# A Novel Negative Sample Generation Method for Contrastive Learning in Hierarchical Text Classification

**Juncheng Zhou[1], Lijuan Zhang[1,*], Yachen He[2], Rongli Fan[1], Lei Zhang[1], Jian Wan[1,*],**

[1]Zhejiang University of Science and Technology, [2]ByteDance Ltd.,
[1]{222308855017, zhanglijuan, fanrongli, leizhang, wanjian}@zust.edu.cn
[2]heyachen@bytedance.com

## Abstract

Hierarchical text classification (HTC) is an important task in natural language processing (NLP). Existing methods typically utilize both text features and the hierarchical structure of labels to categorize text effectively. However, these approaches often struggle with fine-grained labels, which are closely similar, leading to difficulties in accurate classification. At the same time, contrastive learning has significant advantages in strengthening fine-grained label features and discrimination. However, the performance of contrastive learning strongly depends on the construction of negative samples. In this paper, we design a hierarchical sequence ranking (HiSR) method for generating diverse negative samples. These samples maximize the effectiveness of contrastive learning to enhance the ability of the model to distinguish between fine-grained labels and improve the performance of the model in HTC. Specifically, we transform the entire label set into linear sequences based on the hierarchical structure and rank these sequences according to their quality. During model training, the most suitable negative samples are dynamically selected from the ranked sequences. Then contrastive learning amplifies the differences between similar fine-grained labels by emphasizing the distinction between the ground truth and the generated negative samples, thereby enhancing the discriminative ability of the model. Our method has been tested on three public datasets and achieves state-of-art (SOTA) on two of them, demonstrating its effectiveness.

## 1 Introduction

Hierarchical text classification (HTC) involves classifying text into a structured set of categories arranged from the most general to the most specific, capturing the complexity and multidimensionality of language (Vens et al., 2008). The necessity for HTC stems from the inherent complexity and vast amount of textual data in various fields, including but not limited to academic literature (Kowsari et al., 2017b), legal documents, and online content (Lewis et al., 2004; Sandhaus, 2008). In HTC, the hierarchical structures are typically modeled as trees or directed acyclic graphs (DAGs) (Silla and Freitas, 2011), as shown in Figure 1.

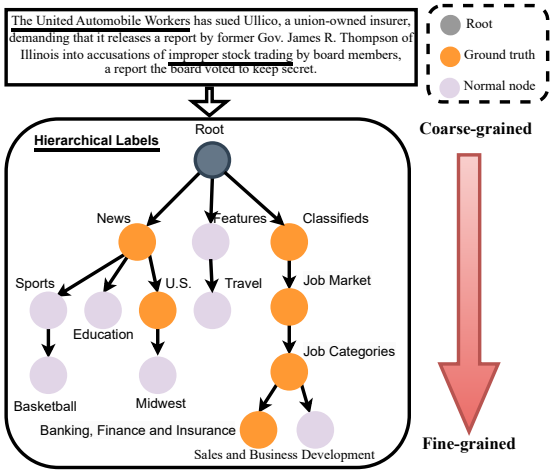


Figure 1: An example of HTC on NYT. The hierarchical labels are arranged from top to bottom, with granularity ranging from coarse to fine.

Existing methods for HTC can be broadly categorized into local and global approaches. Local methods (Wehrmann et al., 2018b; Shimura et al., 2018a; Banerjee et al., 2019a) assign a separate classifier to each node, resulting in an architecture with a large number of parameters, which can easily lead to exposure bias. On the other hand, the global approaches (Zhou et al., 2020; Yu et al., 2022; Chen et al., 2021) use a single classifier for the entire hierarchy, resulting in fewer parameters and higher efficiency. However, these existing methods struggle to address the challenge posed by HTC, where the imbalanced data distribution and complex dependencies across multiple levels in the hierarchy lead to reduced discrimination among fine-grained

*Corresponding authors.



*Proceedings of the 31st International Conference on Computational Linguistics*, pages 5645–5655
January 19–24, 2025. 

labels. Figure 1 provides a clear illustration of the aforementioned challenge. For example, the distinction between coarse-grained labels such as "News", "Features", and "Classifieds" is greater than that between fine-grained labels like "Banking, Finance and Insurance" and "Sales and Business Development". Moreover, distinguishing between deep, fine-grained labels is crucial for the performance of HTC. Fortunately, contrastive learning is highly effective in enhancing label features and label distinguishability (Wang et al., 2022a; Zhang et al., 2024). However, researches combining HTC with contrastive learning are still at an early stage. Existing studies (Wang et al., 2022a; Zhang et al., 2024) have mainly focused on constructing positive samples through data augmentation from a textual perspective, without addressing the challenge of distinguishing similar fine-grained labels. The core of tackling the challenge is to combine complex hierarchical structures to construct appropriate positive and negative samples.

In order to effectively solve the above problems, we design and propose a hierarchical sequence ranking (HiSR) method to generate more effective negative samples from the label perspective, which aims to assist contrastive learning to enhance the ability of the model to distinguish fine-grained labels and improve the performance of the model to classify. The core idea of this strategy is to make full use of the overall hierarchical structure of the label system and the depth-first search algorithm (DFS) (Tarjan, 1972) to construct multiple sequence negative samples, and then dynamically select the most challenging negative sample combinations during model training, ensuring the high quality and diversity of negative samples. Not only that, this targeted dynamic selection process is crucial for amplifying the differences between fine-grained labels.

