Taming Prompt-based Data Augmentation for Long-Tailed Extreme Multi-Label Text Classification

Anonymous EMNLP submission

Abstract

Extreme multi-label text classification (XMC) involves tagging a document with its most relevant subset of labels from a large label set. In real applications, labels usually follow a long-tailed distribution, where most labels (called tail-labels) only contain a small number of documents and limit the performance of XMC. Data augmentation (DA) is a simple but effective strategy to solve such low-resource problems. However, most existing DA approaches struggle in extreme multi-label settings. The augmented samples for one label may inevitably influence the other co-occurring labels and further exacerbate the long-tailed problem. Moreover, the presence of a large label space leads to label confusion, resulting in low-quality augmented samples after DA. To mitigate these issues, we propose a prompt-based DA method called XDA, which is specifically designed for XMC. First, we employ a soft prompt during the fine-tuning process of the T5 model for label-conditional DA, thereby enabling T5 to augment samples while preserving label-compatibility. Subsequently, XDA performs sample filtering on the augmented samples through the diversity of text and the consistency of labels, which enhances the quality of the DA. In contrast to traditional *sample-level* DA, we propose a *pair-level* DA method by masking the augmented sample-label pairs of head-labels during training, effectively mitigating the long-tailed problem. Comprehensive experiments on benchmark datasets have shown that the proposed XDA outperforms the state-of-the-art counterparts.

1 Introduction

Extreme multi-label text classification (XMC) deals with the problem of predicting the most relevant subset of labels from an enormously large label space. It has a wide range of applications, such as product search (Chang et al., 2021), document tagging (Chalkidis et al., 2019), keyword recommendation (Chang et al., 2020), query autocompletion (Yadav et al., 2021), computational advertising (Gupta et al., 2021) and so on.

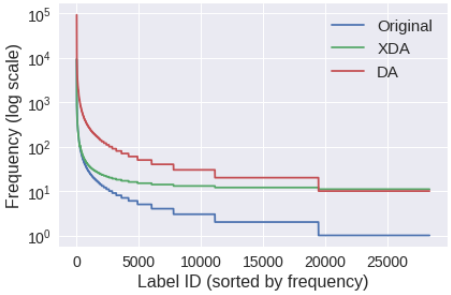

Figure 1: Wiki10-31K shows a long-tailed distribution of labels (denoted as Original). Only 10% of the labels have more than 10 training samples. DA denotes the distribution after augmenting 10 times for tail-labels directly, and XDA denotes the distribution after augmenting 10 times for tail-labels under the XDA framework. 100% of the labels have more than 10 training samples, which mitigates the data sparsity issue.

Even though various techniques have been proposed for XMC, it is still a challenging task due to the "long-tailed" label distribution (Chang et al., 2020; Zhang et al., 2023). Figure 1 illustrates the long-tailed label distribution in the Wiki10-31K dataset. Only 10% of the labels have more than 10 training samples (i.e., head-labels), while the remaining 90% are long-tail labels with much fewer training samples. In this situation, training classification models for the tail-labels is much more difficult than that for head-labels, which suffers severely from the lack of sufficient training samples.

One immediate approach to address the problem is data augmentation (DA) which can compensate the scarce data for tail-labels (Zhang et al., 2020, 2022). Several existing works (Anaby-Tavor et al., 2020; Wu et al., 2022; Zhou et al., 2022; Wang et al., 2022) resort to applying pre-trained language models (PLMs) for DA in a low-resource setting.

However, because of the label co-occurrence, existing approaches of DA struggle in the multi-label scenario (Wu et al., 2020; Zhang et al., 2022). A document usually contains several labels, making the selection for tail-labels no longer independent. For example, a document that contains tail-labels, e.g. "Neptune" and "Pluto", is likely to be also associated with the head-labels, e.g. "astronomy " and "physics". As shown in Figure 1, the long-tailed problem is not necessarily eliminated and may even be exaggerated by directly adopting DA methods.

Moreover, DA sometimes fails to guarantee semantic consistency, and may even bring semantic errors that are harmful to classification (Zhou et al., 2022; Zhao et al., 2022a). Especially in the XMC scenario, due to the complex semantics of multi-label texts, it is difficult to control the strength of semantic changes after augmentation, resulting in a certain amount of low-quality augmented samples.

In this paper, we propose XDA, a new approach that overcomes the aforementioned issues by effectively incorporating prompt-based DA method specifically designed for the XMC. XDA employs three mechanisms to alleviate the low-quality and long-tailed problems of augmented samples: *Prompt*, *Filter* and *Mask*. Figure 2 shows the pipeline of XDA.

Similar to previous works (Wang et al., 2022; Zhou et al., 2022), we use T5 (Raffel et al., 2020) to produce complete synthetic data conditioned on the labels and the masked text. Our approach enables T5 to augment samples while preserving label-compatibility. Notably, we only allow tuning the additional soft prompts during fine-tuning, which significantly reducing the amount of parameters to be tuned.

Generally, given the size of generated data, their diversity and quality are crucial to the performance of downstream tasks (Zhao et al., 2022a; Kamalloo et al., 2022a). Therefore, we perform sample filtering on the augmented samples through the diversity of text and the consistency of labels.

In contrast to traditional *sample-level* DA methods, we propose a novel masked data augmentation (MDA) approach focuses on *pair-level* DA by masking the augmented sample-label pairs of head-labels during training. This approach exclusively augments positive sample-label pairs for the tail-labels, thereby addressing the label sparsity issue without exacerbating the long-tailed problem, as demonstrated in Figure 1.

Our experiments show that XDA significantly and consistently outperforms state-of-the-art baselines on three benchmark datasets, especially on tail-labels. Our source code and hyper-parameter settings are released at `https://anonymous.4open.science/r/EMNLP-XDA`.

2 Method

As depicted in Figure 2, our method XDA consists of three components: *Prompt*, *Filter* and *Mask*. First, we freeze the entire pre-trained model and only allow tuning the additional soft prompts during fine-tuning to produce complete synthetic data conditioned on the output labels. Second, we filter out the low-quality samples in the generated sample set through the diversity of text and the consistency of labels. At last, we mask the head sample-label pairs of the augmented data in the training stage, which mitigates the label sparsity issue.

