LAKA: A Label-Aware and Knowledge-Augmented Framework for Multi-Label Text Classification

Anonymous ACL submission

Abstract

Multi-label text classification (MLTC) predicts multiple labels for a single text instance, with strong correlations between and within labels and text. Previous methods of MLTC are not effective enough to capture label-specific information and suffer from insufficient interaction between labels and text features. In this work, we propose **L**abel-**A**ware and **K**nowledge-**A**ugmented (LAKA) Framework for MLTC, strengthening the attention to relations between key information and labels via introducing structured information, mitigating the mentioned problems. Specifically, we introduce Multi-Label Text Knowledge Augmentation (MTKA), extracting external information via a external information extractor which based on graph neural network (GNN) and augmenting the backbone model with external knowledge fusion (EKF). Meanwhile, we introduce label-aware union embedding (LAUE) to strengthen interactions between and within labels and text by encoding labels and text together for BERT. Experiments on three datasets demonstrate that our method significantly outperforms strong baselines in various mainstream MLTC datasets.

1 Introduction

MLTC, as a sub-task of text classification (TC) in the field of natural language processing (NLP), aims at predicting multiple labels for a single instance input. Since the number of labels of a single instance is not constant and drawn from one up to hundreds, multiple challenges existed, including output logic, semantic capturing, and class imbalances. Recently, most MLTC researches kept an eye on designing better output logic, different from that, our research designed a framework that improves both output logic and semantic capturing.

Previous standard methods for MLTC predict labels independently. However, there are always relations between labels, one specific label can be strongly associated with multiple labels. Thus, we take advantage of transformers and introduce union embedding (LAUE), which enables the backbone transformer-based model to capture relations between and within labels and text.

Modern transformer-based pre-trained language models (PLMs) have brought enormous benefits to multiple tasks in NLP including MLTC, relation extraction (RE), and named entity recognition (NER). Besides parallel computing and representation learning, transformer excels in capturing dependency between tokens and dealing with sequence data. Thus, to utilize these advantages, our work introduced union embedding with labels and text, which cast multi-label prediction to span-prediction.

However, proper guidance and external knowledge for PLMs to adapt to specific domains and tasks is the key to success in downstream tasks (Gururangan et al., 2020; Peters et al., 2019). For MLTC, which faces challenges of domain-specific and high number of classes, effectively integrating external knowledge would help the model to excel. Therefore, inspired by former research, we propose a MLTC framework, in which we designed a graph neural network (GNN) to extract external knowledge and introduced a external knowledge fusion (EKF) layer to help PLMs adapt to the knowledge.

The major contributions of this paper can be summarized as follows:

- We propose a label-aware and knowledge-augmented framework for MLTC, strengthening the attention to relations between key information and labels via introducing structured external information.

- MTKA is introduced to extract external information via a GNN-based external information extractor and augments the backbone model via EKF, while the label-aware union

embedding enhances interactions between and within labels and text.

- Extensive experimentation validates the superior performance of our model across MLTC datasets, while detailed analysis underscores the effectiveness of its individual modules.

2 Related Work

2.1 Multi-label Text Classification

MLTC is a pivotal task within TC, holding significant importance in the field of NLP (Lin et al., 2022). MLTC involves instances comprising text and multiple associated classification labels, with correlations existing among these labels. Successful label prediction requires models to consider both the semantic information within the text and the relationships between labels. Over time, various MLTC methods have been introduced to address this challenge.

Notably, Kim (2014) incorporated convolutional neural networks (CNNs) into TC, presenting the TextCNN model. While successful among models of its era, TextCNN faced limitations due to its neglect of global semantic information. Subsequently, Nam et al. (2017) transformed the MLTC problem into a sequence-to-sequence problem, then Yang et al. (2018) introduced sequence generation models to learn the potential correlation among labels. The advent of transformers and the encoder-decoder framework further influenced MLTC methods, inspiring the development of transformer-based approaches. These models have enhanced semantic understanding and concurrently reduced the computational expenses associated with training (Zhang et al., 2021; Jiang et al., 2021; Gong et al., 2020). Moreover, the Joint Embedding of Words and Labels (JEWL) method has emerged as an effective approach for MLTC, successfully integrating both label and text semantic information (Wang et al., 2018). Recently, Alhuzali and Ananiadou (2021) proposed a framework that cast MLTC prediction as span prediction, which can aid backbone models to learn associations between labels and text in a sentence. This method has been demonstrated to be highly effective and serves as the foundation for our current work.