The major contributions of this work are as follows :

- We innovatively designed a negative sample generation strategy called HiSR, which provides high-quality negative samples for contrastive learning to solve the problem that fine-grained labels are difficult to distinguish.
- HiSR models labels into sequences and sorts them according to their quality, which provides a basis for the introduction of negative-negative sample comparison. In addition, a dynamic selection mechanism is used during training to provide the model with the most appropriate negative samples.
- Experiments demonstrate that our proposed method outperforms previous studies on three widely used public datasets. Our code is available at https://github.com/zjcjason/HiSR.

## 2 Related Work

### 2.1 Hierarchical Text Classification

The main challenges of HTC include making full use of the hierarchical information between categories, distinguishing similar fine-grained labels, and dealing with data imbalance. In response to these challenges, extant researches have proposed various methodologies, which can be broadly categorized into two main categories: local approaches and global approaches.

Local approaches employ specialized classifiers at each node or layer within the hierarchy. (Banerjee et al., 2019b) introduced a parameter passing strategy from parent to child classifiers to improve node efficiency. Wehrmann et al. (2018a) proposed a hybrid method combining local and global strategies to reduce exposure bias. Shimura et al. (2018b) tackled class distribution skewness with a parameter-sharing mechanism for more balanced learning. Peng et al. (2018) used N-gram tokens and GCN with recursive regularization to capture hierarchical representations. These methods reflect the dynamic evolution of local approaches in HTC.

Global approaches utilize a unified model to simultaneously assign multiple hierarchical labels, integrating the entire hierarchy within a single predictive framework. Researchers (Gopal and Yang, 2013; Wu et al., 2019; Mao et al., 2019) have delved into the intricate relationships between successive hierarchies, particularly parent-child dynamics. Following investigations have redirected their focus towards a variety of aspects, including the probability associated with prior hierarchies (Zhou et al., 2020), the overall structure of labels (Wang et al., 2021a), data imbalance (Deng et al., 2021), the alignment between labels and semantics (Chen et al., 2021), the application of contrastive learning in token representation (Wang et al., 2022b). Furthermore, HJCL (Yu et al., 2023) built hierarchical-aware joint supervised contrastive learning based on HGCLR, and HALB (Zhang et al., 2024) added multi-label negative supervision and asymmetric loss function. Moreover,

advancements have been made in prompt-tuning and multi-label masking language models aimed at forming a coherent understanding of hierarchical semantics by Wang et al. (2022c) and in developing common representations that incorporate both local and global hierarchical dimensions as proposed by Jiang et al. (2022). Recently, NERHTC (Cai et al., 2024) treated HTC as a named entity recognition (NER) task.

Consequently, the challenge for HTC is to design an algorithm that can exploit the hierarchical relationships among labels to improve the accuracy and efficiency of classification.

### 2.2 Contrastive Learning

Contrastive learning (He et al., 2020) is a widely used technique in machine learning and deep learning, designed to learn generalizable representations beneficial for downstream tasks by comparing different examples. A critical aspect of this approach is the construction of positive and negative samples. In natural language processing (NLP), standard methods for constructing these samples include back-translation (Wang et al., 2021b), deletion, reordering and replacement of words and spans (Wu et al., 2020), random corruption of original tokens (Yu et al., 2021), and so on. These approaches exemplify the generation of positive samples, while the inclusion of negative samples can further enhance the effectiveness of contrastive learning. For instance, Wang et al. (2021b) generated negative samples by introducing perturbations through semantic-altering word modifications, while Pan et al. (2022) adjusted the embedding layer of the model to generate adversarial samples.

These methods are customized to generate positive and negative samples according to specific downstream tasks. However, due to the complex hierarchical structure of HTC, it is extremely difficult to generate positive and negative samples that conform to both text semantics and hierarchical structure.

## 3 Methodology

In this section, we focus on negative sample generation and contrastive learning module, which play a crucial role in improving the ability of the model to distinguish fine-grained hierarchical labels. Moreover, we describe the text and structure encoders. Finally, we provide an overview of the objective function that integrates all these components to optimize the overall performance of the model. Figure 2 illustrates the overall architecture of the proposed model.

### 3.1 Problem Definition

In the domain of HTC, the objective is to map a given collection of texts $X = \{x_1, x_2, \ldots, x_n\}$, where each $x_i$ represents an independent text instance and $n$ is the number of the collection, to a subset of labels $Y_i \subseteq Y$, with $Y = \{y_1, y_2, \ldots, y_n\}$ constituting the complete set of potential labels. Distinct from traditional text classification, the HTC task is characterized by a predefined hierarchical relationship $H$ among the labels, which may manifest as a tree structure, a directed acyclic graph (DAG), or other intricate hierarchical configurations. This hierarchical organization not only reflects the semantic relationships among labels but also plays a pivotal role in the comprehension of complex informational content.