2.1 Extreme Multi-Label Text Classification

Let calligraphic letter (e.g., $\mathcal{A}$) indicates set, capital and lower-case letters (e.g. A, a) for scalars, lower-case bold letter (e.g., $\mathbf{a}$) for vector and capital bold letter (e.g., $\mathbf{A}$) for matrix. The input of training stage includes N instances $\mathcal{P} = \{(\mathbf{x}^i, \mathbf{y}^i)\}_{i=1}^N$, each of which consists of a document $\mathbf{x} \in \mathcal{D}$ and several labels $\mathbf{y} \in \{0,1\}^L$ related to the document. Here L is the total number of candidate labels. The goal of XMC is to learn a function $f : \mathcal{D} \times [L] \mapsto \mathbb{R}$, that maps the input document $\mathbf{x}$ and a label l to a relevance score $f(\mathbf{x}, l)$. In the testing stage, we aim to recommend the top k labels with the highest relevance scores for a new document. Following previous works, We constructed the scoring function using an A simple one-versus-all (OVA) approach:

$$f(\mathbf{x}, l) = \mathbf{w}_l^\top \phi(\mathbf{x});\ l \in [L], \tag{1}$$

where $\mathbf{w}_l$ is the trainable classifier parameter for the label l and $\phi : \mathcal{D} \mapsto \mathbb{R}^d$ is the text encoder that maps $\mathbf{x}$ to d-dimensional feature vector. Our backbone is the same as LightXML (Jiang et al., 2021), which relied on using the multi-layer features of the transformer model as the text representation. We then employed binary cross entropy (BCE) as the loss function:

$$\begin{aligned} L_{\text{BCE}} = -\sum_{l=1}^{L} & y_l \log(\sigma(f(\mathbf{x}, l))) \\ & + (1 - y_l) \log(1 - \sigma(f(\mathbf{x}, l))), \end{aligned} \tag{2}$$

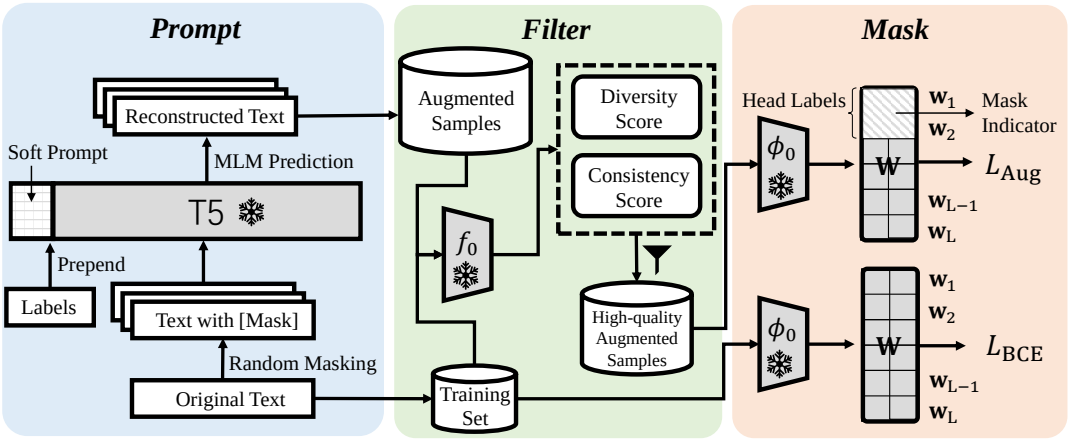

Figure 2: The architecture of the proposed XDA. The snowflake represents frozen parameters.

where σ denotes the activation function, such as the *Sigmoid*.

2.2 Prompt-based Data Augmentation

Prompt-based learning Fine-tuning is a conventional approach for adapting pre-trained language models (PLMs) to downstream tasks. However, as PLMs become larger in scale, directly fine-tuning them incurs significant resource consumption. Therefore, since the emergence of GPT3 (Brown et al., 2020) , researchers have introduced prompt-based learning as an alternative method. Nonetheless, the search for suitable prompts often necessitates additional human effort. Motivated by recent research (Lester et al., 2021; Wang et al., 2022), we employ the soft prompt technique for fine-tuning. This involves adding a sequence of trainable vectors $\mathbf{P}^j = \{\mathbf{p}_1^j, \cdots, \mathbf{p}_k^j\}$ at each transformer layer. In the training process, we exclusively update the parameters of the soft prompt while keeping all other PLM parameters fixed. The hidden states in the Transformer model is defined as:

$$\mathbf{h}_i^j = \begin{cases} \mathbf{p}_i^j & i \leq k \\ \mathbf{e}_i & i > k \wedge j = 0 \\ \mathcal{F}(\mathbf{h}^{j-1})_i & \text{Otherwise} \end{cases} \quad (3)$$

where $\mathbf{h}_i^j$ is the i-th hidden states at the j-th layer, and $\mathbf{e}_i$ is the word embedding layer. The forward function of the Transformer layer is denoted as $\mathcal{F}(\cdot)$. By adopting the soft prompt, we have adapted the PLM for downstream data augmentation tasks while effectively managing computational costs.

Label-Conditional Data Augmentation Previous works often use label information as auxiliary assistance during DA, such as label-conditional generation (Anaby-Tavor et al., 2020), and keywords-conditional generation (Wang et al., 2022). XDA is further utilizing output multi-labels as additional conditioning. Through this method, we strive to enhance the model's proficiency in predicting words in masked positions, taking into account both contextual information and label information.

In our DA process, we generate augmented documents N_a times based on the original ones. Our approach involves merging the text and labels into a single sequence. Then, we randomly mask a specific percentage of the input tokens. Afterward, we utilize the fine-tuned T5 model to predict and fill in the masked tokens to form a new sample.

2.3 Sample Filtering

Generally, given the size of generated data, their diversity and quality are crucial to the performance of targeted tasks (Zhao et al., 2022b). Especially in the XMC scenario, due to the complex semantics of multi-label texts, it is difficult to control the strength of semantic changes after augmentation, resulting in a certain amount of low-quality augmented samples. Given a pool of augmented samples, our approach is to select the best candidates according to the diversity of text and the consistency of labels. Initially, we employ the original training set $\mathcal{P}$ to perform pre-training of the XMC model f. Subsequently, our filtering approach is reliant on the feedback signal derived from the pre-trained f_0.

Diversity Intuitively, if there is significant semantic difference between an augmented sample $\mathbf{x}_i^j$ and its original corresponding sample $\mathbf{x}_i$, the augmented sample can be considered to have high diversity. Note that the XMC model f_0 is pretrained on training set $\mathcal{P}$. Therefore, we can directly use the loss function L_{BCE} to measure the diversity of the samples. We aim to obtain augmented samples with high diversity through the following approach:

$$\max_{\mathbf{x}_i^j} L_{\text{BCE}}(f_0(\mathbf{x}_i^j), \mathbf{y}^i)). \tag{4}$$

