Hierarchy-aware Label Semantics Matching Network for Hierarchical Text Classification

Haibin Chen, Qianli Ma*, Zhenxi Lin, Jiangyue Yan
School of Computer Science and Engineering,
South China University of Technology, Guangzhou, China
haibin_chen@foxmail.com
qianlima@scut.edu.cn*

Abstract

Hierarchical text classification is an important yet challenging task due to the complex structure of the label hierarchy. Existing methods ignore the semantic relationship between text and labels, so they cannot make full use of the hierarchical information. To this end, we formulate the text-label semantics relationship as a semantic matching problem and thus propose a hierarchy-aware label semantics matching network (HiMatch). First, we project text semantics and label semantics into a joint embedding space. We then introduce a joint embedding loss and a matching learning loss to model the matching relationship between the text semantics and the label semantics. Our model captures the text-label semantics matching relationship among coarse-grained labels and fine-grained labels in a hierarchy-aware manner. The experimental results on various benchmark datasets verify that our model achieves state-of-the-art results.

1 Introduction

Hierarchical text classification (HTC) is widely used in Natural Language Processing (NLP), such as news categorization (Lewis et al., 2004) and scientific paper classification (Kowsari et al., 2017). HTC is a particular multi-label text classification problem, which introduces hierarchies to organize label structure. As depicted in Figure 1, HTC models predict multiple labels in a given label hierarchy, which generally construct one or multiple paths from coarse-grained labels to fine-grained labels in a top-down manner (Aixin Sun and Ee-Peng Lim, 2001). Generally speaking, fine-grained labels are the most appropriate labels for describing the input text. Coarse-grained labels are generally the parent nodes of coarse- or fine-grained labels, expressing a more general concept. The key challenges of HTC are to model the large-scale, imbalanced, and structured label hierarchy (Mao et al., 2019).

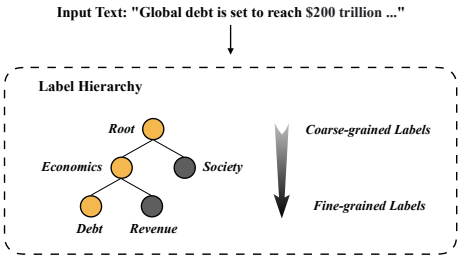

Figure 1: An hierarchical text classification example tagged with labels *Economics* and *Debt* from coarse-grained label to fine-grained label.

Existing work in HTC has introduced various methods to use hierarchical information in a holistic way. To capture the holistic label correlation features, some researchers proposed a hierarchy-aware global model to exploit the prior probability of label dependencies through Graph Convolution Networks (GCN) and TreeLSTM (Zhou et al., 2020). Some researchers also introduced more label correlation features such as label semantic similarity and label co-occurrence (Lu et al., 2020). They followed the traditional way to transform HTC into multiple binary classifiers for every label (Fürnkranz et al., 2008). However, they ignored the interaction between text semantics and label semantics (Fürnkranz et al., 2008; Wang et al., 2019), which is highly useful for classification (Chen et al., 2020). Hence, their models may not be sufficient to model complex label dependencies and provide comparable text-label classification scores (Wang et al., 2019).

A natural strategy for modeling the interaction between text semantics and label semantics is to introduce a text-label joint embedding by label attention (Xiao et al., 2019) or autoencoders (Yeh et al., 2017). Label attention-based methods adopted a

*Corresponding author

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 4370–4379
August 1–6, 2021.

self-attention mechanism to identify label-specific information (Xiao et al., 2019). Autoencoder-based methods extended the vanilla Canonical Correlated Autoencoder (Yeh et al., 2017) to a ranking-based autoencoder architecture to produce comparable text-label scores (Wang et al., 2019). However, these methods assume all the labels are independent without fully considering the correlation between coarse-grained labels and fine-grained labels, which cannot be simply transferred to HTC models (Zhou et al., 2020).

In this paper, we formulate the interaction between text and label as a semantic matching problem and propose a **Hi**erarchy-aware Label Semantics **Match**ing Network (HiMatch). The principal idea is that the text representations should be semantically similar to the target label representations (especially fine-grained labels), while they should be semantically far away from the incorrect label representations. First, we adopt a text encoder and a label encoder (shown in Figure 2) to extract textual semantics and label semantics, respectively. Second, inspired by the methods of learning common embeddings (Wang et al., 2019), we project both textual semantics and label semantics into a text-label joint embedding space where correlations between text and labels are exploited. In this joint embedding space, we introduce a joint embedding loss between text semantics and target label semantics to learn a text-label joint embedding. After that, we apply a matching learning loss to capture text-label matching relationships in a hierarchy-aware manner. In this way, the fine-grained labels are semantically closest to the text semantics, followed by the coarse-grained labels, while the incorrect labels should be semantically far away from the text semantics. Hence, we propose a hierarchy-aware matching learning method to capture different matching relationships through different penalty margins on semantic distances. Finally, we employ the textual representations guided by the joint embedding loss and matching learning loss to perform the hierarchical text classification.