### 3.2 Text Encoder

Given the challenges of HTC, BERT is chosen as the text encoder for its ability to capture bidirectional context, allowing a comprehensive understanding of language. BERT is pre-trained on large-scale data, and then fine-tuned for specific tasks, making it highly effective in various NLP applications.

Assuming the input text as $\mathbf{x}$:

$$\mathbf{x} = \{[CLS], t_1, t_2, \ldots, t_{n-1}, [SEP]\} \quad (1)$$

In this context, $[CLS]$ and $[SEP]$ serve as specific placeholder tokens representing the beginning and end of the input text, respectively. After the text $\mathbf{x}$ is fed into BERT, we can obtain the hidden states for each token:

$$H_{status} = BERT(\mathbf{x}) \quad (2)$$

where $H_{status} \in \mathbb{R}^{n \times d_h}$ and $d_h$ is the hidden size. The $[CLS]$ token, due to its positioning and the function of the self-attention mechanism, can gather comprehensive information from the entire text sequence. Consequently, we utilize the hidden state of the $[CLS]$ token to represent the entire text in the subsequent process.

### 3.3 Structure Encoder

Graph Convolutional Networks (GCNs) are widely utilized as structural encoders for aggregating node

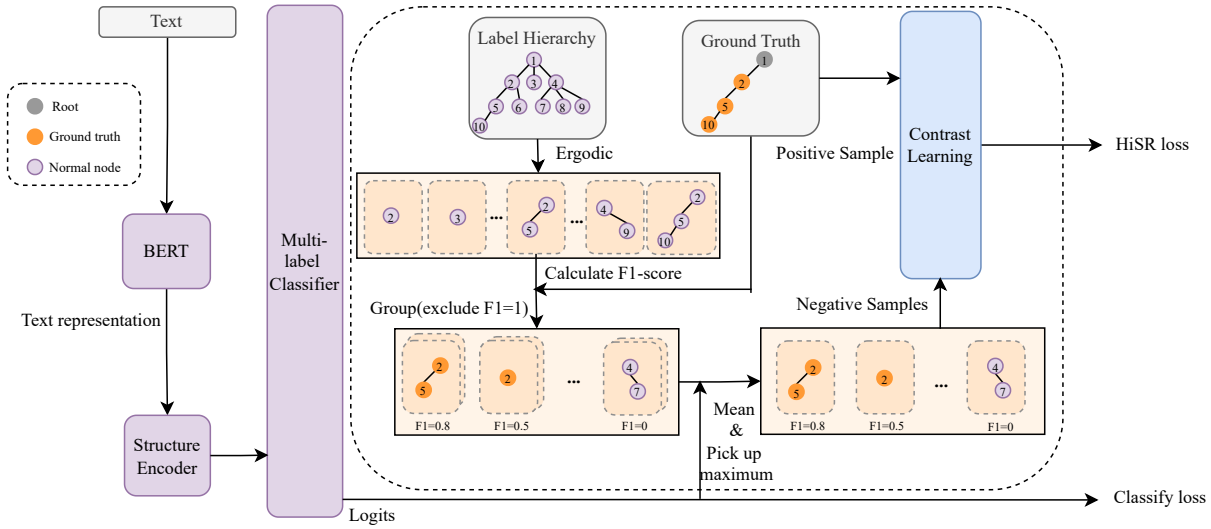


Figure 2: The overall architecture of the proposed model. The dashed box illustrates the main innovations, including the overall process of HiSR and the contrastive learning module.

information in NLP, as noted by (Rios and Kavuluru, 2018). GCNs are a specific type of neural network that has been developed for the processing of graph-structured data. They are designed to effectively capture the complex relationships between nodes and the overall structure of graphs by directly manipulating the nodes and edges that comprise them. The core advantage of GCNs lies in their ability to utilize the local connectivity patterns of nodes, integrating these local insights to form a global understanding of the entire graph.

A Hierarchical-GCN is employed to aggregate global fine-grained hierarchical label structure relationship information. The Hierarchical-GCN aggregates data flows within and across layers through top-down, bottom-up and self-loop edges. In the hierarchical graph, each node represents a corresponding label and each directed edge represents the related features of relationships between labels. The new feature representation of each node is computed by combining its own features with those of adjacent nodes. This process typically involves weighting the features of neighboring nodes, followed by a linear transformation (e.g., multiplying by a weight matrix) and a nonlinear activation function to update the node's feature representation. However, the aforementioned transformations utilize independent weight matrices to encode information for each directed edge. This encoding significantly increases the number of parameters and complexity of the model, which may lead to over-parameterization. Following the approach of (Zhou et al., 2020), we simplify this transformation by using hierarchical prior probabilities as the weighted adjacency matrix.

The hidden state of node $i$ is encoded in the hierarchical GCN through its associated neighbors $M(i) = \{m_i, \text{child}(i), \text{parent}(i)\}$, specifically:

Firstly, defining the union vector $w_{i,t}$ of node $i$ and its neighboring node $t$ as:

$$w_{i,t} = c_{i,t} x_t + d_i^m \tag{3}$$

where $c_{i,t}$ is the hierarchical probability coefficient, with the self-loop edge $c_{i,i} = 1$ , top-down edge $h_c(q_{t,i}) = \frac{M_i}{M_t}$ and bottom-up edge $h_p(q_{t,i}) = 1$, $x_t$ represents the feature vector of neighboring node $t$ and $d_i \in \mathbb{R}^{N \times \text{dim}}$ is the bias term.