This approach also enables us to focus on selecting samples that are "hard" for XMC. Furthermore, we acknowledge that the loss function can be decomposed into:

$$\begin{aligned} L_{\text{BCE}}(f(\mathbf{x}_i^j), \mathbf{y}^i)) &= H(p(\mathbf{y}^i)) \\ &+ D_{\text{KL}}(p(f_0(\mathbf{x}_i^j))||p(\mathbf{y}^i)), \end{aligned} \tag{5}$$

where p, H, D_{KL} indicate probability distribution, Shannon entropy, and KL divergence (relative entropy) respectively. Furthermore, since $p(\mathbf{y}^i)$ is a one-hot vector, it results in $H(p(\mathbf{y}^i))$ being equal to zero. Therefore, in the end, we utilize KL divergence to measure the difference between the two distributions $p(f_0(\mathbf{x}_i^j))$ and $p(\mathbf{y}^i)$, that indicate the diversity of the augmented samples. Consequently, the diversity score $S_{\text{Div}}^{i,j}$ of the augmented sample $\mathbf{x}_i^j$ is defined as:

$$S_{\text{Div}}^{i,j} = D_{\text{KL}}(p(\sigma(\mathbf{W}_0^\top \phi_0(\mathbf{x}_i^j)))||p(\mathbf{y}^i)). \tag{6}$$

Consistency Low-quality augmented samples often lead to label drift (Kamalloo et al., 2022b; Zhou et al., 2022), i.e., the predicted labels of augmented samples are inconsistent with those of the original samples. Therefore, we ensure the consistency of labels by filtering out these inconsistent augmented samples. In the multi-label setting, we consider the top-k_i labels of sample $\mathbf{x}_i$, where k_i is the real number of labels, i.e., $k_i = \text{sum}(\mathbf{y}^i)$. And

$$|\text{Top}_{k_i}(f_0(\mathbf{x}_i^j)) \bigcap \text{Top}_{k_i}(f_0(\mathbf{x}_i))| \quad \leq \quad k_i, \tag{7}$$

where $\text{Top}_{k_i}(\cdot)$ denotes returns top-k_i indices based on the scores returned by f_0. Our objective is to ensure a close alignment between the top-k_i labels predicted by the augmented sample $\mathbf{x}_i^j$ and those assigned to the original sample $\mathbf{x}_i$. As a consequence, we define the consistency score $S_{\text{Con}}^{i,j}$ of augmented sample $\mathbf{x}_i^j$ as:

$$S_{\text{Con}}^{i,j} = \frac{1}{k_i}|\text{Top}_{k_i}(f_0(\mathbf{x}_i^j)) \bigcap \text{Top}_{k_i}(f_0(\mathbf{x}_i))|. \tag{8}$$

By jointly considering $S_{\text{Div}}^{i,j}$ and $S_{\text{Con}}^{i,j}$, high-quality samples with balanced diversity and consistency can be found.

2.4 Masked Data Augmentation

After performing the aforementioned DA steps, if we directly employ the augmented data for training the XMC model, it will inevitably exacerbate the long-tail problem, as illustrated in Figure 1. Hence, we introduce a novel masked data augmentation (MDA) approach focuses on *pair-level* DA by masking the augmented sample-label pairs associated with the head-labels during the training process. First, we divide the labels to head-labels and tail-labels according to the hyper-parameter, head-to-tail threshold $N_t \in \mathcal{R}^+$. label l is a tail-label if $n^l < N_t$. After that, we obtain a head-label set $\mathcal{H}$ and a tail-label set $\mathcal{T}$. Subsequently, we incorporate the mask indicator $\mathbf{m} = \{m_l\}_{l=1}^L$ into the loss function of the augmented samples to facilitate the implementation of MDA:

$$\begin{aligned} L_{\text{Aug}} = -\sum_{l=1}^{L} & m_l y_l \log(\sigma(f_0(\mathbf{x}, l))) \\ &+ (1 - y_l) \log(1 - \sigma(f_0(\mathbf{x}, l))). \end{aligned} \tag{9}$$

The mask indicator $\mathbf{m}$ is:

$$m_l = \begin{cases} 0 & l \in \mathcal{H} \\ 1 & \text{Otherwise} \end{cases} \tag{10}$$

where, m_l is the mask indicator for the label l and $\mathcal{H}$ is the head-label set. Specifically, during the training process, we keep the encoder $\phi_0(\cdot)$ frozen and solely fine-tune the classifier parameters $\mathbf{W}$. By adding L_{Aug} to L_{BCE}, we obtain a more balanced loss for the classifiers. In summary, MDA exclusively augments positive sample-label pairs for the tail-labels, thereby addressing the label sparsity issue without exacerbating the long-tailed problem.

3 Experiments

3.1 Experimental Setup

Datasets We evaluate XDA on 3 public XMC benchmark datasets: Eurlex-4K, Wiki10-31K, AmazonCat-13K. Table 1 contains the statistics

Datasets	N_{trn}	N_{tst}	L	L_{avg}	N_{avg}
Eurlex-4K	15,539	3,809	3,956	5.30	20.79
Wiki10-31K	14,146	6,616	30,938	18.64	8.52
AmazonCat-13K	1,186,239	306,782	13,330	5.04	448.57

Table 1: Data statistics. N_{trn}, N_{tst} refer to the number of documents in the training and test sets, respectively. L is the number of labels. L_{avg} is the average number of labels per documents. N_{avg} is the average number of documents per label. The three benchmark datasets are the same as DEPL (Zhang et al., 2023) for fair comparison.

	Eurlex-4K			Wiki10-31K			AmazonCat-13K		
Methods	P@1	P@3	P@5	P@1	P@3	P@5	P@1	P@3	P@5
DisMEC	83.21	70.39	58.73	84.13	74.72	65.94	93.81	79.18	64.06
PfastreXML	73.14	60.16	50.54	83.57	68.61	59.10	91.75	77.97	63.68
Parabel	82.12	68.91	57.89	84.19	72.46	63.37	93.02	79.14	64.51
Bonsai	82.30	69.55	58.35	84.52	73.76	64.69	92.98	79.13	64.46
AttentionXML	85.12	72.80	61.01	86.46	77.22	67.98	95.53	82.03	67.00
X-Transformer	85.46	72.87	60.79	87.12	76.51	66.69	95.75	82.46	67.22
XLNet-APLC	86.83	74.34	61.94	88.99	78.79	69.79	94.56	79.78	64.59
LightXML	86.12	73.87	61.67	87.39	77.02	68.21	94.61	79.83	64.45
DEPL	85.38	71.86	59.91	84.54	73.44	64.75	94.86	80.85	64.55
DEPL+c	86.43	73.77	62.19	87.32	77.05	67.39	96.16	82.23	**67.65**
XDA	**87.14**	**74.82**	**63.44**	**89.75**	**79.68**	**70.28**	**96.67**	**82.85**	67.40

Table 2: The all-label prediction results of representative classification systems evaluated in the micro-avg P@k metric. The bold phase and underscore highlight the best and second best model performance.

of these three datasets. For fair comparison, we use the same raw text input and same train-test split as DEPL (Zhang et al., 2023).

Evaluation Metric Following the settings of previous works (You et al., 2019; Jiang et al., 2021; Kharbanda et al., 2022; Zhang et al., 2023), We chose precision at k (P@k) as our evaluation metrics for performance comparison. We also examined the performance on tail-labels by propensity scored precision at k (PSP@k) (Jain et al., 2016). The definition of these metrics can be used to reference in Appendix A.3.