2.2 MLTC with Graph Neural Network

GNNs have gained prominence in NLP tasks, including MLTC (Zhou et al., 2020).

Yao et al.'s (2019) TextGCN showcased the effective use of graph convolutional networks (GCN) in capturing intricate correlations among words and sentences. Pal et al. (2020) introduced graph attention networks (GAT) for global label relationship modeling. Liu et al. (2016) proposed Text Level GCN with a graph information transmitting mechanism to update nodes, reducing parameter scale. Zheng et al. (2022) proposed label-dividing gated graph neural network (LD-GGNN), which enhances sibling label distinction and enables adaptive text-label interaction, reflecting the evolving landscape of GNNs in MLTC. These researches applies GNN to MLTC directly, which inspired our work to make use of GNN as an external information extractor.

2.3 Knowledge-Augmented Pre-trained Language Model

Incorporating external knowledge into PLMs is a pivotal strategy for enhancing their performance in downstream tasks (Kang et al., 2022). Consequently, a multitude of research has been proposed to integrate knowledge into PLMs. These approaches encompass both incorporating knowledge during pretraining and introducing knowledge post-pretraining.

Namely, for incorporating knowledge during pre-training, Zhang et al. (2019) and Peters et al. (2019) proposed ERNIRE and KnowBERT respectively, which introduced entities as additional inputs in the pre-training phase. Moreover, Qin et al. (2021) proposed ERICA, in which they defined two novel pre-training tasks and incorporated contrastive learning during the pre-training phase to enhance natural language understanding (NLU) of PLMs. Though these methods achieve successes in multiple tasks, they requires additional parameters for knowledges and high computational resources for pre-training. In contrast, Kang et al. (2022) proposed knowledge-augmented language model adaptation (KALA) as a solution to address these challenges. KALA involves adjusting the intermediate hidden representations of pre-trained language models (PLMs) by incorporating domain knowledge, specifically entities and their relational facts.

Thus, for MLTC, in which both structured and domain-oriented knowledge is crucial, we introduced external information via GNN-based external information extractor and EKF.

3 Methodology

The proposed LAKA framework for MLTC has been demonstrated in Figure 1. In general, the framework utilizes BERT as the base encoder, by applying LAUE, this model accepts labels and text simultaneously and outputs the scores of labels by applying a fully connected layer. In this way, the encoder enables the representation generation of labels and text to interact in the same semantic space, considering correlations between tokens. Additionally, the encoder is enhanced by MTKA, which fuses information from external knowledge sources by utilizing the EKF layer.

3.1 Label-Aware Union Embeddings

BERT, as a powerful transformer-based pre-trained language model to capture contextual semantic information and cross-label correlations, is adopted as the basic encoder in the proposed framework.

Different from other common usages, our framework does not extract the hidden states of [CLS] token for subsequent classification, instead, we cast the classification task to the span-prediction task, which helps the model to generate label-aware embeddings to capture correlations between and within labels and text. In the proposed framework, the label and text sequences are concatenated together as a single-sequence input, as known as the label-aware union embeddings (LAUE) of labels and text. Therefore, the hidden states of this sequence can be extracted from the encoder, which denotes as $H_i = \{h_{[CLS]}, h_{y_1}, ..., h_{y_m}, h_{[SEP]}, h_{x_1}, ..., h_{x_n}, h_{[SEP]}\}$. Also, the following equations show the encoding process.

$$Y_i = [y_1^i, ..., y_m^i] \tag{1}$$

$$X_i = [x_1^i, ..., x_n^i] \tag{2}$$

$$H_i = \text{BERT}(\{t_{[CLS]}||Y_i||t_{[SEP]}||X_i||t_{[SEP]}\}) \tag{3}$$

where $t_{[CLS]}$ and $t_{[SEP]}$ represent two special tokens of BERT, $||$ donotes the concatenation operation between token and token list.