The major contributions of this paper are:

1. By considering the text-label semantics matching relationship, we are the first to formulate HTC as a semantic matching problem rather than merely multiple binary classification tasks.

2. We propose a hierarchy-aware label semantics matching network (HiMatch), in which we introduce a joint embedding loss and a matching learning loss to learn the text-label semantics matching relationship in a hierarchy-aware manner.

3. Extensive experiments (with/without BERT) on various datasets show that our model achieves state-of-the-art results.

2 Related Work

2.1 Hierarchical Text Classification

Hierarchical text classification is a particular multi-label text classification problem, where the classification results are assigned to one or more nodes of a taxonomic hierarchy. Existing state-of-the-art methods focus on encoding hierarchy constraint in a global view such as directed graph and tree structure. Zhou et al. (2020) proposed a hierarchy-aware global model to exploit the prior probability of label dependencies. Lu et al. (2020) introduced three kinds of label knowledge graphs, i.e., taxonomy graph, semantic similarity graph, and co-occurrence graph to benefit hierarchical text classification. They regarded hierarchical text classification as multiple binary classification tasks (Fürnkranz et al., 2008). The limitation is that these models did not consider the interaction of label semantics and text semantics. Therefore, they failed to capture complex label dependencies and can not provide comparable text-label classification scores (Wang et al., 2019), which leads to restricted performance (Chen et al., 2020). Hence, it is crucial to exploit the relationship between text and label semantics, and help the model distinguish target labels from incorrect labels in a comparable and hierarchy-aware manner. We perform matching learning in a joint embedding of text and label to solve these problems in this work.

2.2 Exploit Joint Embedding of Text and Label

To determine the correlation between text and label, researchers proposed various methods to exploit a text-label joint embedding such as (Xiao et al., 2019) or Autoencoder (Yeh et al., 2017). In the field of multi-label text classification, Xiao et al. (2019) proposed a Label-Specific Attention Network (LSAN) to learn a text-label joint embedding by label semantic and document semantic. Wang et al. (2019) extended vanilla Canonical Correlated AutoEncoder (Yeh et al., 2017) to a ranking-based autoencoder architecture to produce comparable label scores. However, they did not fully consider label semantics and holistic label correlation

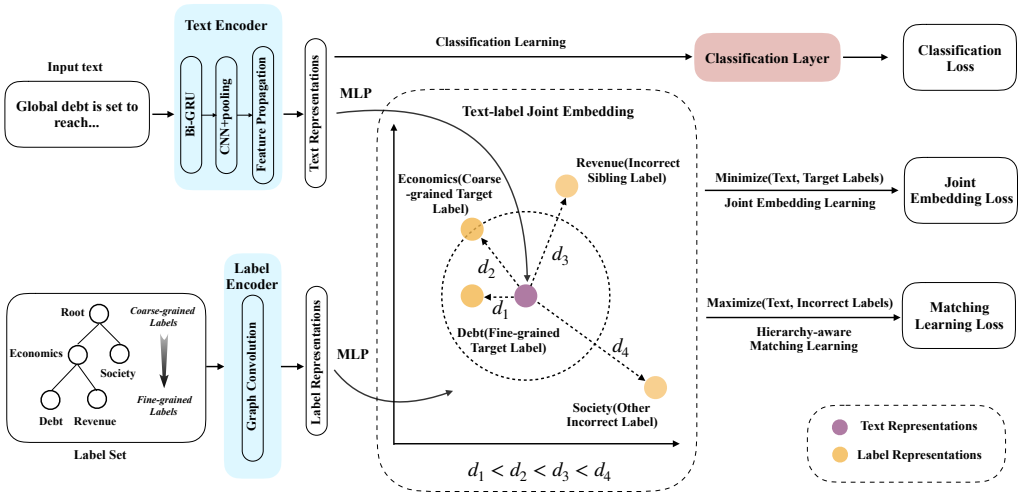

Figure 2: The overall architecture of the proposed model. Firstly, the text encoder and label encoder extract the text semantics and label semantics, respectively. Then text semantics and label semantics are projected into a joint embedding space. Joint embedding loss encourages the text semantics to be similar to the target label semantics. By introducing matching learning loss, fine-grained labels semantics (Debt) is semantically closest to the text semantics, followed by coarse-grained labels (Economics), while other incorrect labels semantics is semantically far away from text semantics (Revenue, Society). The relative order is $d_1 < d_2 < d_3 < d_4$, where d represents the metric distances in joint embedding.

among fine-grained labels, coarse-grained labels, and incorrect labels. In addition, we can not simply transfer these multi-label classification methods to HTC due to the constraint of hierarchy (Zhou et al., 2020).

