior probabilities is undoubtedly beneficial for HTC. We also investigate different choices of prior probabilities with hierarchy-GCN encoder on the HiAGM-TP variant, clarified as Table 5. Note that the weighted adjacent matrix is initialized by prior probabilities.

The simple weighted adjacent matrix performs better than the complex edge-wise weight matrix for node transformation. The fixed weighted adjacent matrix also achieves better results than the original unweighted adjacent matrix and the trainable randomly initialized one. It demonstrates that the prior probability of the hierarchy is capable of representing hierarchical label dependencies. Furthermore, the best result is obtained by the setting that obeys the calculating direction of prior probability. When comparing the results of the fixed adjacent matrix and trainable one, we can find that the weighted adjacent matrix could be finetuned for higher flexibility and better performance.

In Table 5, the settings that allows all interac-

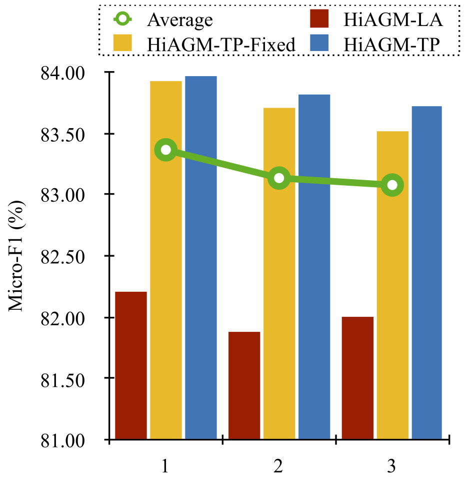

Figure 4: Ablation study on the depth of GCN.

Top-Down	Bottom-Up	Fixed		Trainable	
		Micro	Macro	Micro	Macro
Edge-Wise Matrix		-	-	82.75	60.81
Randomly Initialized		-	-	83.86	62.12
Randomly Initialized*		-	-	82.80	62.51
1	1	83.77	62.31	83.86	62.96
P	P	83.61	**63.65**	83.83	63.14
1	P	83.65	62.46	83.95	63.23
P	1	**83.92**	63.01	**83.96**	**63.35**
P*	1*	-	-	83.33	62.86

Table 5: Ablation study of the fine-grained hierarchy information on RCV1-V2 based on GCN-based HiAGM-TP. *Edge-Wise Matrix* denotes that each directional edge has a distinct trainable weight matrix for transformation while the others use the weighted adjacent matrix. *P* is $f_c(e_{i,j}) = \frac{N_j}{N_i}$ and *1* is $f_p(e_{i,j}) = 1.0$. "*" allows the information propagation between all nodes while the others obey the constraint of hierarchy.

tions perform worse than the others that allow propagation throughout the hierarchy paths. As analyzed on GCN layers, the interaction between non-adjacent nodes would lead to negative impact on the HTC. We also validate this conclusion based on the ablation study of prior probability.

Performance Study We analyze the improvement on performance by dividing labels based on their levels. We compute level-based Micro-F1 scores of NYT on baseline, HiAGM-LA, and HiAGM-TP. Figure 5 shows that our models retain a better performance than the baseline on all levels, especially among deep levels.

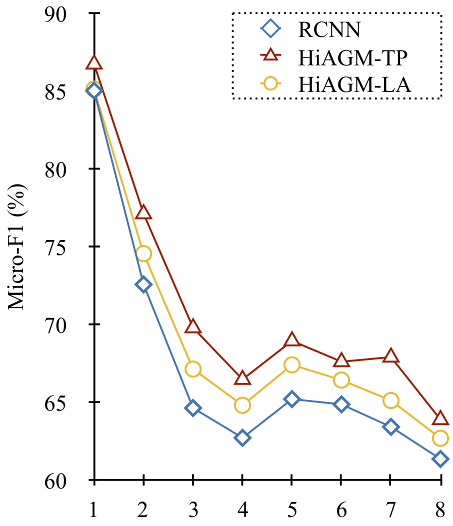

Figure 5: Evaluation of labels among different levels. Note that we observe similar results for other datasets and omit them for a cleaner view.

6 Conclusion

In this paper, we propose a novel end-to-end hierarchy-aware global model that extracts the label structural information for aggregating label-wise text features. We present a bidirectional TreeLSTM and a hierarchy-GCN as the hierarchy-aware structure encoder. Furthermore, our framework is extended into a parallel variant based on multi-label attention and a serial variant of text feature propagation. Our approaches empirically achieve significant and consistent improvement on three distinct datasets, especially on the low-frequency labels. Specifically, both variants outperform the state-of-the-art model on the RCV1-V2 benchmark dataset. And our best model obtains a Macro-F1 score of 63.35% and a Micro-F1 score of 83.96%.

Acknowledgments

We thank all the anonymous reviewers for their valuable suggestions. This research work was supported by the National Natural Science Foundation of China (Grant No.61772337, U1736207).

References

Siddhartha Banerjee, Cem Akkaya, Francisco Perez-Sorrosal, and Kostas Tsioutsiouliklis. 2019. Hierarchical transfer learning for multi-label text classification. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6295–6300. Association for Computational Linguistics.

Lijuan Cai and Thomas Hofmann. 2004. Hierarchical document categorization with support vector machines. In *Proceedings of the 2004 ACM CIKM International Conference on Information and Knowledge Management, Washington, DC, USA, November 8-13, 2004*, pages 78–87. ACM.

Huadong Chen, Shujian Huang, David Chiang, and Jiajun Chen. 2017a. Improved neural machine translation with a syntax-aware encoder and decoder. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1936–1945. Association for Computational Linguistics.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017b. Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1657–1668. Association for Computational Linguistics.