Specifically, Y_i and X_i denote the i-th label token sequence and text token sequence respectively, m and n represents the corresponding length of the label token sequence and text token sequence.

3.2 Multi-Label Text Knowledge Augmentation

Since external knowledge can bring enormous improvement to language models, we propose Multi-Label Text Knowledge Augmentation (MTKA), enhancing the model's attention to crucial label-specific information.

3.2.1 External Entity Knowledge Extraction

As mentioned above, external knowledge can bring enormous improvement to language models. The entity itself contains rich information, such as name, attributes, and contextual information, providing crucial distinguishable features to the model. When these entities appear in different instances in the same way, the model captures these patterns and benefits from these entity information for identification and classification tasks.

To extract these entities and relations between entities, inspired by Kang et al. (2022), we applied the package spaCy[1] and Entity Linker[2], and constructed the entity relation graph.

In order to accomplish effective information aggregation for the target entity, we introduced the idea of GAT to allocate different attention scores to different neighbor nodes, which measure the importance of neighbor entities. Therefore, the attention scores of neighbour nodes can be calculated as:

$$a_{i,j} = \frac{\exp(\alpha^\tau \delta(W[X_i||X_j||e_{i,j}]))}{\sum_{k \in N(i) \cup \{i\}} \exp(\alpha^\tau \delta(W[X_i||X_k||e_{i,k}]))} \tag{4}$$

where α^τ is a learnable parameter, δ is the LeakyReLU activation function, W is a linear transformation weight matrix, $||$ denotes vector concatenation, X_i,X_j are embeddings of entity node i and node j, and $e_{i,j}$ is the relation embeddings between the entity nodes, and $N(i)$ stands for all neighboring nodes of node i.

In order to strengthen the connection between the entity hidden information h_{x_i} and the graph entity information X_i , we jointly consider the entity hidden information in the updating process of the graph entity information. The formula for iteration of graph entity node information is denoted as follows:

$$\Gamma(\alpha, \beta) = a_{\alpha,\beta}(W_1 X_\beta + W_2 h_{x_\beta}) \tag{5}$$

[1]https://spacy.io/

[2]https://github.com/egerber/spaCy-entity-linker

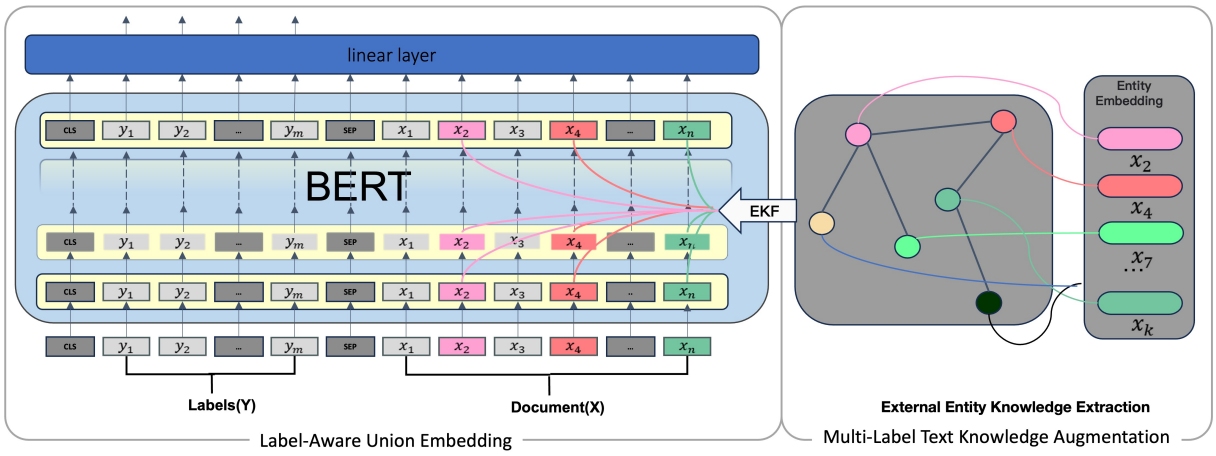

Figure 1: Overview of the proposed framework LAKA for MLTC.

$$X'_i = \Gamma(i,i) + \sum_{j \in N(i)} \Gamma(i,j) \tag{6}$$

where $a_{\alpha,\beta}$ is the attention score of node α with its neighbor node β, W_1,W_2 are linear transformation weight matrices, X_β is the initial entity embedding of entity node β, and h_{x_β} is the entity hidden information vectors corresponding to entity nodes β. Moreover, X'_i is the updated embedding of entity node i while $N(i)$ stands for all the neighboring nodes of node i.