3 Proposed Method

In this section, we will describe the details about our Hierarchy-aware Label Semantics Matching Network. Figure 2 shows the overall architecture of our proposed model.

3.1 Text Encoder

In the HTC task, given the input sequence $x_{seq} = \{x_1, ..., x_n\}$, the model will predict the label $y = \{y_1, ..., y_k\}$ where n is the number of words and k is the number of label sets. The label with a probability higher than a fixed threshold (0.5) will be regarded as the prediction result. The sequence of token embeddings is firstly fed into a bidirectional GRU layer to extract contextual feature $H = \{h_1, ..., h_n\}$. Then, CNN layers with top-k max-pooling are adopted for generating key n-gram features $T \in \mathbb{R}^{k \times d_{cnn}}$ where d_{cnn} indicates the output dimension of the CNN layer.

Following the previous work (Zhou et al., 2020), we further introduce a hierarchy-aware text feature propagation module to encode label hierarchy information. We define a hierarchy label structure as a directed graph $G = \left(V_t, \overleftarrow{E}, \overrightarrow{E}\right)$, where V_t indicates the set of hierarchy structure nodes. $\overleftarrow{E}$ are built from the top-down hierarchy paths representing the prior statistical probability from parent nodes to children nodes. $\overrightarrow{E}$ are built from the bottom-up hierarchy paths representing the connection relationship from children nodes to parent nodes. The feature size of graph adjacency matrix $\leftarrow E$ and $\rightarrow E$ is $\in R^{k \times k}$, where k is the number of label sets. Text feature propagation module firstly projects text features T to node inputs V_t by a linear transformation $W_{proj} \in \mathbb{R}^{k \times d_{cnn} \times d_t}$, where d_t represents the hierarchy structure node dimension from text feature. Then a Graph Convolution Network (GCN) is adopted to explicitly combine text semantics with prior hierarchical information $\overleftarrow{E}$ and $\overrightarrow{E}$:

$$S_t = \sigma\left(\overleftarrow{E} \cdot V_t \cdot W_{g_1} + \overrightarrow{E} \cdot V_t \cdot W_{g_2}\right) \quad (1)$$

where σ is the activation function ReLU. $W_{g1}, W_{g2} \in \mathbb{R}^{d_t \times d_t}$ are the weight matrix of GCN. S_t is the text representation aware of prior hierarchy paths.

3.2 Label Encoder

In the HTC task, the hierarchical label structure can be regarded as a directed graph $G = \left(V_l, \overleftarrow{E}, \overrightarrow{E}\right)$,

where V_l indicates the set of hierarchy structure nodes with label representation. The graph G in label encoder shares the same structure $\overleftarrow{E}$ and $\overrightarrow{E}$ with the graph in text encoder. Given the total label set $y = \{y_1, ..., y_k\}$ as input, we create label embeddings $V_l \in \mathbb{R}^{d_l}$ by averaging of pre-trained label embeddings first. Then GCN could be utilized as label encoder:

$$S_l = \sigma\left(\overleftarrow{E} \cdot V_l \cdot W_{g3} + \overrightarrow{E} \cdot V_l \cdot W_{g4}\right) \quad (2)$$

where σ is the activation function ReLU. $W_{g3}, W_{g4} \in \mathbb{R}^{d_l \times d_l}$ are the weight matrix of GCN. S_l is the label representation aware of prior hierarchy paths. It must be noted that the weight matrix and input representation of the label encoder are different with those in the text encoder.

3.3 Label Semantics Matching

3.3.1 Joint Embedding Learning

In this section, we will introduce the methods of learning a text-label joint embedding and hierarchy-aware matching relationship. For joint embedding learning, firstly, we project text semantics S_t and label semantics S_l into a common latent space as follows:

$$\Phi_t = \mathrm{FFN_t}\left(S_t\right), \quad (3)$$

$$\Phi_l = \mathrm{FFN_l}\left(S_l\right) \quad (4)$$

where $\mathrm{FFN_t}$ and $\mathrm{FFN_l}$ are independent two-layer feedforward neural networks. $\Phi_t, \Phi_l \in \mathbb{R}^{d_\varphi}$ represent text semantics and label semantics in joint embedding space, respectively. d_φ indicates the dimension of joint embedding.

In order to align the two independent semantic representations in the latent space, we employ the mean squared loss between text semantics and target labels semantics:

$$\mathcal{L}_{joint} = \sum_{p \in P(y)} \left\|\Phi_t - \Phi_l^p\right\|_2^2 \quad (5)$$

where $P(y)$ is target label sets. $\mathcal{L}_{joint}$ aims to minimize the common embedding loss between input text and target labels.