Implementation Details For all three datasets, we utilize raw texts without any preprocessing, and these texts are truncated to fit the maximum input token limit. These are the conventional setups for XMC methods (You et al., 2019; Jiang et al., 2021). Our model was trained by AdamW (Kingma and Ba, 2015), and we also used stochastic weight averaging (SWA) (Jiang et al., 2021) with a constant learning rate to avoid overfitting. To optimize GPU memory usage and enhance training, we employ automatic mixed precision (AMP). As for the key hyper-parameters of our proposed method: head-to-tail threshold N_t and times of augmentation N_a, we set $N_t = 50, N_a = 2$ for Eurlex-4K. For Wiki10-31K and AmazonCat-13K, we set $N_t = 10, N_a = 4$ and $N_t = 700, N_a = 2$ respectively. More details about the implementation setting can be found in Appendix A.4.

Baselines We compare XDA with state-of-the-art (SOTA) XMC methods including the one-versus-all DiSMEC (Babbar and Schölkopf, 2017); instance tree based PfastreXML (Jain et al., 2016); label tree based Parabel (Prabhu et al., 2018), Bonsai (Khandagale et al., 2020); RNN-based AttentionXML (You et al., 2019) and Transformer-based X-Transformer (Chang et al., 2020), XLNet-APLC (Ye et al., 2020), LightXML (Jiang et al., 2021), DEPL (Zhang et al., 2023) methods. We obtain most baseline results from (Zhang et al., 2023, Table 5) and (Zhang et al., 2023, Table 3). To ensure fair comparison, the experimental results of all comparative algorithms as well as our method are based on single-model predictions.

3.2 Main Results

The comparisons of P@k are shown in Table 2. The proposed XDA achieves new SOTA results in **8 out of 9** evaluation columns (combination of datasets and P@k).

Pre-trained language model (PLM) is crucial for XMC. Traditional machine learning methods (DiSMEC, PfastreXML, Parabel and Bonsai) perform relatively poorly as they utilize the bag-of-words

	Eurlex-4K			Wiki10-31K			AmazonCat-13K		
Methods	PSP@1	PSP@3	PSP@5	PSP@1	PSP@3	PSP@5	PSP@1	PSP@3	PSP@5
AttentionXML	44.20	50.85	53.87	14.49	15.65	16.54	53.94	68.48	76.43
X-Transformer	37.85	47.05	51.81	13.52	14.62	15.63	51.42	66.14	75.57
XLNet-APLC	42.21	49.83	52.88	14.43	15.38	16.47	52.55	65.11	71.36
LightXML	40.54	47.56	50.50	14.09	14.87	15.52	50.70	63.14	70.13
DEPL	45.60	52.28	53.52	16.30	16.26	16.27	55.94	70.01	76.87
DEPL+c	44.60	52.74	54.64	14.90	15.53	16.20	55.21	69.73	75.94
XDA	**46.43**	**54.15**	**56.97**	**16.95**	**17.09**	**17.51**	**56.36**	**71.40**	**77.11**
Improvement	*1.82%*	*2.67%*	*4.26%*	*3.99%*	*5.10%*	*5.86%*	*0.75%*	*1.99%*	*0.31%*

Table 3: Tail label prediction results of methods in PSP@k.

Dataset	Method	P@1	P@3	P@5	PSP@1	PSP@3	PSP@5
Eurlex-4K	Baseline	86.12	73.87	61.67	40.54	47.56	50.50
	EDA	85.80	73.64	61.70	41.10	49.37	52.26
	BackTrans.	86.68	74.17	62.61	43.49	52.32	54.20
	CBERT	86.56	73.82	61.76	41.74	50.38	51.56
	T5-MLM	86.42	73.93	62.59	43.37	52.24	53.61
	XDA	**87.14**	**74.82**	**63.44**	**46.43**	**54.15**	**56.97**
Wiki10-31K	Baseline	87.39	77.02	68.21	14.09	14.87	15.52
	EDA	87.49	77.14	68.81	15.60	15.97	16.39
	BackTrans.	88.95	78.50	69.17	17.11	17.50	18.32
	CBERT	88.26	77.50	68.33	16.36	16.88	17.38
	T5-MLM	88.33	78.34	68.67	16.89	17.15	18.41
	XDA	**89.75**	**79.68**	**70.28**	**16.95**	**17.09**	**17.51**

Table 4: Performance comparison with different DA algorithms on two datasets.

(BoW) features such as TF-IDF, which capture the word importance in a document to induce classification model. PLM-based methods (X-Transformer, XLNet-APLC, LightXML, DEPL and XDA) outperform traditional methods by a large margin on each dataset. This is not surprising since these methods introduce contextualized word embeddings that capture the rich semantic and syntactic information present in each word. This allows for more accurate and nuanced representations of the text, leading to improved performance in XMC task.

Addressing the long-tailed problem significantly enhances the performance of XMC tasks. As the most competitive baseline, DEPL+c also alleviates the data sparsity issue of tail-labels by matching the semantics between documents and augmented label descriptions. Our proposed method, XDA, takes an explicit data augmentation approach, directly mitigating the long-tailed problem in XMC tasks from a distributional perspective, resulting in better overall performance. Additionally, the backbone of our approach is LightXML. XDA achieves a significant performance improvement compared to LightXML, which serves as evidence of the effectiveness of our data augmentation framework.

3.3 Performance Analysis on Tail-Labels

To further verify the effectiveness of the proposed XDA in alleviating the long-tailed problem, we compare the performance of XDA with SOTA baselines by PSP@k. Table 3 shows XDA has remarkable improvement compared to the baselines in tail-labels classification.

The Wiki10-31K dataset presents the most skewed distribution, with only 10% of the labels having more than 10 training instances, resulting in a low PSP score. Since XDA relies on the high-quality augmented data specifically for tail-labels, it is less affected by the dominating training instances. Consequently, the PSP@k scores achieved by XDA surpassed those of the SOTA models by a significant margin.

