Open-world Multi-label Text Classification with Extremely Weak Supervision

Anonymous ACL submission

Abstract

This work focuses on a new challenging problem, open-world multi-label text classification under extremely weak supervision, where only raw documents are provided without any labels or ground-truth label space. The multi-label nature makes the existing (hard-)clustering-based methods ineffective. We observe that (1) most documents have a dominant class covering the majority of content and (2) long-tail labels would appear in some documents as dominant class. Following these observations, we propose a novel method, X-MLClass, to discover a comprehensive label space and construct a multi-label classifier. Specifically, we start with a reasonable subset of all the documents and prompt a large language model (LLM) for their most dominant keyphrases to obtain an initial set of labels. We then leverage a zero-shot multi-label classifier, identifying the documents with lower predicted scores and revisiting the keyphrases in those documents for more long-tail labels. Later, we include these long-tail labels into the label set and reiterate this process. Extensive experiments demonstrate that X-MLClass exhibits a remarkable 40% increase on the AAPD dataset in ground-truth label space coverage compared to traditional topic modeling methods. Additionally, it achieves higher accuracy in zero-shot multi-label text classification.

1 Introduction

Multi-label text classification (MLTC) aims to assign one or more labels to each input document in the corpus. While the traditional methods work in a fully supervised setting, recent works start to pay more attention to weakly supervised settings using limited labeled data (Liu et al., 2022) or even in the absence of any labeled data (Shen et al., 2021; Xiong et al., 2021). The state-of-the-art zero-shot (single-label) text classification methods (Pàmies et al., 2023; Gera et al., 2022) follow the textual entailment framework by comparing the document and the label in a pairwise manner. However, all these methods still require a complete list of class names, which might be challenging even for domain experts to provide beforehand given the massive number of documents.

This work focuses on a new challenging problem, open-world multi-label text classification under extremely weak supervision, where only raw documents are provided without any labels or ground-truth label space. The most related problems are text clustering (Zhang et al., 2023; Wang et al., 2023b) and topic modeling (Grootendorst, 2022; Pham et al., 2023), where those methods are typically only capable of assigning a single label to each document. The multi-label nature makes the existing (hard-)clustering-based methods ineffective.

We observe that (1) most documents have a dominant class covering the majority of content and (2) long-tail labels would appear as the dominant class in some documents. Our observations are confirmed by experiments based on 5 benchmark datasets: AAPD (Yang et al., 2018), Reuters-21578 (Debole and Sebastiani, 2005), RCV1-V2 (Lewis et al., 2004), DBPedia-298 (Lehmann et al., 2015), and Amazon-531 (McAuley and Leskovec, 2013). Specifically, we prompt a large language model (LLM) to check if any of the ground truth labels of a given document is dominant, i.e., covering more than 50% of the content; and if it exists, which one is the dominant label. [1] After checking two thousand randomly sampled documents, the LLM believes that more than 90% of documents contain a dominant class, and human spot-checking results agree with this too. Moreover, in every dataset, inspecting all the labels, the LLM believes that 100% of them are dominant classes of at least one document.