3.3.2 Hierarchy-aware Matching Learning

Based on the text-label joint embedding loss, the model only captures the correlations between text semantics and target labels semantics, while correlations among different granular labels are ignored.

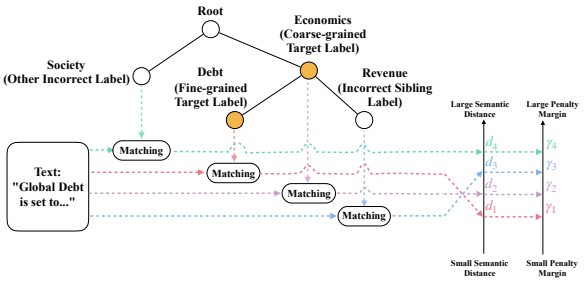

Figure 3: Illustration of hierarchy-aware margin. Target labels are colored yellow. Each colored line represent the matching operation between text and different labels. The two vertical axes for semantic matching distance and penalty margin are on the right. The semantic matching distance can be sorted by the order of d_1 (fine-grained target labels) $< d_2$ (coarse-grained target labels) $< d_3$ (incorrect sibling labels) $< d_4$ (other incorrect labels). We introduce penalty margins γ to model the relative matching relationships.

In the HTC task, it is expected that the matching relationship between text semantics and fine-grained labels should be the closest, followed by coarse-grained labels. Text semantics and incorrect labels semantics should not be related.

Insight of these, we propose a hierarchy-aware matching loss L_{match} to incorporate the correlations among text semantics and different labels semantics. L_{match} aims to penalize the small semantic distance between text semantics and incorrect labels semantics with a margin γ:

$$\mathcal{L}_{match} = \max\left(0, D\left(\Phi_t, \Phi_l^p\right) - D\left(\Phi_t, \Phi_l^n\right) + \gamma\right) \quad (6)$$

where Φ_l^p represents target labels semantics and Φ_l^n represents incorrect labels semantics. We use L2-normalized euclidean distance for metric D and γ is a margin constant for margin-based triplet loss. We take the average of all the losses between every label pairs as the margin loss.

Hierarchy-aware Margin Due to the large label sets in the HTC task, it is time-consuming to calculate every label's matching loss. Therefore, we propose hierarchy-aware sampling to alleviate the problem. Specifically, we sample all parent labels (coarse-grained labels), one sibling label, and one random incorrect label for every fine-grained label to obtain its negative label sets $n \in N(y)$. It is also unreasonable to assign the same margin for different label pairs since the label semantics similarity is quite different in a large structured label hierarchy. Our basic idea is that the semantics relationship should be closer if two labels are closer

in the hierarchical structure. Firstly, the text semantics should match fine-grained labels the most, which is exploited in joint embedding learning. Then we regard the pair with the smallest semantic distance (d_1) as a positive pair and regard other text-label matching pairs as negative pairs. As depicted in the schema figure 3, compared with the positive pair, the semantics matching distance between text and coarse-grained target labels (d_2) should be larger. The incorrect sibling labels have a certain semantic relationship with the target labels. Hence, the semantics matching distance between text and the incorrect sibling labels of fine-grained labels (d_3) should be further larger, while the semantics matching distance between text and other incorrect labels (d_4) should be the largest. We introduce hierarchy-aware penalty margins $\gamma_1, \gamma_2, \gamma_3, \gamma_4$ to model the comparable relationship. The penalty margin is smaller if we expect the semantic matching distance to be smaller. We neglect γ_1 because the matching relationships between text semantics and fine-grained labels are exploited in joint embedding learning. $\gamma_2, \gamma_3, \gamma_4$ are penalty margins compared with the matching relationships between text semantics and fine-grained labels semantics. We introduce two hyperparameters α, β to measure different matching relationships of γ:

$$\gamma_2 = \alpha\gamma; \quad \gamma_3 = \beta\gamma; \quad \gamma_4 = \gamma \tag{7}$$

where $0 < \alpha < \beta < 1$. The proposed loss captures the relative semantics similarity rankings among target labels and incorrect labels in a hierarchy-aware manner.

3.4 Classification Learning and Objective Function

We find that it is easier to overfit for classification learning if we perform classification learning in the text-label joint embedding directly. Hence, we use the text semantics representation S_t guided by joint embedding loss and matching learning loss to perform classification learning. S_t is fed into a fully connected layer to get the label probability $\hat{y}$ for prediction.

The overall objective function includes a cross-entropy category loss, joint embedding loss and hierarchy-aware matching loss:

$$\mathcal{L} = \mathcal{L}_{cls}(y, \hat{y}) + \lambda_1 \mathcal{L}_{joint} + \lambda_2 \mathcal{L}_{match} \tag{8}$$

where y and $\hat{y}$ are the ground-truth label and output probability, respectively. λ_1, λ_2 are the hyperparameters for balancing the joint embedding loss and matching learning loss. We minimize the above function by gradient descent during training.