By utilizing the graph attention mechanism, the entity node effectively aggregate features of its neighbors by allocating different attention scores to neighbor nodes. Also, in this process, the entity hidden information is considered to strengthen the connections between entity hidden information and graph entity information.

3.2.2 External Knowledge Fusion

Scaling and panning original features based on external knowledge is an effective method to fuse external knowledge to the original embedding with no modifications to the original model structure (Dumoulin et al., 2018). This feature-level affine transformation method is effective in a variety of tasks, e.g., style transformation in image generation (Huang and Belongie, 2017), linguistic conditional image inference (Perez et al., 2017), and named entity tasks (Dumoulin et al., 2018).

The proposed framework has a EKF layer which transform the entity hidden states in the BERT, this transformation is conditioned on the external knowledge to accomplish external knowledge fusion. The following formulas demonstrates the transformation process:

$$\hat{H}^l = A \cdot \mathrm{LN}(H^{l-1} + \mathrm{Attn}(H^{l-1})) + B \tag{7}$$

$$\mathrm{FFN}(\hat{H}^l) = \sigma(\hat{H}^l \cdot W_1) \cdot W_2 \tag{8}$$

$$\tilde{H}^l = \tilde{A} \cdot \mathrm{LN}(\hat{H}^l + \mathrm{FFN}(\hat{H}^l)) + \tilde{B} \tag{9}$$

where H^{l-1} denotes the hidden states from the former encoder layer while Attn, LN, FFN represents the multi-head attention, layer normalization, and feed-forward neural network respectively. Additionally, σ denotes the activation function, and W_1, W_2 represents two weight matrices. A,B,$\tilde{A}$, and $\tilde{B}$ are derived from multiple multi-layer perceptron, which can be calculated as follows:

$$\alpha_k = 1 + h_1(v_k) \tag{10}$$

$$\beta_k = h_2(v_k) \tag{11}$$

$$\tilde{\alpha_k} = 1 + h_3(v_k) \tag{12}$$

$$\tilde{\beta_k} = h_4(v_k) \tag{13}$$

where v_k denotes the graph embedding of the k-th entity, h_1, h_2, h_3, h_4 are multi-layer perceptrons which independent from each other.

The proposed framework extracts the external knowledge by constructing entity relation graph and aggregates information to the target entity. Subsequently, the hidden states of PLM is augmented with external knowledge by applying conditional transformation. Thus, the framework captures label-specific text features by strengthen connections between these features.

3.3 Training Objective

We use binary cross entropy loss as the loss function to calculate the loss value as follows:

$$L = -\sum_{i=1}^{N}\sum_{j=1}^{l} y_{ij}\log(\hat{y}_{ij}) + (1-y_{ij})\log(1-\hat{y}_{ij}) \tag{14}$$

where N is the number of documents, l is the number of labels, $\hat{y}_{ij}$ and y_{ij} are the predicted and true values of the j-th label for the i-th instance, respectively.

4 Experiments

4.1 Datasets and Baselines

Datasets. For the experiment, we employ three benchmark datasets designed for multi-label text classification to assess the classification performance of our model.

AAPD (Yang et al., 2018): The dataset consists of abstracts from 55,840 academic papers and their corresponding subject categories, with each academic paper belonging to one or more disciplines, totaling 54 disciplines in all.

ArXiv-ACM (Fallah et al., 2023): The dataset comprises 13,917 abstracts from computer science literature and their corresponding subject categories, with a total of 64 different categories.

Reuters-21578 [3]: The dataset is composed of news articles from Reuters and is a commonly used dataset for classification within the Reuters Financial News service.

We follow the approach of (Fallah et al., 2023) in dividing the aforementioned dataset into training, validation, and test sets, with the statistical data for each set illustrated in Table 1.