[1] The specific prompt can be found in Appendix A

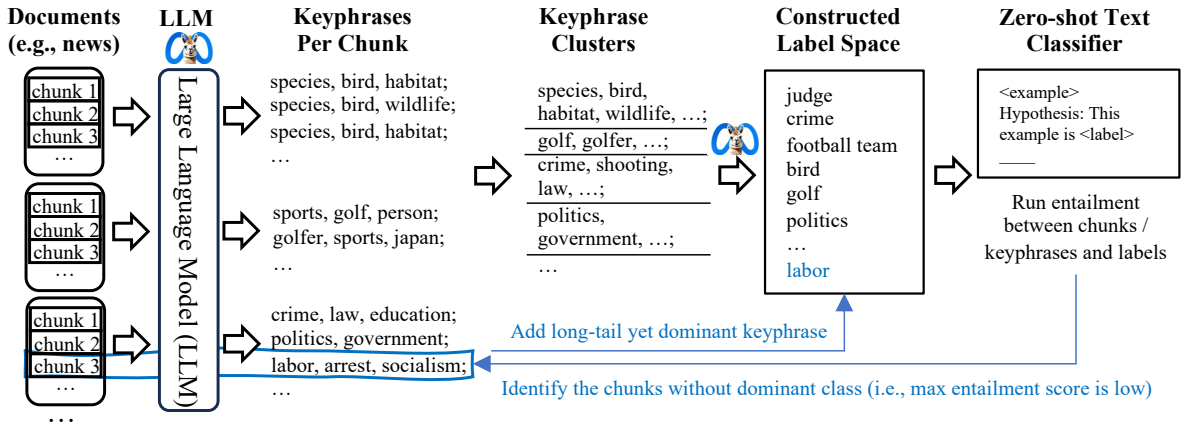

Figure 1: An overview of our X-MLClass framework.

Following these observations, we propose a novel method, X-MLClass, to discover a pragmatic label space and construct a multi-label text classification classifier with the assistance of a customer-sized LLM (i.e., `llama-2-13b-chat` in our experiments), as illustrated in Figure 1.

The first step in X-MLClass is to construct a high-quality label space. To balance the label coverage and the cost of LLM, we work on a reasonably large subset of all the documents. For each document, we partition it into chunks to better align with the context length of LLM while ensuring that each chunk contains a single topic, and then prompt the LLM to generate the most dominant keyphrases for each chunk. This process also anticipates a higher chance of having only one label per chunk. As previous LLM-based text clustering work has suggested (Wang et al., 2023b,a), there are very likely some semantically redundant yet lexically different keyphrases among the generated ones. We cluster these keyphrases, and within every cluster, we pull together the corresponding chunks of the keyphrases closest to the cluster center to prompt the LLM once again, generating one single label for each cluster. We further eliminate labels exhibiting extremely high similarity scores, and for those borderline similar label pairs, a little human effort becomes integral. Combining all these survived labels constitutes an initial label space.

We then apply the state-of-the-art textual entailment-based classification methods (Pàmies et al., 2023; Gera et al., 2022) to construct a classifier to re-access the documents and identify long-tail labels. Specifically, we query every text chunk against all the labels for the entailment score. We identify the chunks with small top predicted scores, indicating that they lack a dominant class. Therefore, we revisit the keyphrases generated by these chunks to unveil more long-tail labels. We selectively choose keyphrases that exhibit a modest presence within the entire keyphrase set, but are notably absent in the original label space. We include these new keyphrases in the label set and repeat re-accessing documents with this newly updated label set for a fixed number of iterations. A caveat is that to ensure wider coverage of the long-tail keyphrases, we hold back a portion of high-popularity labels in the label set each iteration. These high-popularity labels are included back after all iterations.

Extensive experiments on 5 benchmark datasets reveal the superiority of X-MLClass outperforming all compared methods. Remarkably, compared with baselines, X-MLClass achieves a significant enhancement of 40% and 25% in ground-truth label space coverage on the AAPD and RCV1-V2 datasets, respectively. Furthermore, it achieves higher accuracy in zero-shot multi-label text classification, surpassing the top-ranking models on HuggingFace across all datasets.

Our contributions are summarized as:

- We attack a new, challenging problem, open-world MLTC with extremely weak supervision, where only raw documents are available, without any labeled data or ground-truth label space.
- We propose a novel framework, X-MLClass, based on two intuitive, empirically confirmed observations. X-MLClass discovers the label space and builds an MLTC classifier with the assistance of LLM. The only required human effort is to resolve a few pairs of candidate labels with borderline similarity scores.

- X-MLClass achieves a significantly higher coverage score compared to traditional topic modeling methods, along with superior end-to-end classification accuracy.

Reproducibility. We will release the code upon acceptance.