Then, the eigenvectors of node $i$ are calculated using weight matrix transformation and activation function $sigmod\ \sigma$ to obtain edge features $f_{i,t}$ :

$$f_{i,t} = \sigma(U_f^{e(t,i)} x_i + d_f^i) \tag{4}$$

where $U_f^{e(t,i)} \in \mathbb{R}^{\text{dim}}$ is the hierarchical direction weight matrix from node $t$ to node $i$ and $d_f^i \in \mathbb{R}^N$ is the bias term.

Finally, the output hidden state $p_i$ of node $i$ is calculated by the weighted sum of all neighboring node features and is activated by the ReLU activation function:

$$p_i = \text{ReLU}(\sum_{t \in M(i)} f_{i,t} \odot w_{i,t}) \tag{5}$$

The output hidden state $p_i$ of node $i$ represents its label representation, capturing the hierarchical structural information.

### 3.4 Negative Sample Generation and Contrastive learning Module

Previous researches on HTC have achieved substantial results. However, at the deeper level, the existing models still face a challenge in accurately classifying some complex, fine-grained labels. To navigate this challenge, we employ contrastive learning to enhance the differentiation between these labels (Wang et al., 2022b). The quality of negative samples is a crucial factor for contrastive learning to be effective (Yu et al., 2023; Zhang et al., 2024). To address this, we propose a novel negative sample generation method—HiSR, to provide a comprehensive and diverse set of negative samples for contrastive learning.

Specifically, HiSR first uses the DFS algorithm to traverse the entire label tree and generate a sequence for each label from top to bottom. The sequence corresponding to each label starts with the label of the first layer and ends with itself. All sequences constitute the negative sample set $S$. The predicted probability $P$ of each sequence is calculated as the average of probabilities of all labels in the sequence, and the formula is as follows:

$$P = \frac{\sum_{i=1}^{L} p_i}{L} \tag{6}$$

where $L$ represents the sequence length and $p_i$ represents the probability of the $i$-th label in the sequence. In conclusion, the contrastive loss function of positive-negative samples $\mathcal{L}_{pn}$ can be expressed as follows:

$$\mathcal{L}_{pn} = \sum_{j \in S} \max(0, P_p - P_j + \lambda) \tag{7}$$

where $P_p$ represents the prediction probability of the ground truth, $P_j$ represents the prediction probability of the $j$-th sequence and $\lambda$ is the margin between negative samples and positive sample.

However, simply comparing positive-negative samples will cause the model to overlook the subtle differences between negative samples, so we introduce negative-negative sample comparison. The negative-negative sample comparison further refines the distinction between negative samples, enhances the sensitivity of the model to subtle feature differences in fine-grained labels, and prevents the model from simply classifying all negative samples into one category, thereby improving the ability of the model to classify fine-grained labels. Furthermore, negative-negative sample comparison can help the model better understand the complex relationships in the label hierarchy. In HTC, different negative samples may share partially overlapping label paths. By comparing these negative samples, the model can gain a deep understanding of parent-child and sibling relationships within the label hierarchy, thereby enhancing its cognitive ability to discern the intricacies of the label structure. This enhancement of cognitive ability is crucial to improving the classification performance of the model in a complex hierarchical label system.

Specifically, in the negative-negative sample comparison, it is assumed that sequences that are closer to the ground truth should be assigned a higher prediction probability. Thus, HiSR initially calculates the F1 score (Manning et al., 2008) of the ground truth and the generated sequences, which is a commonly utilized evaluation metric in HTC. Thereafter, sequences with the same F1 score are grouped together. Subsequently, these groups are sorted in descending order according to the F1 score, so that sequences that are closer to the ground truth are ranked first. Finally, in the training phase, the sequence with the highest prediction probability is selected from the sorted groups in each epoch as the negative sample. This dynamic selection strategy will provide the most challenging negative samples for contrastive learning, thereby optimally leveraging the potential of contrastive learning. The negative-negative sample contrastive loss function is shown below.

$$\mathcal{L}_{nn} = \sum_{i} \sum_{j>i} \max(0, P_i - P_j + \lambda_{ij}) \tag{8}$$

where $\lambda_{ij}$ is the margin multiplied by the distance in rank between the samples, i.e., $\lambda_{ij} = (j - i) * \lambda$.

The negative-negative sample contrastive loss function improves the capacity of the model to identify negative samples by assessing samples with different scores in the absence of standard samples, thereby showing unique advantages and broad application prospects in contrastive learning (Liu et al., 2022).