Baselines. We compared our method with some of the state-of-the-art models, including Bert(Devlin et al., 2018), SpanEmo (Alhuzali and Ananiadou, 2021), GCN_MTC (Vu et al., 2022) and LR_GCN (Vu et al., 2023). Specifically, SpanEmo approaches the multi-label classification problem as a span prediction issue by jointly embedding labels and texts to better facilitate the model's learning of the associations between labels and the words in sentences, and introduces a label-aware attention loss. GCN_MTC presents a label correlation matrix based on normalization and filtering mechanisms, and integrates the learned label relevance information with fine-grained document information for classification. LR_GCN models label correlations using label co-occurrence information and leverages external information to enhance the semantic representation of labels. It then combines the label relevance information learned from the GCN with fine-grained document representations generated from another subnet for classification.

[3]https://www.kaggle.com/nltkdata/reuters

4.2 Experimental Settings

We used PyTorch (Paszke et al., 2017) for implementation and ran all experiments on RTX8000 with 48G memory. We use the English base-uncased version of BERT as the base encoder,with a batch size of 32, a dropout rate of 0.1, and the number of epochs set to 30. Additionally, we employ an Adam optimizer with weight decay set at 0.01, and we set the learning rate for the BERT encoder to 5e-5. For the graph attention network, we set its learning rate to 3e-5, the number of layers to 2, and the number of attention heads to 7. For the multilayer perceptron , we use two linear layers with ReLU activation functions. Additionally, we set the dimension of entity embeddings to 768 and initialize them using bert-base-uncased. For relation embeddings, we perform random initialization with the dimension size set to 128. We employ evaluation metrics such as macro F1-score, micro F1-score, and Jaccard index score to assess the performance of the model. Table 4 in appendix A summarizes the hyper-parameters used in our experiments.

4.3 Experimental Results

As shown in Table 2, our method outperforms the baseline models in overall performance across all datasets. Compared to the baseline models, our method is particularly outstanding in terms of macro F1-score, achieving increases of 1.68%, 2.67%, and 0.76% on the AAPD, ArXiv-ACM, and Reuters-21578 datasets, respectively, over the best benchmark (SpanEmo). Although our method is slightly inferior to the LR_GCN benchmark on the micro F1-score for the ArXiv-ACM dataset and on the Jaccard index score for the Reuters-21578 dataset, our approach still holds an advantage in terms of overall performance.

Moreover, it is of significant note that our method, as well as the SpanEmo benchmark that utilizes union embedding, consistently outperforms methods solely reliant on individual text feature extraction (e.g., Bert) or dual independent models

Dataset	Class	Avg. label	Avg.word	Train	Valid	Test
AAPD	54	2.41	163.43	53840	1000	1000
ArXiv-ACM	64	2.33	152	9600	2157	2160
Reuters-21578	90	1.24	127.76	6770	1000	3019

Table 1: Data Statistics.

Model	AAPD			ArXiv-ACM			Reuters-21578		
	maF1	miF1	Jaccard	maF1	miF1	Jaccard	maF1	miF1	Jaccard
Bert	0.5688	0.7166	0.6436	0.2846	0.5485	0.4483	0.5116	0.8962	0.9089
SpanEmo	0.5934	0.7289	0.6553	0.3831	0.5602	0.4644	0.6495	0.9050	0.9165
GCN_MTC	0.5556	0.7253	0.6441	0.3020	0.5466	0.4460	0.5688	0.9020	0.9150
LR_GCN	0.5743	0.7260	0.6544	0.3628	**0.5702**	0.4725	0.5977	0.9078	**0.9208**
Ours	**0.6102**	**0.7374**	**0.6647**	**0.4098**	0.5694	**0.4729**	**0.6572**	**0.9086**	0.9169

Table 2: Comparisons among different multi-label text classifications on three datasets. The best results are in bold, while maF1, miF1, and Jaccard represents macro F1, micro F1, and Jaccard index score respectively.