An interesting point is that our proposed method, XDA, achieved larger performance improvements on two low-resource datasets, Eurlex-4K and Wiki10-31K. The reason behind this is the presence of a greater number of tail-labels in these datasets, making their long-tailed problem more severe compared to AmazonCat-13K. Our method effectively addresses this issue by leveraging PLM for DA, significantly increasing the number of samples for tail-labels and thus alleviating the long-tailed distri-

Dataset	Method	P@1	P@3	P@5	PSP@1	PSP@3	PSP@5
Eurlex-4K	Baseline	86.12	73.87	61.67	40.54	47.56	50.50
	XDA (w/o *Prompt*)	86.42	73.93	62.59	43.37	52.24	53.61
	XDA (w/o *Filter*)	86.83	74.34	62.84	45.37	52.63	54.96
	XDA (w/o *Mask*)	86.08	73.58	61.55	38.24	46.74	49.74
	XDA	87.14	74.82	63.44	46.43	54.15	56.97
Wiki10-31K	Baseline	87.39	77.02	68.21	14.09	14.87	15.52
	XDA (w/o *Prompt*)	88.63	78.71	68.85	16.89	16.79	16.96
	XDA (w/o *Filter*)	88.37	78.45	68.67	16.24	16.33	16.60
	XDA (w/o *Mask*)	86.77	76.48	67.33	13.71	14.12	15.27
	XDA	89.75	79.68	70.28	16.95	17.09	17.51

Table 5: Ablation test of XDA on two datasets.

bution in the data. However, Most SOTA methods tend to ignore the data sparsity issue on tail-labels.

3.4 Performance with Different DA Algorithms

XDA is a framework that can work with different DA algorithms. Here, to check how XDA performs with different DA algorithms, we consider four frequently-used DA algorithms: rule-based EDA (Wei and Zou, 2019), model-based BackTranslation (Xie et al., 2020), CBERT (Wu et al., 2019) and T5-MLM (Raffel et al., 2020).

As shown in Table 4, Under the XDA framework, DA methods based on pre-trained language models have achieved performance improvements compared to the Baseline (Vanilla LightXML). Particularly, there is a significant performance enhancement in predicting tail labels. This indicates the effectiveness of the proposed XDA framework. The XDA framework uses the *filter* to select high-quality augmented samples with diversity and consistency. Additionally, through the *mask*, it alleviates the long-tail problem in the dataset, leading to a significant improvement in model performance.

XDA brings the largest improvement compared with other DA methods. These results convincingly highlight the superior performance of the XDA algorithm, especially when it comes to tail-labels.

3.5 Ablation Study

We analyze the impacts of key components in XDA via ablation test. The complete XDA is compared with the following variants: XDA (w/o *Prompt*), XDA removes prompt-based learning and employs vanilla T5-MLM for DA; XDA (w/o *Filter*), XDA removes sample filtering; XDA (w/o *Mask*), XDA removes the masked data augmentation (MDA) strategy and directly utilizes augmented data for training. Table 5 shows the results evaluated on Eurlex-4K and Wiki10-31K datasets in terms of

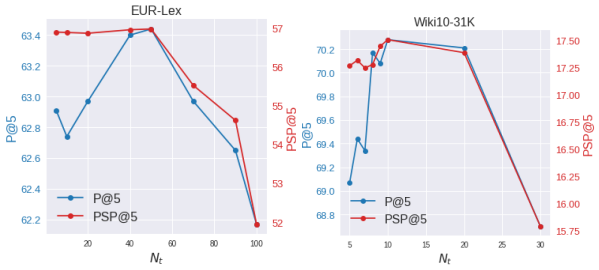

Figure 3: Effect of the Head-to-tail threshold N_t.

P@k and PSP@k. There are several interesting observations:

Mask training is crucial for the XDA framework. As shown in the table, the performance of XDA (w/o *Mask*) is even worse than the baseline, indicating the significant role of MDA. This is because, if DA is performed directly without considering the issue of imbalance in the XMC task, augmentation not only fails to alleviate the problem but also exacerbates the long-tail issue. This leads to overfitting on the head-label samples, resulting in a significant decrease in model performance.

Prompt-based learning also notably enhances the performance of the XDA framework. Soft prompt assists XDA in producing domain-specific synthetic data with a certain level of diversity.

Filtering also plays a crucial role in XDA's performance. Without removing low-quality synthetic data, there is a decrease in the performance gap.

3.6 Parameter Sensitivity

There are two major hyper-parameters we proposed in XDA, including head-to-tail threshold N_t and times of augmentation N_a that controlling the intensity of the augmentation. The impacts of the N_t are shown in Figure 3. The effect of N_a can be found in Appendix A.5.

The performance initially improves but then declines as the value of N_t increases. N_t determines

the proportion of head and tail labels. A smaller N_t corresponds to a larger number of head labels. Consequently, many head-label classifiers are masked in the MDA phase, causing the model to focus more on learning the tail-label classifiers. This significantly reduces the classification performance of the head-labels, resulting in a deterioration of the overall model performance. Interestingly, due to the model's increased emphasis on learning the tail labels, the decline in the evaluation metric PSP@5, which is more indicative of long-tail labels, is not significant. Conversely, when N_t is large, the size of head labels diminishes, indicating that only a few head-label classifiers are masked during MDA. As a result, the XDA framework fails to effectively address the long-tail problem, leading to a decrease in performance. In conclusion, we observe a trade-off between the head-to-tail threshold N_t and model performance.

4 Related Work

Extreme Multi-Label Text Classification Conventional methods for dealing with extreme multi-label text classification (XMC) is to utilize the fixed input representations such as sparse TF-IDF features to induce classification models (Babbar and Schölkopf, 2017; Wydmuch et al., 2018; Khandagale et al., 2020) and study different partitioning techniques to reduce complexity. For instance, sparse linear one-versus-all (OVA) methods such as DiSMEC (Babbar and Schölkopf, 2017) explore parallelism to solve OVA losses and reduce the model size by weight truncations. Tree-based methods (Wydmuch et al., 2018; Khandagale et al., 2020; Yu et al., 2022), such as Parabel (Prabhu et al., 2018), partition labels with a hierarchical label trees (HLT), leading to inference time complexity that is logarithmic in the output space. Neural-based XMC models (Liu et al., 2017; You et al., 2019; Dahiya et al., 2021; Mittal et al., 2021b; Saini et al., 2021; Mittal et al., 2021a) employ various network architectures to learn semantic embeddings of the input text. Recently, pre-trained Transformer models have been applied to XMC problems with promising results (Chang et al., 2020; Ye et al., 2020; Jiang et al., 2021; Zhang et al., 2021a; Kharbanda et al., 2022). LightXML (Jiang et al., 2021) fine-tunes Transformer encoders with the OVA loss function end-to-end via dynamic negative sampling from the matching network trained on label cluster signals.

Even though previous techniques have achieved encouraging performance in XMC , it is still a challenging task due to the long-tailed label distribution (Chang et al., 2020; Zhang et al., 2023). DEPL (Zhang et al., 2023) focuses the challenge of tail label prediction by leveraging the dense neural retrieval model to generate label descriptions from relevant input documents. We takes an explicit data augmentation approach, directly mitigating the long-tailed problem in XMC tasks.