In conclusion, the total loss function of contrastive learning can be expressed as Eq. 9. The integration of positive-negative and negative-negative sample comparison strategies can comprehensively improve the performance of the model in the HTC task. This approach not only optimizes the capacity of the model to represent features and classify data accurately but also offers novel insights and methodologies for the application of contrastive

learning in intricate label structures.

$$\mathcal{L}_{HiSR} = \mathcal{L}_{pn} + \mathcal{L}_{nn} \tag{9}$$

### 3.5 Classification Loss and Objection Function

Following previous work (Zhou et al., 2020; Wang et al., 2022a), a binary cross-entropy loss function $L^C$ is employer for classification,

$$L_{ij}^C = -y_{ij}\log(p_{ij}) - (1-y_{ij})\log(1-p_{ij}) \tag{10}$$

$$L^C = \sum_{i=1}^{N}\sum_{j=1}^{k} L_{ij}^C \tag{11}$$

where $y_{ij}$ is the ground truth. $L_{ij}^C$ and $p_{ij}$ is the binary cross-entropy loss and probability of text $i$ on label $j$.

The final loss function of training is a combination of classification and HiSR:

$$L = L^C + \alpha L_{HiSR} \tag{12}$$

where $\alpha$ is a hyperparameter for balancing the HiSR loss.

## 4 Experiments

In this section, we will introduce datasets, evaluation metrics, implementation details, experimental results, and ablation studies.

### 4.1 Datasets and Evaluation Metrics

This study employs three widely recognized datasets to ensure the generalizability and reliability of the experimental results: Web of Science (WOS) (Kowsari et al., 2017a), RCV1 (Lewis et al., 2004) and New York Times (NYT) (Sandhaus, 2008). RCV1 and NYT are text classification datasets from the news domain, whereas WOS comprises abstracts of research papers collected from the Web of Science. These datasets are annotated with hierarchical labels. All data preprocessing and partitioning are derived from Zhou et al. (2020). Notably, WOS is suited for HTC with a single-path framework, while RCV1 and NYT are multi-path classification labels. Detailed statistical data about these datasets can be found in Table 1. In accordance with previous researches (Zhou et al., 2020; Wang et al., 2022c), the same evaluation metrics are used: Macro-F1 and Micro-F1.

| 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: Detailed information on three datasets. $|L|$ is the number of classes. **Depth** represents the maximum level of the hierarchy. **Avg**($|L_i|$) is the average number of classes for each sample in the dataset.

### 4.2 Baselines

To verify the performance of the proposed method, we select some representative baselines. HiAGM (Zhou et al., 2020) leverages a hierarchy-aware multi-label attention mechanism to exploit the prior probability of label dependencies for generating mixed features. HTCInfoMax (Deng et al., 2021) enhances HiAGM by implementing information maximization to model text-hierarchy interactions, optimizing text-label mutual information and regularizing label representations to align with a prior distribution. HiMatch (Chen et al., 2021) addresses the problem as a semantic matching task by aligning text and label representations within a joint embedding space, utilizing this joint representation for classification. HGCLR (Wang et al., 2022b) enhances the representation of the encoder through contrastive learning and introduces a new graph encoder to extract hierarchical label information. HPT (Wang et al., 2022c), a Hierarchy-aware Prompt Tuning method, which constructs dynamic virtual templates and label words and introduces a zero-bounded multi-label cross-entropy loss. HJCL (Yu et al., 2023) combines instance-level and label-level contrastive learning techniques. HALB (Zhang et al., 2024) uses a multi-label negative supervision method to enhance the perception of text representation on the label hierarchy and introduces an asymmetric loss function to solve the label imbalance problem. NERHTC (Cai et al., 2024) transforms HTC into a named entity recognition (NER) task and combines conditional random fields (CRF) and Global Pointer to establish hierarchical dependencies.

### 4.3 Implementation Details

In our study, we implement the model end-to-end using the PyTorch deep learning framework. We select the HiAGM-TP variant from Zhou et al. (2020), substituting the text encoder with $bert-base-uncased$ from the Hugging Face Transformer (Wolf et al., 2020) library while retaining all default parameters. Additionally, we adhere to

| Model | WOS | | NYT | | RCV1-V2 | |
|---|---|---|---|---|---|---|
| | Micro-F1 | Macro-F1 | Micro-F1 | Macro-F1 | Micro-F1 | Macro-F1 |
| **Hierarchy-Aware Models** | | | | | | |
| HiAGM(Zhou et al., 2020) | 85.82 | 80.28 | 74.97 | 60.83 | 83.96 | 63.35 |
| HTCInfoMax(Deng et al., 2021) | 85.58 | 80.05 | - | - | 83.51 | 62.71 |
| HiMatch(Chen et al., 2021) | 86.20 | 80.53 | - | - | 84.73 | 64.11 |
| **Pretrained Language Models** | | | | | | |
| BERT (Our implement) | 85.72 | 79.34 | 77.64 | 65.91 | 85.77 | 67.04 |
| BERT+HiAGM (Our implement) | 86.32 | 80.67 | 78.27 | 66.04 | 86.37 | 67.14 |
| BERT+HTCInfoMax(Wang et al., 2022b) | 86.30 | 79.97 | 78.75 | 67.31 | 85.53 | 67.09 |
| BERT+HiMatch(Chen et al., 2021) | 86.70 | 81.06 | - | - | 86.33 | 68.66 |
| HGCLR(Wang et al., 2022b) | 87.11 | 81.20 | 78.86 | 67.96 | 86.49 | 68.31 |
| HPT(Wang et al., 2022c) | 87.16 | 81.93 | 80.42 | 70.42 | 87.26 | 69.53 |
| HJCL(Yu et al., 2023) | - | - | 80.52 | 70.02 | 87.04 | 70.49 |
| HALB(Zhang et al., 2024) | 87.45 | **82.04** | 79.56 | 69.28 | 86.94 | 69.32 |
| NERHTC(Cai et al., 2024) | 87.42 | 81.93 | **80.97** | **70.99** | 87.5 | 69.76 |
| HiSR(Ours) | **87.52** | **82.04** | 80.32 | 70.11 | **87.59** | **70.72** |

Table 2: Our proposed model is evaluated on three datasets and compared with previous research findings. The reported results represent the mean values obtained from three independent trials. The best results are in bold.

the original settings for HiAGM-TP as specified in the corresponding paper.