Dataset	Model	maF1	miF1	Jaccard
AAPD	Ours	**0.6102**	**0.7374**	**0.6647**
	w/o MTKA	0.589	0.7177	0.6484
	w/o LAUE	0.5749	0.7208	0.6462
	w/o MTKA & LAUE	0.5688	0.7166	0.6436
ArXiv-ACM	Ours	**0.4098**	**0.5694**	**0.4729**
	w/o MTKA	0.3957	0.5575	0.4564
	w/o LAUE	0.3172	0.5561	0.4557
	w/o MTKA & LAUE	0.2846	0.5485	0.4483
Reuters-21578	Ours	**0.6572**	**0.9086**	**0.9169**
	w/o MTKA	0.6259	0.9021	0.9127
	w/o LAUE	0.5524	0.9019	0.9155
	w/o MTKA & LAUE	0.5116	0.8962	0.9089

Table 3: Ablation experiment results.

for extracting features from labels and texts (e.g., GCN_MTC and LR_GCN), in terms of macro F1-score across all datasets. This corroborates the advantage of union embedding in aligning labels and texts within a unified semantic space, thereby enhancing the interaction between label and textual features—particularly beneficial for capturing label-specific text features within minority classes. In comparison with the SpanEmo benchmark, our method exhibits superior performance on all evaluated metrics across all datasets, thereby proving the efficacy of integrating textual entity relationship information into the framework of union embedding. Compared to GCN_MTC, LR_GCN exhibits enhanced semantic label representation by incorporating external information, leading to performance improvements and optimal results in specific datasets. However, our approach surpasses this by eliminating the need for external information resources. The extraction of external entity knowledge in our method is derived directly from raw data text, providing a distinct advantage.

4.4 Ablation Study

In order to further discern the contribution of each element within our methodology, we undertook ablation studies on the performance of the model under three distinct configurations: initially, by

training the model devoid of Multi-Label Text Knowledge Augmentation (MTKA) ; secondly, by conducting training in the absence of LAUE; and thirdly, by training without both MTKA and LAUE. The latter configuration serves as a baseline and is comparable to the Bert model.

As shown in Table 3, when our method is trained without MTKA, the macro F1-score, micro F1-score, and Jaccard index score decreased by 1.42-3.13%, 0.65-1.97%, and 0.42%-1.65%, respectively. This demonstrates that by incorporating entity relationship information to strengthen the connections between text features, the model can capture label-specific text features more effectively. Additionally, training without LAUE resulted in a greater decrease across all evaluation metrics, particularly the macro F1-score for the ArXiv-ACM and Reuters-21578 datasets, indicating that the Union embedding approach is especially effective for minority classes in datasets with a smaller number of samples. It is also noteworthy that our model's performance without Union embeddings still surpasses that of the method lacking both MTKA and LAUE (i.e., Bert), indirectly affirming the effectiveness of the MTKA module within our model.

4.5 Limitations

Though our framework excels, there are some limitations that should be addressed. Firstly, our framework may not be suitable for short texts due to its reliance on capturing relationships between information and labels, which may require longer context to be effective. Secondly, while our framework shows promising results, further exploration is needed to improve its performance on imbalanced classes, as it may struggle with classes that have significantly fewer samples compared to others. Addressing these limitations could further enhance the applicability and effectiveness of our proposed framework in practical MLTC scenarios.

5 Conclusion

Our work introduces the label-aware and knowledge-augmented framework for MLTC, enhancing attention to the relations between key information and labels by incorporating structured external information. The introduction of MTKA facilitates the extraction of external information through a GNN-based extractor and augments the backbone model via EKF. Additionally, the label-aware union embedding enhances interactions both between and within labels and text. Through extensive experiments, our framework demonstrates superior performance across MLTC datasets, with detailed analysis highlighting the effectiveness of its individual modules.

References

Hassan Alhuzali and Sophia Ananiadou. 2021. Spanemo: Casting multi-label emotion classification as span-prediction. *arXiv preprint arXiv:2101.10038*.

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*.

Vincent Dumoulin, Ethan Perez, Nathan Schucher, Florian Strub, Harm de Vries, Aaron Courville, and Yoshua Bengio. 2018. Feature-wise transformations. *Distill*, 3(7):e11.

Haytame Fallah, Emmanuel Bruno, Patrice Bellot, and Elisabeth Murisasco. 2023. Exploiting label dependencies for multi-label document classification using transformers. In *Proceedings of the ACM Symposium on Document Engineering 2023*, pages 1–4.