Dataset	$\|L\|$	$Depth$	$Avg(\|L_i\|)$	$Train$	Val	$Test$
RCV1-V2	103	4	3.24	20833	2316	781265
WOS	141	2	2	30070	7518	9397
EURLEX-57K	4271	5	5	45000	6000	6000

Table 1: Statistics of three datasets for hierarchical multi-label text classification. $|L|$: Number of target classes. $Depth$: Maximum level of hierarchy. $Avg(|L_i|)$: Average Number of classes per sample. $Train/Val/Test$: Size of train/validation/test set.

4 Experiment

4.1 Experiment Setup

Datasets To evaluate the effectiveness of our model, we conduct experiments on three widely-studied datasets for hierarchical multi-label text classification. Statistics of these datasets are listed in Table 1. RCV1-V2 (Lewis et al., 2004) is a news categorization corpora, and WOS (Kowsari et al., 2017) includes abstracts of published papers from Web of Science. EURLEX57K is a large hierarchical multi-label text classification (LMTC) dataset that contains 57k English EU legislative documents, and is tagged with about 4.3k labels from the European Vocabulary (Chalkidis et al., 2019). The label sets are split into zero-shot labels, few-shot labels, and frequent labels. Few-shot labels are labels whose frequencies in the training set are less than or equal to 50. Frequent labels are labels whose frequencies in the training set are more than 50. The label setting is the same as previous work (Lu et al., 2020). In EURLEX57K, the corpora are only tagged with fine-grained labels, and the parent labels of fine-grained labels are not tagged as the target labels.

Evaluation Metric On RCV1-V2 and WOS datasets, we measure the experimental results by Micro-F1 and Macro-F1. Micro-F1 takes the overall precision and recall of all the instances into account, while Macro-F1 equals the average F1-score of labels. We report the results of two ranking metrics on large hierarchical multi-label text classification dataset EURLEX-57K, including Recall@5 and nDCG@5. The ranking metrics are preferable for EURLEX-57K since it does not introduce a significant bias towards frequent labels (Lu et al., 2020).

Implementation Details We initialize the word embeddings with 300D pre-trained GloVe vectors

(Pennington et al., 2014). Then we use a one-layer BiGRU with hidden dimension 100 and used 100 filters with kernel size [2,3,4] to setup the CNNs. The dimension of the text propagation feature and graph convolution weight matrix are both 300. The hidden size of joint embedding is 200. The matching margin γ is set to 0.2 on RCV1-V2 and WOS datasets, and set to 0.5 on EURLEX-57K dataset. We set the value of hierarchy-aware penalty hyperparameters α, β to 0.01 and 0.5, respectively. The loss balancing factor λ_1, λ_2 are set to 1. For fair comparisons with previous work (Lu et al., 2020; Chalkidis et al., 2019) on EURLEX-57K dataset, firstly, we do not use CNN layer and text feature propagation module. Secondly, to adapt to the zero-shot settings, the prediction is generated by the dot product similarity between text semantics and label semantics. Our model is optimized by Adam with a learning rate of 1e-4.

For pretrained language model BERT (Devlin et al., 2018), we use the top-level representation h_{CLS} of BERT's special CLS token to perform classification. To combine our model with BERT, we replace the text encoder of HiMatch with BERT, and the label representations are initiated by pretrained BERT embedding. The batch size is set to 16, and the learning rate is 2e-5.

Comparison Models On RCV1-V2 and WOS datasets, we compare our model with three types of strong baselines: 1) Text classification baselines: TextRCNN (Lai et al., 2015), TextRCNN with label attention (TextRCNN-LA) (Zhou et al., 2020), and SGM (Yang et al., 2018). 2) Hierarchy-aware models: HE-AGCRCNN (Peng et al., 2019), HMCN (Mao et al., 2019), Htrans (Banerjee et al., 2019), HiLAP-RL (Mao et al., 2019) which introduced reinforcement learning to simulate the assignment process, HiAGM (Zhou et al., 2020) which exploited the prior probability of label dependecies through Graph Convolution Network and TreeLSTM. 3) Pretrained language model: a more powerful pretrained language model BERT (Devlin et al., 2018) than tradition text classification models when fine-tuned on downstream tasks.

On EURLEX-57K dataset, we compare our model with strong baselines with/without zero-shot settings such as BIGRU-ATT, BIGRU-LWAN (Chalkidis et al., 2019) which introduced label-wise attention. The models starting with "ZERO" make predictions by calculating similarity scores between text and label semantics for zero-shot settings. AGRU-KAMG (Lu et al., 2020) is a state-of-the-art model which introduced various label knowledge.