For our experiments, the batch size is configured to 8 for WOS and RCV1, but for NYT, it is set to 4. We employ the Adam optimizer with the learning rate set to 1e-5. During the training phase, we apply an early stopping strategy based on the performance of the model on the development set after each epoch. Specifically, if neither of the two F1 scores improves within six consecutive epochs, training will be terminated immediately.

In the context of contrastive learning, the weight is denoted as $\alpha$ and set to 0.01. The margin $\lambda$ for negative-negative samples is configured to 0.15 and for positive-negative samples is set to 0.5.

### 4.4 Results

The experimental results of our method on three datasets and the comparison with previous researches are shown in Table 2. To ensure an equitable comparison, we conducted essential experiments using our own device. Except for NYT, our method achieves SOTA performance.

HiSR demonstrates significant improvements over BERT across all datasets. On WOS, we observe performance gains of 1.8% and 2.7% in Micro-F1 and Macro-F1 scores compared with BERT. Compared to NERHTC, HiSR achieves marginal yet consistent improvements of 0.1% and 0.11% in Micro-F1 and Macro-F1. The hierarchical two-layer structure of WOS aligns well with the strengths of HiSR, facilitating efficient construction of appropriate sample sets and yielding robust performance. For RCV1, characterized by its four-layer structure and largest test set compared to other datasets, HiSR outperforms BERT by 1.82% and 3.68% in Micro-F1 and Macro-F1. When compared to NERHTC, HiSR shows improvements of 0.09% and 0.96% in Micro-F1 and Macro-F1. The NYT, despite its complexity due to multi-path and multi-label characteristics, which pose challenges for HiSR, still sees substantial improvements. HiSR surpasses BERT by 2.68% and 4.2% in Micro-F1 and Macro-F1 scores. These results consistently demonstrate the efficacy of our proposed model across datasets with varying structures and complexities.

## 5 Analysis

### 5.1 Ablation Study

To demonstrate the effectiveness of our proposed method, we conduct an ablation study on RCV1 and the results are shown in Table 3. The core component of the model is HiSR. After removing it, Macro-F1 drops drastically by 3.58% and Micro-F1 by 1.22%. Such a large drop shows the effectiveness of HiSR. After that, GCN is a graph encoder that can encode hierarchical structures. After removing it, Micro-F1 and Macro-F1 drop by 1.06% and 2.27% respectively. This shows that hierarchical encoding is indispensable in HTC.

### 5.2 Effect of Local Structure

Our approach aims to address the challenge of fine-grained labels that are difficult to distinguish, which is common in HTC research. To demonstrate

| Ablation Models | Micro-F1 | Macro-F1 |
|---|---|---|
| BERT | 85.77 | 67.04 |
| HiSR(Ours) | **87.59** | **70.72** |
| $-r.m.$ HiSR | 86.37 | 67.14 |
| $-r.m.$ GCN | 86.53 | 68.45 |

Table 3: Ablation experiments on RCV1. $-r.m.$ stands for remove, $-r.p.$ stands for replaced with.

| Model | WOS | | WOS | |
|---|---|---|---|---|
| | PMicro-F1 | PMacro-F1 | CMicro-F1 | CMacro-F1 |
| **BERT** | 79.96 | 78.40 | 85.43 | 79.37 |
| **HPT** | 80.69 | 79.03 | 86.57 | 80.85 |
| **NERHTC** | 81.41 | 79.52 | 87.14 | 81.36 |
| **HiSR(Ours)** | **81.66** | **80** | **87.29** | **81.56** |

Table 4: Further experiments of path-based and path-constrained metrics on WOS.

| Method | RCV1-V2 | | NYT | |
|---|---|---|---|---|
| | CMicro-F1 | CMacro-F1 | CMicro-F1 | CMacro-F1 |
| **BERT** | 85.68 | 66.96 | 78.05 | 64.62 |
| **HPT** | 86.95 | 68.15 | 79.51 | 68.38 |
| **NERHTC** | 86.99 | 68.46 | **80.11** | **69.42** |
| **HiSR(Ours)** | **87.12** | **69.81** | 79.14 | 68.16 |

Table 5: Further experiments of path-constrained metrics on RCV1 and NYT.

the effectiveness of our model, we conduct further experiments from two aspects: path consistency and label granularity.