Data Augmentation Data augmentation (DA) has shown its effectiveness in many low-resource data scenarios(Kumar et al., 2020; Zhang et al., 2020; Zhou et al., 2022; Wang et al., 2022). The most commonly used DA method is the word substitution-based method, such as EDA (Wei and Zou, 2019). Later, Pre-trained Language Models (PLMs) have also been employed for DA . For instance, Back Translation (Xie et al., 2020) utilizes machine translation models to synthesize new data samples. CBERT (Wu et al., 2019) masks some tokens and predicts their contextual substitutions with the pretrained BERT model. PromDA (Wang et al., 2022) leverages soft prompting to facilitate efficient learning from few-shots, building upon the T5 model (Raffel et al., 2020).

Due to the label co-occurrence, it is challenging for these prior methods to handle XMC (Xiao et al., 2021; Zhang et al., 2022). Instead of previous *sample-level* augmentation, XDA is a *pair-level* augmentation approach, which focuses on augmenting positive feature-label pairs for the tail-labels. This strategy effectively alleviates the severe long-tail problem in XMC tasks.

5 Conclusions and Future Work

In this study, we propose XDA, a prompt-based data augmentation framework designed specifically for extreme multi-label text classification (XMC). By employing *pair-level* data augmentation on the tail-labels, XDA aims to ameliorate the long-tail problem in XMC tasks. Experiments on three benchmarks show the effectiveness of our proposed XDA method, especially on tail-labels. In the future, we would like to explore how to leverage XDA in the scenarios with larger label space. In addition, we are also interested in boosting more advanced large language models for XMC.

Limitations

As an initial exploration of prompt-based data augmentation methods in the field of extreme multi-label text classification (XMC) , we have not yet considered the scenarios involving an extremely large number of labels. This is primarily because, in such cases, a hierarchical label tree (HLT) needs to be constructed to facilitate the XMC task. However, at present, there is a lack of well-established strategies for effectively combining the HLT approach with data augmentation methods. Additionally, we have not explored the use of more advanced large language models within the XDA framework, such as GPT4 and LLaMA(Touvron et al., 2023).

References

Ateret Anaby-Tavor, Boaz Carmeli, Esther Goldbraich, Amir Kantor, George Kour, Segev Shlomov, Naama Tepper, and Naama Zwerdling. 2020. Do not have enough data? deep learning to the rescue! In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7383–7390. AAAI Press.

Rohit Babbar and Bernhard Schölkopf. 2017. Dismec: Distributed sparse machines for extreme multi-label classification. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM 2017, Cambridge, United Kingdom, February 6-10, 2017*, pages 721–729. ACM.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

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.

Wei-Cheng Chang, Daniel L. Jiang, Hsiang-Fu Yu, Choon-Hui Teo, Jiong Zhang, Kai Zhong, Kedarnath Kolluri, Qie Hu, Nikhil Shandilya, Vyacheslav Ievgrafov, Japinder Singh, and Inderjit S. Dhillon. 2021. Extreme multi-label learning for semantic matching in product search. In *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, pages 2643–2651. ACM.

Wei-Cheng Chang, Hsiang-Fu Yu, Kai Zhong, Yiming Yang, and Inderjit S. Dhillon. 2020. Taming pretrained transformers for extreme multi-label text classification. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 3163–3171. ACM.

Kunal Dahiya, Deepak Saini, Anshul Mittal, Ankush Shaw, Kushal Dave, Akshay Soni, Himanshu Jain, Sumeet Agarwal, and Manik Varma. 2021. DeepXML: A deep extreme multi-label learning framework applied to short text documents. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 31–39.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Nilesh Gupta, Sakina Bohra, Yashoteja Prabhu, Saurabh Purohit, and Manik Varma. 2021. Generalized zero-shot extreme multi-label learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 527–535.

Himanshu Jain, Yashoteja Prabhu, and Manik Varma. 2016. Extreme multi-label loss functions for recommendation, tagging, ranking & other missing label applications. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 935–944. ACM.

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 *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 7987–7994. AAAI Press.

Ehsan Kamalloo, Mehdi Rezagholizadeh, and Ali Ghodsi. 2022a. When chosen wisely, more data is what you need: A universal sample-efficient strategy for

data augmentation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1048–1062, Dublin, Ireland. Association for Computational Linguistics.

Ehsan Kamalloo, Mehdi Rezagholizadeh, and Ali Ghodsi. 2022b. When chosen wisely, more data is what you need: A universal sample-efficient strategy for data augmentation. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1048–1062. Association for Computational Linguistics.

Sujay Khandagale, Han Xiao, and Rohit Babbar. 2020. Bonsai: diverse and shallow trees for extreme multi-label classification. *Mach. Learn.*, 109(11):2099–2119.

Siddhant Kharbanda, Atmadeep Banerjee, Erik Schultheis, and Rohit Babbar. 2022. Cascadexml: Rethinking transformers for end-to-end multi-resolution training in extreme multi-label classification. In *NeurIPS*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Varun Kumar, Ashutosh Choudhary, and Eunah Cho. 2020. Data augmentation using pre-trained transformer models. In *Proceedings of the 2nd Workshop on Life-long Learning for Spoken Language Systems*, pages 18–26, Suzhou, China. Association for Computational Linguistics.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jingzhou Liu, Wei-Cheng Chang, Yuexin Wu, and Yiming Yang. 2017. Deep learning for extreme multi-label text classification. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, pages 115–124. ACM.

Julian J. McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Seventh ACM Conference on Recommender Systems, RecSys '13, Hong Kong, China, October 12-16, 2013*, pages 165–172. ACM.

Eneldo Loza Mencía and Johannes Fürnkranz. 2008. Efficient pairwise multilabel classification for large-scale problems in the legal domain. In *Machine Learning and Knowledge Discovery in Databases, European Conference, ECML/PKDD 2008, Antwerp, Belgium, September 15-19, 2008, Proceedings, Part II*, volume 5212 of *Lecture Notes in Computer Science*, pages 50–65. Springer.

A. Mittal, N. Sachdeva, S. Agrawal, S. Agarwal, P. Kar, and M. Varma. 2021a. ECLARE: Extreme classification with label graph correlations. In *Proceedings of The ACM International World Wide Web Conference*.

Anshul Mittal, Kunal Dahiya, Sheshansh Agrawal, Deepak Saini, Sumeet Agarwal, Purushottam Kar, and Manik Varma. 2021b. DECAF: Deep extreme classification with label features. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 49–57.

Yashoteja Prabhu, Anil Kag, Shrutendra Harsola, Rahul Agrawal, and Manik Varma. 2018. Parabel: Partitioned label trees for extreme classification with application to dynamic search advertising. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*, pages 993–1002. ACM.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

D. Saini, A.K. Jain, K. Dave, J. Jiao, A. Singh, R. Zhang, and M. Varma. 2021. GalaXC: Graph neural networks with labelwise attention for extreme classification. In *Proceedings of The Web Conference*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Yufei Wang, Can Xu, Qingfeng Sun, Huang Hu, Chongyang Tao, Xiubo Geng, and Daxin Jiang. 2022. PromDA: Prompt-based data augmentation for low-resource NLU tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4242–4255, Dublin, Ireland. Association for Computational Linguistics.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. 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 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers:

State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Tong Wu, Qingqiu Huang, Ziwei Liu, Yu Wang, and Dahua Lin. 2020. Distribution-balanced loss for multi-label classification in long-tailed datasets. In *European Conference on Computer Vision*, pages 162–178. Springer.