2 Related Work

Topic Modeling: Topic modeling has been widely adopted for discovering latent thematic structures within collections of text documents. Traditional models, such as Latent Dirichlet Allocation (LDA) (Blei et al., 2003) and Non-Negative Matrix Factorization (NMF) (Févotte and Idier, 2011) represent documents as mixtures of latent topics using bag-of-words representations, but they often neglect the semantic relationships between words. Addressing this limitation, new techniques like Top2Vec (Angelov, 2020) and BERTopic (Grootendorst, 2022) build primarily on clustering embeddings, demonstrating the potential of embedding-based topic modeling approaches. Another recent method, TopicGPT (Pham et al., 2023), takes a different approach by prompting large language models for topic generation, aligning more closely with ground truth labels. However, these existing methods typically provide a single topic for each document, which poses challenges when extending them to multi-label scenarios.

Multi-label Text Classification: Numerous approaches have been proposed to tackle the complexities of Multi-Label Text Classification (MLTC) problems. Bhatia and Jain (Bhatia et al., 2015) employ embedding-based methods, leveraging the power of embeddings to train individual classifiers for each label. Later, there has been a notable surge in the application of Neural Network-based models to address MLTC tasks. For instance, XML-CNN (Liu et al., 2017) uses a Convolutional Neural Network (CNN) to learn text representations, demonstrating improvements in MLTC accuracy. It is important to note that all these methods rely on labeled data, restricting their applicability in scenarios where labeled information is unavailable. Recent works have started to tackle MLTC problems using a small amount of labeled data or even with no labels at all. For example, Shen et al. (2021) achieves impressive results by using only class names and taxonomies. Rios and Kavuluru (2018) train a neural architecture with both true labels and their natural language descriptor. However, these methods still require access to the ground-truth label space, at the very least.

Open-world Single-label Text Classification: In recent developments, there has been a surge in open-world models utilizing LLM prompts to derive labels without relying on ground-truth label spaces. Notably, GOALEX (Wang et al., 2023b) generates labels for text samples based on users' specific goals, demonstrating a goal-driven approach. Another noteworthy model, CLUSTER-LLM (Zhang et al., 2023), leverages API-based LLMs to guide text clustering, resulting in improved performance. The approach of intent discovery (Zhang et al., 2022), aiming to infer latent intents from a document set, has proven effective in generating label spaces. A newly introduced method, IDAS (De Raedt et al., 2023), prompts LLMs to succinctly summarize utterances, enhancing intent prediction. However, akin to topic modeling methods, all these approaches are currently limited to assigning only a single label to each document.

3 Problem Formulation

Given an unlabeled corpus $\mathcal{D} = \{D_1, D_2, \ldots, D_n\}$, where $D_i \in \mathcal{D}$ represents a document in the collection. Our task is to (1) identify class names $\mathcal{C} = \{C_j\}_{j=1}^{K}$, where K is the *unknown* number of classes, and (2) build a text classifier $f(\cdot)$ to map any raw document D_i to its target labels $Y_i = \{y_i^j\}_{j=1}^{p}$, where y_i^j is the single label name and p is the number of target labels for D_i.

To the best of our knowledge, this is the first work that explores open-world multi-label text classification without the presence of a ground-truth label space. This is a very challenging problem, so we assume that human experts are willing to devote some very limited effort, i.e., *extremely weak supervision*. For example, the human expert shall be able to annotate tens of label pairs and confirm whether they appear equivalent or not. We also assume that human experts possess insights into the magnitude of the label space based on dataset characteristics. For instance, news datasets typically contain a broader range of classes compared to datasets consisting of computer science paper abstracts.

4 Our X-MLClass Framework

X-MLClass consists of three key steps. First, every document is split into chunks and transformed into

keyphrases by prompting an LLM to construct an initial label space through clustering. We further assign labels to each raw document D_i using a custom keyphrase-chunk zero-shot textual entailment classifier. Finally, we iteratively enhance the label space by incorporating additional long-tail labels. The framework overview is depicted in Figure 1, and the below sections provide a detailed discussion of each step.