### 5.2.1 Path Consistency

Failure to accurately classify fine-grained labels can lead to path inconsistency. Path inconsistency refers to the situation where the parent node is correctly predicted and the child node is incorrectly predicted, or the child node is correctly predicted and the parent node is incorrectly predicted on the same path. Following Chen et al. (2021) and Ji et al. (2023), we use path-constrained metrics and path-based metrics to evaluate the effectiveness of our method in solving the path inconsistency problem, further proving that our method can address the challenge of fine-grained label misclassification.

In the path-constrained metrics (CMicro-F1 and CMacroF1), a node is considered to be correctly predicted only if all of its ancestor nodes are correctly predicted. The path-based metrics (PMicro-F1 and PMacro-F1) are used to evaluate the correctness of all labels on the entire path, but they can only be used on the mandatory-leaf (Bi and Kwok, 2012) dataset, such as WOS.

We conducted further experiments on the three datasets, and the results are shown in the table 4 and 5. Both path-constraints and path-based metrics are applicable to WOS, and our model also achieves SOTA on WOS. This shows that our method performs excellently on a two-layer structure dataset like WOS. In further experiments, only the path-constraint metrics can be used for RCV1 and NYT. In RCV1, HiSR surpasses all existing models as expected. However, the eight-layer complex structure and the large number of labels of NYT pose some challenges to HiSR.

### 5.2.2 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 NYT on BERT, HPT, and our model. The results are shown in Figure 3. Since NERHTC does not open source their code and report specific results on label granularity in their paper, we can not include NERHTC in the comparison. NYT has eight layers, and the number of labels in each layer is 4, 27, 51, 47, 17, 12, 6, and 2. We can see that the number of labels from the second to fourth layers is larger, and they contain confusing labels with similar concepts, so the two metrics drop quickly in these layers. At the same time, as the number of layers increases, the label granularity becomes finer, and the performance of the model further decreases despite the small number of labels. According to Figure 3, our model is slightly inferior to HPT in the third layer, but the rest of the layers are better than other models. The gap with other models is even bigger in deeper layers, which fully proves the effectiveness of our method in improving the model's ability to distinguish fine-grained labels.

## 6 Conclusion

This paper proposes a negative sample generation method called hierarchical sequence ranking (HiSR) to provide high-quality negative samples for contrastive learning, thereby solving the challenge of similar fine-grained labels being difficult to distinguish, which is common in HTC tasks. The negative samples generated by HiSR can not only transfer the subtle differences between fine-grained labels to the model, but also the negative-negative sample comparison introduced according to its ordered characteristics can help the model better understand the complex relationship between label levels. HiSR provides a new idea for constructing negative samples for hierarchically structured data. Experiments show that HiSR achieves consis-

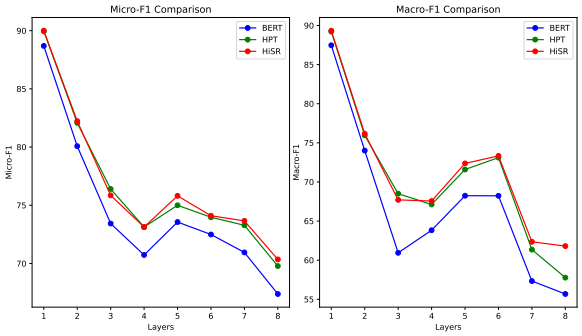


Figure 3: Results of the performance of hierarchical label granularity on NYT.

tent improvements over the selected baselines and reaches SOTA performance on some datasets.

## Limitations

Experimental results show that HiSR can improve the discrimination of fine-grained labels. However, it does not achieve SOTA on NYT. We speculate that it is difficult for HiSR to construct perfect negative samples due to the excessive number of layers and the existence of multiple paths of different lengths. In addition, the ranking metric may not be optimal. We will investigate the above issues further in future work.

## Ethics Statement

All data utilized in this research were obtained from publicly available datasets that have been previously released for academic and research purposes.

## Acknowledgments

The research was partially funded by National Natural Science Youth Science Foundation Project (Grant No. 62201508), Zhejiang Provincial Natural Science Foundation Youth Fund Project (Grant No. LQ23F010004), Scientific Research Project funded by Zhejiang Provincial Department of Education (Grant No. Y202455182), the "Pioneer" and "Leading Goose" R&D Program of Zhejiang Province (Grant No. 2024C01166) and Local Science and Technology Development Fund Projects Guided by the Central Government(Grant No. 2023ZY1068).

## References

Siddhartha Banerjee, Cem Akkaya, Francisco Perez-Sorrosal, and Kostas Tsioutsiouliklis. 2019a. 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.

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

Wei Bi and James Kwok. 2012. Mandatory leaf node prediction in hierarchical multilabel classification. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc.

Fuhan Cai, Duo Liu, Zhongqiang Zhang, Ge Liu, Xiaozhe Yang, and Xiangzhong Fang. 2024. NER-guided comprehensive hierarchy-aware prompt tuning for hierarchical text classification. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 12117–12126, Torino, Italia. ELRA and ICCL.

Haibin Chen, Qianli Ma, Zhenxi Lin, and Jiangyue Yan. 2021. Hierarchy-aware label semantics matching network for hierarchical text classification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*.