Xing Wu, Chaochen Gao, Meng Lin, Liangjun Zang, and Songlin Hu. 2022. Text smoothing: Enhance various data augmentation methods on text classification tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 871–875, Dublin, Ireland. Association for Computational Linguistics.

Xing Wu, Shangwen Lv, Liangjun Zang, Jizhong Han, and Songlin Hu. 2019. Conditional BERT contextual augmentation. In *Computational Science - ICCS 2019 - 19th International Conference, Faro, Portugal, June 12-14, 2019, Proceedings, Part IV*, volume 11539 of *Lecture Notes in Computer Science*, pages 84–95. Springer.

Marek Wydmuch, Kalina Jasinska, Mikhail Kuznetsov, Róbert Busa-Fekete, and Krzysztof Dembczynski. 2018. A no-regret generalization of hierarchical softmax to extreme multi-label classification. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 6358–6368.

Lin Xiao, Xiangliang Zhang, Liping Jing, Chi Huang, and Mingyang Song. 2021. Does head label help for long-tailed multi-label text classification. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 14103–14111. AAAI Press.

Qizhe Xie, Zihang Dai, Eduard H. Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Nishant Yadav, Rajat Sen, Daniel N Hill, Arya Mazumdar, and Inderjit S Dhillon. 2021. Session-aware query auto-completion using extreme multi-label ranking. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 3835–3844.

Hui Ye, Zhiyu Chen, Da-Han Wang, and Brian D. Davison. 2020. Pretrained generalized autoregressive model with adaptive probabilistic label clusters for extreme multi-label text classification. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 10809–10819. PMLR.

Ronghui You, Zihan Zhang, Ziye Wang, Suyang Dai, Hiroshi Mamitsuka, and Shanfeng Zhu. 2019. Attentionxml: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5812–5822.

Hsiang-Fu Yu, Kai Zhong, Jiong Zhang, Wei-Cheng Chang, and Inderjit S. Dhillon. 2022. PECOS: prediction for enormous and correlated output spaces. *J. Mach. Learn. Res.*, 23:98:1–98:32.

Danqing Zhang, Tao Li, Haiyang Zhang, and Bing Yin. 2020. On data augmentation for extreme multi-label classification. *ArXiv preprint*, abs/2009.10778.

Jiaxin Zhang, Jie Liu, Shaowei Chen, Shaoxin Lin, Bingquan Wang, and Shanpeng Wang. 2022. Adam: An attentional data augmentation method for extreme multi-label text classification. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 131–142. Springer.

Jiong Zhang, Wei-Cheng Chang, Hsiang-Fu Yu, and Inderjit S. Dhillon. 2021a. Fast multi-resolution transformer fine-tuning for extreme multi-label text classification. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 7267–7280.

Jiong Zhang, Wei-Cheng Chang, Hsiang-Fu Yu, and Inderjit S. Dhillon. 2021b. Fast multi-resolution transformer fine-tuning for extreme multi-label text classification. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 7267–7280.

Ruohong Zhang, Yau-Shian Wang, Yiming Yang, Donghan Yu, Tom Vu, and Likun Lei. 2023. Long-tailed extreme multi-label text classification by the retrieval of generated pseudo label descriptions. In *Findings of the Association for Computational Linguistics: EACL 2023, Dubrovnik, Croatia, May 2-6, 2023*, pages 1062–1076. Association for Computational Linguistics.

Minyi Zhao, Lu Zhang, Yi Xu, Jiandong Ding, Jihong Guan, and Shuigeng Zhou. 2022a. EPiDA: An easy plug-in data augmentation framework for high performance text classification. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4742–4752, Seattle, United States. Association for Computational Linguistics.

Minyi Zhao, Lu Zhang, Yi Xu, Jiandong Ding, Jihong Guan, and Shuigeng Zhou. 2022b. EPiDA: An easy plug-in data augmentation framework for high performance text classification. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4742–4752, Seattle, United States. Association for Computational Linguistics.

Jing Zhou, Yanan Zheng, Jie Tang, Li Jian, and Zhilin Yang. 2022. FlipDA: Effective and robust data augmentation for few-shot learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8646–8665, Dublin, Ireland. Association for Computational Linguistics.

Arkaitz Zubiaga. 2012. Enhancing navigation on wikipedia with social tags. *CoRR*, abs/1202.5469.

A Appendix

A.1 Datasets

We evaluate the proposed model on three benchmark datasets for XMC, which are Eurlex-4K, Wiki10-31K and AmazonCat-13K.

- Eurlex-4K (Mencía and Fürnkranz, 2008) is a dataset consisting of documents related to European Union law, covering a wide range of 3,956 subjects. The dataset's public version includes 15,539 instances for training and 3,809 instances for testing purposes.

- Wiki10-31K (Zubiaga, 2012) is a dataset with 20,762 tagged Wikipedia articles.

- AmazonCat-13K (McAuley and Leskovec, 2013) is a superset of existing publicly-available Amazon datasets. It contains more than 13,000 category labels and more than 1,000,000 product description texts.

A.2 Baselines

Baselines of XMC We compare our proposed XDA method to the most representative and state-of-the-art (SOTA) XMC methods:

- DisMEC (Babbar and Schölkopf, 2017): a distributed framework which employs a double layer of parallelization and explicit capacity control.

- PfastreXML (Jain et al., 2016): an algorithm suited to predicting tail labels which uses classifiers designed specifically for tail labels.

- Parabel (Prabhu et al., 2018): a probabilistic model which learns a balanced label hierarchy and generalizes hierarchical softmax to multi-label setting.

- Bonsai (Khandagale et al., 2020): a suite of algorithms which extends label representation by partitioning labels in the input,output and joint space.

- AttentionXML (You et al., 2019): a label tree-based model which uses multi-label attention and a probabilistic label tree to handle large label sets.

- X-Transformer (Chang et al., 2020): a scalable approach for fine-tuning deep transformer models in extreme multi-label classification.

- XLNet-APLC (Ye et al., 2020): a Transformer-based model which fine-tunes XLNet and exploits the unbalanced label distribution.

- LightXML (Jiang et al., 2021): a deep learning method which uses generative cooperative networks to recall and rank labels.

- DEPL (Zhang et al., 2023): a retrieval-based model which leverags dense neural retrieval model and generats pseudo label descriptions.

Baselines of DA In this work, we consider four data augmentation methods as our baselines.