4.2 Experiment Results

Models	Micro	Macro
Baselines		
TextRCNN (Zhou et al., 2020)	81.57	59.25
TextRCNN-LA (Zhou et al., 2020)	81.88	59.85
SGM (Zhou et al., 2020)	77.30	47.49
Hierarchy-Aware Models		
HE-AGCRCNN (Peng et al., 2019)	77.80	51.30
HMCN (Mao et al., 2019)	80.80	54.60
Htrans (Banerjee et al., 2019)	80.51	58.49
HiLAP-RL (Mao et al., 2019)	83.30	60.10
HiAGM (Zhou et al., 2020)	83.96	63.35
HiMatch	**84.73**	**64.11**
Pretrained Language Models		
BERT (Devlin et al., 2018)	86.26	67.35
BERT+HiMatch	**86.33**	**68.66**

Table 2: The experimental results comparing to other state-of-the-art models on RCV1-V2 dataset.

Models	Micro	Macro
Baselines		
TextRNN (Zhou et al., 2020)	77.94	69.65
TextCNN (Zhou et al., 2020)	82.00	76.18
TextRCNN (Zhou et al., 2020)	83.55	76.99
Hierarchy-Aware Models		
HiAGM (Zhou et al., 2020)	85.82	80.28
HiMatch	**86.20**	**80.53**
Pretrained Language Models		
BERT (Devlin et al., 2018)	86.26	80.58
BERT+HiMatch	**86.70**	**81.06**

Table 3: The experimental results comparing to other state-of-the-art models on Web-of-Science dataset.

Table 2, 3 and 4 report the performance of our approaches against other methods. HiAGM is an effective baseline on RCV1-V2 and WOS due to the introduction of holistic label information. However, they ignored the semantic relationship between text and labels. Our model achieves the best results by capturing the matching relationships among text and labels in a hierarchy-aware manner, which achieves stronger performances especially on Macro-F1. The improvements show that our model can make better use of structural information to help imbalanced HTC classification.

The pretrained language model BERT is an effective method when fine-tuned on downstream tasks. Compared with the results regarding HTC

	Frequent		Few		Zero		Overall	
	R@5	nDCG@5	R@5	nDCG@5	R@5	nDCG@5	R@5	nDCG@5
BIGRU-ATT (Chalkidis et al., 2019)	0.740	0.813	0.596	0.580	0.051	0.027	0.675	0.789
BIGRU-LWAN (Chalkidis et al., 2019)	0.755	0.819	0.661	0.618	0.029	0.019	0.692	0.796
ZERO-CNN-LWAN (Chalkidis et al., 2019)	0.683	0.745	0.494	0.454	0.321	0.264	0.617	0.717
ZERO-BIGRU-LWAN (Chalkidis et al., 2019)	0.716	0.780	0.560	0.510	0.438	0.345	0.648	0.752
AGRU-KAMG (Lu et al., 2020)	0.731	0.795	0.563	0.518	**0.528**	**0.414**	0.661	0.766
HiMatch	**0.769**	**0.830**	**0.697**	**0.648**	0.399	0.372	**0.705**	**0.807**

Table 4: The experimental results comparing to other state-of-the-art models on EURLEX-57K dataset.

as multiple binary classifiers, our results show that the full use of structured label hierarchy can bring great improvements to BERT model on RCV1-V2 and WOS datasets.

On EURLEX57K dataset, our model achieves the best results on different matrics except for zero-shot labels. The largest improvements come from few-shot labels. AGRU-KAMG achieves the best results on zero-shot labels by fusing various knowledge such as label semantics similarities and label co-occurrence. However, our model performs semantics matching among seen labels based on training corpora, which is not designed for a specific zero-shot learning task.

4.3 Analysis

4.3.1 Ablation Study

In this section, we investigate to study the independent effect of each component in our proposed model. Firstly, we validate the influence of two proposed losses, and the hierarchy-aware sampling. The results are reported in Table 5. The results show that F1 will decrease with removing joint embedding loss or matching learning loss. Joint embedding loss has a great influence since label semantics matching relies on the joint embedding. Besides, in the hierarchy-aware margin subsection, we perform hierarchy-aware sampling by sampling coarse-grained labels, incorrect sibling labels, and other incorrect labels as negative label sets. When we remove hierarchy-aware sampling and replace it with random sampling, the results will decrease, which shows the effectiveness of hierarchy-aware sampling.