Cunxiao Du, Zhaozheng Chen, Fuli Feng, Lei Zhu, Tian Gan, and Liqiang Nie. 2019. Explicit interaction model towards text classification. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6359–6366. AAAI Press.

Susan T. Dumais and Hao Chen. 2000. Hierarchical classification of web content. In *SIGIR 2000: Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, July 24-28, 2000, Athens, Greece*, pages 256–263. ACM.

David Duvenaud, Dougal Maclaurin, Jorge Aguilera-Iparraguirre, Rafael Gómez-Bombarelli, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P. Adams. 2015. Convolutional networks on graphs for learning molecular fingerprints. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 2224–2232.

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. 2017. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1263–1272. PMLR.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, volume 9 of *JMLR Proceedings*, pages 249–256. JMLR.org.

Siddharth Gopal and Yiming Yang. 2013. Recursive regularization for large-scale classification with hierarchical and graphical dependencies. In *The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2013, Chicago, IL, USA, August 11-14, 2013*, pages 257–265. ACM.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pages 1026–1034. IEEE Computer Society.

Shexia He, Zuchao Li, Hai Zhao, and Hongxiao Bai. 2018. Syntax for semantic role labeling, to be, or not to be. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2061–2071. Association for Computational Linguistics.

Wei Huang, Enhong Chen, Qi Liu, Yuying Chen, Zai Huang, Yang Liu, Zhou Zhao, Dan Zhang, and Shijin Wang. 2019. Hierarchical multi-label text classification: An attention-based recurrent network approach. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM 2019, Beijing, China, November 3-7, 2019*, pages 1051–1060. ACM.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Kamran Kowsari, Donald E. Brown, Mojtaba Heidarysafa, Kiana Jafari Meimandi, Matthew S. Gerber, and Laura E. Barnes. 2017. Hdltex: Hierarchical deep learning for text classification. In *16th IEEE International Conference on Machine Learning and Applications, ICMLA 2017, Cancun, Mexico, December 18-21, 2017*, pages 364–371. IEEE.

David D. Lewis, Yiming Yang, Tony G. Rose, and Fan Li. 2004. RCV1: A new benchmark collection for text categorization research. *J. Mach. Learn. Res.*, 5:361–397.

Zuchao Li, Shexia He, Jiaxun Cai, Zhuosheng Zhang, Hai Zhao, Gongshen Liu, Linlin Li, and Luo Si. 2018. A unified syntax-aware framework for semantic role labeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2401–2411. Association for Computational Linguistics.

Xiaodong Liu, Jianfeng Gao, Xiaodong He, Li Deng, Kevin Duh, and Ye-Yi Wang. 2015. Representation learning using multi-task deep neural networks for semantic classification and information retrieval. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 912–921. The Association for Computational Linguistics.

Yuning Mao, Jingjing Tian, Jiawei Han, and Xiang Ren. 2019. Hierarchical text classification with reinforced label assignment. 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 2019, Hong Kong, China*, pages 445–455. Association for Computational Linguistics.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. In *Proceedings of the 2017*

Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017, pages 1506–1515. Association for Computational Linguistics.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. 2018. Explainable prediction of medical codes from clinical text. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 1101–1111. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2007. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135.

Hao Peng, Jianxin Li, Qiran Gong, Senzhang Wang, Lifang He, Bo Li, Lihong Wang, and Philip S. Yu. 2019. Hierarchical taxonomy-aware and attentional graph capsule rcnns for large-scale multi-label text classification. *CoRR*, abs/1906.04898.

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 on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*, pages 1063–1072. ACM.

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 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543. ACL.

Anthony Rios and Ramakanth Kavuluru. 2018. Few-shot and zero-shot multi-label learning for structured label spaces. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3132–3142. Association for Computational Linguistics.

Evan Sandhaus. 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.

Kazuya Shimura, Jiyi Li, and Fumiyo Fukumoto. 2018. HFT-CNN: learning hierarchical category structure for multi-label short text categorization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 811–816. Association for Computational Linguistics.

Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1556–1566. The Association for Computer Linguistics.

Shikhar Vashishth, Manik Bhandari, Prateek Yadav, Piyush Rai, Chiranjib Bhattacharyya, and Partha P. Talukdar. 2019. Incorporating syntactic and semantic information in word embeddings using graph convolutional networks. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3308–3318. Association for Computational Linguistics.

Jonatas Wehrmann, Ricardo Cerri, and Rodrigo C. Barros. 2018. Hierarchical multi-label classification networks. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 5225–5234. PMLR.

Jiawei Wu, Wenhan Xiong, and William Yang Wang. 2019. Learning to learn and predict: A meta-learning approach for multi-label 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 2019, Hong Kong, China*, pages 4353–4363. Association for Computational Linguistics.

Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018. SGM: sequence generation model for multi-label classification. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 3915–3926. Association for Computational Linguistics.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alexander J. Smola, and Eduard H. Hovy. 2016. Hierarchical attention networks for document classification. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 1480–1489. The Association for Computational Linguistics.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Graph convolutional networks for text classification. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI*

2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019, pages 7370–7377. AAAI Press.

Ronghui You, Zihan Zhang, Ziye Wang, Suyang Dai, Hiroshi Mamitsuka, and Shanfeng Zhu. 2019. Attentionxml: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification. In *Advances in Neural Information Processing Systems*, pages 5812–5822.

Xingxing Zhang, Liang Lu, and Mirella Lapata. 2016. Top-down tree long short-term memory networks. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 310–320. The Association for Computational Linguistics.