Jibing Gong, Zhiyong Teng, Qi Teng, Hekai Zhang, Linfeng Du, Shuai Chen, Md Zakirul Alam Bhuiyan, Jianhua Li, Mingsheng Liu, and Hongyuan Ma. 2020. Hierarchical graph transformer-based deep learning model for large-scale multi-label text classification. *IEEE Access*, 8:30885–30896.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Xun Huang and Serge J. Belongie. 2017. Arbitrary style transfer in real-time with adaptive instance normalization. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1510–1519.

Ting Jiang, Deqing Wang, Leilei Sun, Huayi Yang, Zhengyang Zhao, and Fuzhen Zhuang. 2021. Lightxml: Transformer with dynamic negative sampling for high-performance extreme multi-label text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7987–7994.

Minki Kang, Jinheon Baek, and Sung Ju Hwang. 2022. KALA: knowledge-augmented language model adaptation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5144–5167, Seattle, United States. Association for Computational Linguistics.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Nankai Lin, Sihui Fu, Xiaotian Lin, and Lianxi Wang. 2022. Multi-label emotion classification based on adversarial multi-task learning. *Information Processing & Management*, 59(6):103097.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Recurrent neural network for text classification with multi-task learning. *arXiv preprint arXiv:1605.05101*.

Jinseok Nam, Eneldo Loza Mencía, Hyunwoo J Kim, and Johannes Fürnkranz. 2017. Maximizing subset accuracy with recurrent neural networks in multi-label classification. *Advances in neural information processing systems*, 30.

Ankit Pal, Muru Selvakumar, and Malaikannan Sankarasubbu. 2020. Multi-label text classification using attention-based graph neural network. *arXiv preprint arXiv:2003.11644*.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron C. Courville. 2017. Film: Visual reasoning with a general conditioning layer. In *AAAI Conference on Artificial Intelligence*.

Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. 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 43–54, Hong Kong, China. Association for Computational Linguistics.

Yujia Qin, Yankai Lin, Ryuichi Takanobu, Zhiyuan Liu, Peng Li, Heng Ji, Minlie Huang, Maosong Sun, and Jie Zhou. 2021. ERICA: Improving entity and relation understanding for pre-trained language models via contrastive learning. 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)*, pages 3350–3363, Online. Association for Computational Linguistics.

Huy-The Vu, Minh-Tien Nguyen, Van-Chien Nguyen, Minh-Hieu Pham, Van-Quyet Nguyen, and Van-Hau Nguyen. 2023. Label-representative graph convolutional network for multi-label text classification. *Applied Intelligence*, 53(12):14759–14774.

Huy-The Vu, Minh-Tien Nguyen, Van-Chien Nguyen, Manh-Tran Tien, and Van-Hau Nguyen. 2022. Label correlation based graph convolutional network for multi-label text classification. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 01–08. IEEE.

Guoyin Wang, Chunyuan Li, Wenlin Wang, Yizhe Zhang, Dinghan Shen, Xinyuan Zhang, Ricardo Henao, and Lawrence Carin. 2018. Joint embedding of words and labels for text classification. *arXiv preprint arXiv:1805.04174*.

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

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Graph convolutional networks for text classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7370–7377.

Ximing Zhang, Qian-Wen Zhang, Zhao Yan, Ruifang Liu, and Yunbo Cao. 2021. Enhancing label correlation feedback in multi-label text classification via multi-task learning. *arXiv preprint arXiv:2106.03103*.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

Siqi Zheng, Jie Zhou, Kui Meng, and Gongshen Liu. 2022. Label-dividing gated graph neural network for hierarchical text classification. In *2022 International Joint Conference on Neural Networks (IJCNN)*, pages 01–08. IEEE.

Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2020. Graph neural networks: A review of methods and applications. *AI open*, 1:57–81.

A Hyper-parameter Settings

Parameter	Value
Batch size	32
Dropout	0.1
Number of epochs	30
Weight decay	0.01
lr-BERT	0.00005
lr-GAT	0.00003
Number of layers in GAT	2
Number of attention heads in GAT	7
Entity embedding dimension	768
Relation embedding dimension	128

Table 4: Hyper-parameter values, where lr refers to the learning rate.