4.3.2 Hyperparameters Study

To study the influence of the hyperparameters γ, α, and β, we conduct seven experiments on RCV1-V2 dataset. The results are reported in Table 6. The first experiment is the best hyperparameters of our model. Then we fine-tune the matching learning margin γ in experiments two and three. We find that a proper margin $\gamma = 0.2$ is beneficial for matching learning compared with a large or small margin. Furthermore, we validate the effectiveness of the hierarchy-aware margin. In experiment four, the performance will decrease if we violate the hierarchical structure by setting a large penalty margin for coarse-grained labels, and setting a small penalty margin for incorrect sibling labels. In experiment five, the performance has a relatively larger decrease if we set $\alpha = 1$ and $\beta = 1$, which ignores hierarchical structure completely. We speculate that the penalty margin that violates the hierarchical structure will affect the results, since the semantics relationship should be closer if the labels are closer in the hierarchical structure. Moreover, we validate the effectiveness of different penalty margins among different granular labels. In experiments six and seven, the results will degrade if we ignore the relationships between coarse-grained target labels and incorrect sibling labels, by setting the same margin for α and

Ablation Models	Micro	Macro
TextRCNN	81.57	59.25
HiMatch	**84.73**	**64.11**
- w/o Joint Embedding Loss	84.49	62.57
- w/o Matching Learning Loss	84.46	63.58
- w/o Hierarchy-aware Sampling	84.67	63.45

Table 5: Ablation study on RCV1-V2 dataset.

No.	γ	α	β	Micro	Macro
HiMatch					
①	0.2	0.01	0.5	**84.73**	**64.11**
Fine-tuning γ					
②	0.02	0.01	0.5	84.51	63.26
③	2	0.01	0.5	84.69	63.55
Fine-tuning α, β					
④	0.2	0.5	0.01	84.52	63.35
⑤	0.2	1	1	84.37	63.45
⑥	0.2	0.01	0.01	84.49	63.20
⑦	0.2	0.5	0.5	84.47	64.02

Table 6: Hyperparameter study on RCV1-V2 dataset.

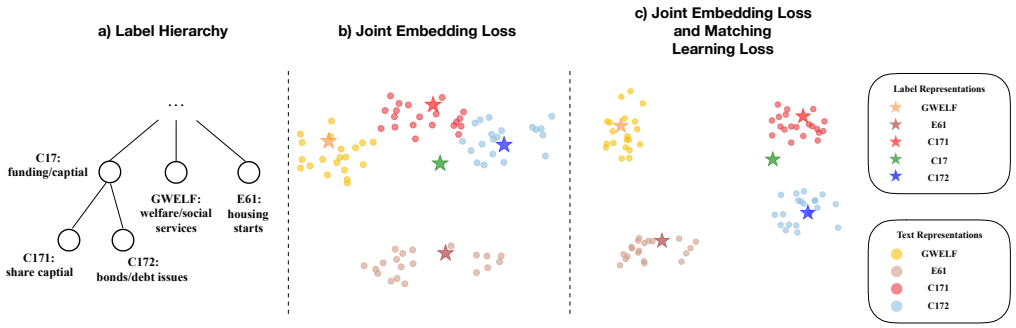

Figure 4: Figure a) is a part of the hierarchical label structure. Figure b) is the T-SNE visualization of text representations and label representations of the labels in Figure a) by introducing joint embedding loss. Figure c) is the T-SNE visualization with both joint embedding loss and matching learning loss.

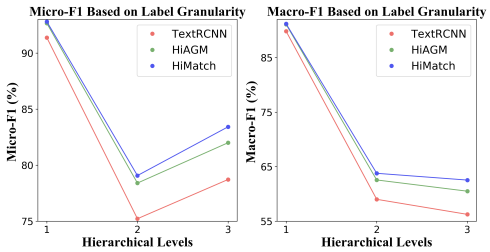

Figure 5: Performance study on label granularity based on hierarchical levels.

β. Therefore, it is necessary to set a small penalty margin for coarse-grained target labels, and a larger penalty margin for incorrect sibling labels.

4.3.3 T-SNE Visualization of Joint Embedding

We plot the T-SNE projection of the text representations and label representations in the joint embedding in Figure 4. Figure a) is a part of the hierarchical label structure in RCV1-V2. Label C171 and C172 are fine-grained labels, and label C17 is coarse-grained label of C171 and C172. GWELF and E61 are other labels with different semantics with C17, C171 and C172. In Figure b), by introducing joint embedding loss, we can see that the text representations are close to their corresponding label representations. Furthermore, the text representations of labels C171 and C172 are close to the label representation of their coarse-grained label C17. However, the text representations of different labels may overlap, since the matching relationships among different labels are ignored. In Figure c), by introducing both joint embedding loss and matching learning loss, the text representations of different labels are more separable. Other unrelated text representations and label representations such as labels GWELF, E61 are far away from C17, C171, C172. Besides, the text representations of semantically similar labels (C171 and C172) are far away relatively compared with Figure b). The T-SNE visualization shows that our model can capture the semantics relationship among texts, coarse-grained labels, fine-grained labels and unrelated labels.