4.1 Initial Label Space Construction

The first step in X-MLClass is to construct a high-quality label space. To balance label coverage and the computational cost of LLM, X-MLClass is applied to a reasonably large subset of the corpus $\mathcal{D}$, denoted as $\mathcal{D}_{sub} \subset \mathcal{D}$.

Dominant Keyphrase Generation: For each document, we partition it into chunks to better align with the context length of LLM, and then prompt for the most dominant keyphrases per chunk. Specifically, each document $D_i \in \mathcal{D}_{sub}$ is segmented into chunks $\{S_i^1, S_i^2, \dots\}$, with a predefined chunk size of 50 tokens. This choice is also made to ensure each chunk primarily contains one label, allowing us to leverage state-of-the-art textual entailment-based single-label text classification methods for every chunk later. To generate keyphrases for each chunk S_i^j, we employ an LLM and provide it with an instruction such as "*find at most three labels for this document*". The LLM then refines keyphrases p_i^j from the chunk S_i^j, serving as potential class candidates for subsequent stages of our X-MLClass model. Keyphrases generated from each chunk collectively form a set $\mathcal{P}$.

Keyphrase Clustering: As previous LLM-based text clustering work has suggested (Wang et al., 2023b,a), there are very likely some semantically redundant yet lexically different keyphrases among the generated ones. Therefore it is necessary to cluster the keyphrases at the semantic level. Specifically, we employ the state-of-the-art instruction-tuned text embedding model, `instructor-large` (Su et al., 2022), to generate vector representations for all the keyphrases in $\mathcal{P}$. Traditional clustering methods face challenges in high-dimensional spaces (Aggarwal et al., 2001; Wang et al., 2020b), primarily attributed to variations in distance measurements. To address this limitation, we apply the dimensionality reduction method to trim down the embedding dimension. Particularly, we choose UMAP (McInnes et al., 2018) because it can effectively balance local and global structures, demonstrating improved performance in handling high-dimensional data. Finally, we obtain the clusters using the Gaussian Mixture Model (GMM) in the projected low-dimensional space, renowned for its enhanced flexibility in capturing intricate data distributions.

Number of Clusters: The number of clusters is determined by considering both the insights of human experts regarding the magnitude of the label space and non-parametric clustering methods such as BERTopic (Grootendorst, 2022), a highly effective topic modeling method for generating topics in a corpus. For example, one can train BERTopic on the keyphrase set $\mathcal{P}$ to obtain the topic number K^0, serving as the hyper-parameter to GMM.

Redundant Keyphrase Removal: Within every cluster, we focus on the three keyphrases closest to the cluster center to synthesize one single label. Instead of directly employing the keyphrases for label space creation, we trace back to the original chunks responsible for generating these keyphrases. Concatenating these three chunks for each cluster results in a new document S'. K^0 documents collectively form a new corpus $\mathcal{S}' = \{S_j'\}_{j=1}^{K^0}$. For each S_j', we prompt LLM with an instruction "*find one label for this document*", thereby yielding the initial K^0 classes $\{C_j\}_{j=1}^{K^0}$.

This initial label space may contain redundant labels and sometimes requires expert assistance for refinement. Sentence-Transformer models (Wang et al., 2020a) are used to identify distinct pairs of classes with cosine similarity ≥ 0.75. The first class in each identified pair is then removed, representing a straightforward approach to eliminate redundant labels. This method proves effective in creating a robust label space $\{C_j\}_{j=1}^{K^1}$, and while human involvement can enhance the refinement process especially for those borderline similar label pairs, it is not mandatory. Further details on human involvement in the label space refinement are provided in appendix B.