Zhongfen Deng, Hao Peng, Dongxiao He, Jianxin Li, and Philip Yu. 2021. Htcinfomax: A global model for hierarchical text classification via information maximization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Siddharth Gopal and Yiming Yang. 2013. Recursive regularization for large-scale classification with hierarchical and graphical dependencies. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738.

Ke Ji, Yixin Lian, Jingsheng Gao, and Baoyuan Wang. 2023. Hierarchical verbalizer for few-shot hierarchical text classification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2918–2933, Toronto, Canada. Association for Computational Linguistics.

Ting Jiang, Deqing Wang, Leilei Sun, Zhongzhi Chen, Fuzhen Zhuang, and Qinghong Yang. 2022. Exploiting global and local hierarchies for hierarchical text classification. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4030–4039, Abu Dhabi, United

Arab Emirates. Association for Computational Linguistics.

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

Kamran Kowsari, Donald E. Brown, Mojtaba Heidarysafa, K. Meimandi, Matthew S. Gerber, and Laura E. Barnes. 2017b. Hdltex: Hierarchical deep learning for text classification. *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 364–371.

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.

Yixin Liu, Pengfei Liu, Dragomir Radev, and Graham Neubig. 2022. BRIO: Bringing order to abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2890–2903, Dublin, Ireland. Association for Computational Linguistics.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press, USA.

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)*.

Lin Pan, Chung-Wei Hang, Avirup Sil, and Saloni Potdar. 2022. Improved text classification via contrastive adversarial training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11130–11138.

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 '18*.

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*.

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

Kazuya Shimura, Jiyi Li, and Fumiyo Fukumoto. 2018a. 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*, pages 811–816, Brussels, Belgium. Association for Computational Linguistics.

Kazuya Shimura, Jiyi Li, and Fumiyo Fukumoto. 2018b. 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*.

Carlos N Silla and Alex A Freitas. 2011. A survey of hierarchical classification across different application domains. *Data mining and knowledge discovery*, 22:31–72.

Robert Tarjan. 1972. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160.

Celine Vens, Jan Struyf, Leander Schietgat, Sašo Džeroski, and Hendrik Blockeel. 2008. Decision trees for hierarchical multi-label classification. *Machine learning*, 73:185–214.

Boyan Wang, Xuegang Hu, Peipei Li, and S Yu Philip. 2021a. Cognitive structure learning model for hierarchical multi-label text classification. *Knowledge-Based Systems*, 218:106876.

Dong Wang, Ning Ding, Piji Li, and Haitao Zheng. 2021b. Cline: Contrastive learning with semantic negative examples for natural language understanding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*.

Zihan Wang, Peiyi Wang, Lianzhe Huang, Xin Sun, and Houfeng Wang. 2022a. Incorporating hierarchy into text encoder: a contrastive learning approach for hierarchical text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7109–7119, Dublin, Ireland. Association for Computational Linguistics.

Zihan Wang, Peiyi Wang, Lianzhe Huang, Xin Sun, and Houfeng Wang. 2022b. Incorporating hierarchy into text encoder: a contrastive learning approach for hierarchical text classification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7109–7119, Dublin, Ireland. Association for Computational Linguistics.

Zihan Wang, Peiyi Wang, Tianyu Liu, Binghuai Lin, Yunbo Cao, Zhifang Sui, and Houfeng Wang. 2022c. HPT: Hierarchy-aware prompt tuning for hierarchical text classification. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3740–3751, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jonatas Wehrmann, RodrigoC. Barros, and Ricardo Cerri. 2018a. Hierarchical multi-label classification networks. *International Conference on Machine Learning,International Conference on Machine Learning*.

Jonatas Wehrmann, Ricardo Cerri, and Rodrigo Barros. 2018b. Hierarchical multi-label classification networks. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 5075–5084. PMLR.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*.

Jiawei Wu, Wenhan Xiong, and William Wang. 2019. Learning to learn and predict: A meta-learning approach for multi-label classification. *Cornell University - arXiv,Cornell University - arXiv*.

Zhuofeng Wu, Sinong Wang, Jiatao Gu, Madian Khabsa, Fei Sun, and Hao Ma. 2020. Clear: Contrastive learning for sentence representation. *arXiv preprint arXiv:2012.15466*.

Chao Yu, Yi Shen, and Yue Mao. 2022. Constrained sequence-to-tree generation for hierarchical text classification. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 1865–1869, New York, NY, USA. Association for Computing Machinery.

Meng Yu, Chenyan Xiong, Payal Bajaj, Saurabh Tiwary, Paul Bennett, Jiawei Han, and Xinshan Song. 2021. Coco-lm: Correcting and contrasting text sequences for language model pretraining. *Cornell University - arXiv,Cornell University - arXiv*.

Simon Chi Lok Yu, Jie He, Victor Basulto, and Jeff Pan. 2023. Instances and labels: Hierarchy-aware joint supervised contrastive learning for hierarchical multi-label text classification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8858–8875, Singapore. Association for Computational Linguistics.

Jun Zhang, Yubin Li, Fanfan Shen, Chenxi Xia, Hai Tan, and Yanxiang He. 2024. Hierarchy-aware and label balanced model for hierarchical text classification. *Knowledge-Based Systems*, 300:112153.

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.