- EDA (Wei and Zou, 2019) is a straightforward augmentation technique that replaces words in a text, and it has demonstrated its effectiveness in improving text classification performance when dealing with limited data.

- Back Translation (Xie et al., 2020) is a technique that involves translating a sentence into a temporary language (EN-DE) and subsequently translating the previously translated text back into the original source language (DE-EN). This process of translation back and forth aids in augmenting the dataset.

- CBERT (Wu et al., 2019) is an approach that involves masking certain tokens in a text and then predicting their contextual substitutions using a pre-trained BERT model. This method leverages the power of BERT for generating augmented data.

- T5-MLM (Raffel et al., 2020) introduces synonym phrase replacements in specific text regions to increase the diversity of augmented samples within the dataset based on a pre-trained T5 model.

A.3 Evaluation Metrics

In this section, we define the evaluation metrics used in this paper. Following previous works (You et al., 2019; Zhang et al., 2021b; Kharbanda et al., 2022; Zhang et al., 2023), we use two main metrics which are commonly used in XMC evaluations: the precision at k ($P@k$) and propensity scored precision at k ($PSP@k$).

The precision of the top-k labels is defined as:

$$P@k = \frac{1}{k} \sum_{l=1}^{k} \boldsymbol{y}_{rank(l)} \tag{11}$$

where $\boldsymbol{y} \in \{0,1\}^L$ is the ground truth label vector, and $rank(l)$ is the index of the l-th highest predicted label.

Since $P@k$ gives an equal weight to the per-instance scores, the resulted average is dominated by the system's performance on the head-labels but not the tail-labels. In other words, the performance comparison in $P@k$ cannot provide enough insights to the effectiveness of methods in tail label prediction. As an alternative metric, we use $PSP@k$ (Jain et al., 2016) as our evaluation metric for performance comparison on tail-labels. $PSP@k$ re-weights the precision on each instance as:

$$PSP@k = \frac{1}{k} \sum_{l=1}^{k} \frac{\boldsymbol{y}_{rank(l)}}{\boldsymbol{p}_{rank(l)}} \tag{12}$$

where $\boldsymbol{p}_{rank(l)}$ is the propensity score (Jain et al., 2016) of label $rank(l)$, and it gives higher weights to tail-labels. The metric involves application specific parameters A and B. For consistency, we use the same setting as AttentionXML (You et al., 2019) for all datasets.

A.4 Implementation Details

We adopt the pre-trained BERT (Devlin et al., 2019) as the backbone of our XMC model, using the Pytorch implementation from HuggingFace Transformers (Wolf et al., 2019). The maximum document length is 512 due to BERT limitations (Devlin et al., 2019), and documents are zero-padded or truncated to this length. All experiments are carried out in a Linux environment with a single Tesla V100 GPU (32G). The training took about 2.4 hours and 5.3 hours for Eurlex-4K and Wiki10-31K datasets respectively. For the AmazonCat-13K dataset, it required about 109 hours.

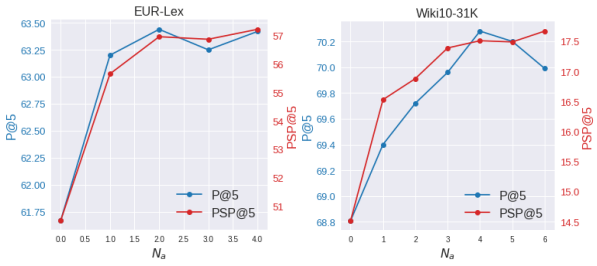

Figure 4: Effect of the Times of augmentation N_a

DA is built on the top of the T5-Large model (Raffel et al., 2020). DA requires Prompt fine-tuning with down-stream tasks. We use Adafactor optimizer with learning rate 1e-3 and weight decay 1e-5 to train the soft prompt parameters. When fine-tuning on the XMC data, we set the batch size 32. To better initialize the soft prompt parameters for the data augmentation tasks, we utilized the checkpoint obtained from the PromDA (Wang et al., 2022).

A.5 Parameter Sensitivity

Increasing N_a from 0 to 4 can greatly help XDA to gain strong improvement on PSP@5. That's to say, augmenting more samples for tail-labels can effectively strengthen the generalizability of the classifiers. However, when N_a becomes too large, it means that there are too many augmented samples for tail-labels, resulting in the model focusing excessively on the tail labels, which can cause a decline in overall performance P@5.

A.6 Case Study

Table 6 shows representative examples generated by our proposed XDA and other DA methods. The rule-based EDA approach rearranges the original word order and introduces some synonymous word replacements, but this process disrupts the coherence of the original text. The model-based BackTrans. generates a paragraph that preserves semantic similarity to the original text and exhibits higher consistency. However, it lacks diversity. With T5-MLM, certain text regions undergo synonym phrase replacements, resulting in enhanced diversity of the augmented samples with the dataset. Since XDA incorporates prompt-based learning on

Labels:	amazon aws cloud-computing computing services web webmaster webservice
Original:	the amazon web services aws are a collection of remote computing services also called web services offered over the internet by amazon.com launched in july 2002 amazon web services provide online services for other web sites or client-side applications most of these services are not exposed directly to end users but instead offer functionality that other developers can use in june 2007 amazon claimed that more than 330,000 developers had signed up to use amazon web services.
EDA:	the amazon web services aws a collection of services also called web services offered over the internet amazon com launched in july services provide online services for web sites or side most of these services are exposed directly to end users but offer functionality that other can use june amazon claimed that more than developers had signed up to use amazon.
BackTrans.:	Amazon web services aws are a collection of remote computing services, also known as web services, offered over the Internet by amazon.com, which was founded in July 2002. Amazon web services provide online services for other websites or client-side applications. Most of these services are not directly exposed to end users, but instead provide features that other developers can use. Amazon claimed in June 2007 that more than 330,000 developers had signed up to use Amazon web services.
T5-MLM:	the amazon web services . amazon has also launched its own collection of remote web services offered on the internet by amazon.com . In 2002 amazon web services launched online services for remote web sites , most of these services are not available to developers but instead offer functionality that you can use. In june 2007 amazon announced that more than 330,000 developers have signed up for the amazon web services.
XDA:	the Amazon Web Services (AWS) are a set of remote computing services also called web services offered over the internet Amazon Web Services (AWS) are an online service provided by Amazon that was launched in July 2002. AWS is the acronym for Amazon Web Service and it stands as the name of its main product. Amazon web services provide online services that are used to build other web sites or client-side applications most of these services do not provide functionality directly to end users but instead offer functionality that other developers can use in their own applications In 2007 amazon claimed that more than 1 million developers had signed up to use Amazon web services.

Table 6: Generated synthetic data from our proposed DA and other baseline methods.

the current corpus, the generated samples maintain a high level of consistency. Additionally, guided by labels, XDA can extract the embedded real-world knowledge from the PLMs and introduces these knowledge into a relatively long paragraph in a fluent way.