4.3.4 Performance Study on Label Granularity

We analyze the performance with different label granularity based on their hierarchical levels. We compute level-based Micro-F1 and Macro-F1 scores of the RCV1-V2 dataset on TextRCNN, HiAGM, and our model in Figure 5. On RCV1-V2 dataset, both the second and third hierarchical levels contain fine-grained labels (leaf nodes). The second level has the largest number of labels and contains confusing labels with similar concepts, so its Micro-F1 is relatively low. Both the second and third levels contain some long-tailed labels, so their Macro-F1 are relatively low. Figure 5 shows that our model achieves a better performance than other models on all levels, especially among deep levels. The results demonstrate that our model has a better ability to capture the hierarchical label semantic, especially on fine-grained labels with a complex hierarchical structure.

4.3.5 Computational Complexity

In this part, we compare the computational complexity between HiAGM and our model. For time complexity, the training time of HiMatch is 1.11 times that of HiAGM with batch size 64. For space complexity during training, HiMatch has 37.4M parameters, while HiAGM has 27.8M. The increase mainly comes from the label encoder with large

label sets. However, during testing, the time and space complexity of HiMatch is the same as HiAGM. The reason is that only the classification results are needed, and we can remove the joint embedding. HiMatch achieves new state-of-the-art results, and we believe that the increase of computational complexity is acceptable.

5 Conclusion

Here we present a novel hierarchical text classification model called HiMatch that can capture semantic relationships among texts and labels at different abstraction levels. Instead of treating HTC as multiple binary classification tasks, we consider the text-label semantics matching relationship and formulate it as a semantic matching problem. We learn a joint semantic embedding between text and labels. Finally, we propose a hierarchy-aware matching strategy to model different matching relationships among coarse-grained labels, fine-grained labels and incorrect labels. In future work, we plan to extend our model to the zero-shot learning scenario.

Acknowledgments

We thank the anonymous reviewers for their helpful feedbacks. The work described in this paper was partially funded by the National Natural Science Foundation of China (Grant No. 61502174, and 61872148), the Natural Science Foundation of Guangdong Province (Grant No. 2017A030313355, 2019A1515010768 and 2021A1515011496), the Guangzhou Science and Technology Planning Project (Grant No. 201704030051, and 201902010020), the Key R&D Program of Guangdong Province (No. 2018B010107002) and the Fundamental Research Funds for the Central Universities.

References

Aixin Sun and Ee-Peng Lim. 2001. Hierarchical text classification and evaluation. In *Proceedings 2001 IEEE International Conference on Data Mining*, pages 521–528.

Siddhartha Banerjee, Cem Akkaya, Francisco Perez-Sorrosal, and Kostas Tsioutsiouliklis. 2019. Hierarchical transfer learning for multi-label text classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6295–6300, Florence, Italy. Association for Computational Linguistics.

Ilias Chalkidis, Emmanouil Fergadiotis, Prodromos Malakasiotis, and Ion Androutsopoulos. 2019. Large-scale multi-label text classification on EU legislation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6314–6322, Florence, Italy. Association for Computational Linguistics.

Boli Chen, Xin Huang, Lin Xiao, Zixin Cai, and Liping Jing. 2020. Hyperbolic interaction model for hierarchical multi-label classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7496–7503.

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*.

Johannes Fürnkranz, Eyke Hüllermeier, Eneldo Loza Mencía, and Klaus Brinker. 2008. Multilabel classification via calibrated label ranking. *Mach. Learn.*, 73(2):133–153.

K. Kowsari, D. E. Brown, M. Heidarysafa, K. Jafari Meimandi, M. S. Gerber, and L. E. Barnes. 2017. Hdltex: Hierarchical deep learning for text classification. In *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 364–371.

Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Recurrent convolutional neural networks for text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29.

David D. Lewis, Yiming Yang, Tony G. Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research. 5:361–397.

Jueqing Lu, Lan Du, Ming Liu, and Joanna Dipnall. 2020. Multi-label few/zero-shot learning with knowledge aggregated from multiple label graphs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2935–2943, Online. 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)*, pages 445–455, Hong Kong, China. Association for Computational Linguistics.

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.

Jeffrey Pennington, Richard Socher, and Christopher 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.

Bingyu Wang, Li Chen, Wei Sun, Kechen Qin, Kefeng Li, and Hui Zhou. 2019. Ranking-based autoencoder for extreme multi-label classification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2820–2830, Minneapolis, Minnesota. Association for Computational Linguistics.

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, Hong Kong, China. 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*, pages 3915–3926, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Chih-Kuan Yeh, Wei-Chieh Wu, Wei-Jen Ko, and Yu-Chiang Frank Wang. 2017. Learning deep latent space for multi-label classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Jie Zhou, Chunping Ma, Dingkun Long, Guangwei Xu, Ning Ding, Haoyu Zhang, Pengjun Xie, and Gongshen Liu. 2020. Hierarchy-aware global model for hierarchical text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1106–1117, Online. Association for Computational Linguistics.