4.2 Textual Entailment-based Classifier

Given a label space, we build a zero-shot textual entailment-based classifier (Yin et al., 2019). As our chunks are short enough (i.e., only 50 tokens), there is typically only one label per chunk. Therefore, the state-of-the-art zero-shot single-label text classification methods (Pàmies et al., 2023; Gera

et al., 2022; He et al., 2021) are all applicable here. Specifically, we compare every text chunk against all the labels using a textual entailment model. For each chunk $s \in \mathcal{S}$ and each class name $c \in \mathcal{C}$, we derive $E_{s,c}$ representing the confidence for the chunk s entailing the hypothesis "*This example is constructed for c*". Similarly, we obtain $E_{p,c}$ for each keyphrase $p \in \mathcal{P}$ representing the confidence for the phrase p entailing the same hypothesis as above. Subsequently, for each example in $\mathcal{S}$, we identify the label c with the top entailment score, denoted by $E_{s,c} > E_{s,c'}, \forall c' \neq c$.

Finally, we find all s and p belong to the same document D_i and group them into a new set Q. For each instance in Q, we rank the label candidates according to their entailment scores. We identify the labels that occur most frequently with the same ranking as the predicted labels for document D_i, progressing from the top-ranking to the lowest-ranking order.

4.3 Label Space Improvement

We further identify the chunks with lower top predicted scores — these chunks lack a dominant class. We rank $E_{s,c}$ in ascending order and select a subset $\mathcal{S}_{sub} \subset \mathcal{S}$ with relatively lower entailment scores. Lower entailment scores suggest a potential association with a tail class, not included in our label space. Considering the possibility of multiple chunks belonging to the same document D_i, we refine $\mathcal{S}_{sub}$ by selecting s only if $E_{s'c} < 0.6$, where $\forall s' \in D_i$. For each $s \in \mathcal{S}_{sub}$, we examine all keyphrases in the corresponding p. If a keyphrase is absent in the label space but occurs more than 15 times in $\mathcal{P}$, we incorporate it into the label space $\mathcal{C}$.

Additionally, we compute the frequency of each label c with the top entailment score. Labels with lower frequency are removed from the label space $\mathcal{C}$. The high-frequency labels, secured as a part of the label space, are temporarily excluded from the later label space improvement process. By iteratively training the classifier based on the updated label space, the label set gets finalized by adding more long-tail labels. In the concluding stages, all high-frequency labels are reintroduced, culminating in the formation of ultimate label space.

5 Experiments

We assess the performance of X-MLClass through two primary criteria: label space quality and zero-shot MLTC accuracy. Our evaluation involves a comparison of our model's label coverage with that of four topic modeling methods. In terms of end-to-end classification accuracy, we test our method with several top-ranking models available on HuggingFace. The subsequent section provides comprehensive details on the datasets, baseline methods, evaluation metrics, implementation specifics, and performance analysis.

Table 1: Dataset statistics.

Dataset	# Train	# Text	# Class
AAPD	53,840	2,000	54
Reuters	7,769	3,019	90
RCV1-V2	643,531	160,883	103
DBPedia	196,665	49,167	298
Amazon	29,487	19,685	531

5.1 Datasets

We perform experiments on five benchmark datasets for multi-label text classification across various domains: **AAPD**, **Reuters**-21578, **RCV1-V2**, **DBPedia**-298, and **Amazon**-531. Detailed information about each dataset is provided in Appendix C. Table 1 shows that the number of labels in these datasets varies from tens to hundreds. All the methods will be applied on the documents from the training set, and then evaluated on the test set.

5.2 Compared Methods

We compare our X-MLClass framework with two types of methods.

Topic Modeling: We select four representative topic modeling methods with distinct paradigms. These methods include Latent Dirichlet Allocation (**LDA**) (Blei et al., 2003), Non-Negative Matrix Factorization (**NMF**) (Févotte and Idier, 2011), **Topic2Vec** (Angelov, 2020), and **BERTopic** (Grootendorst, 2022). LDA and NMF serve as foundational algorithms, extracting topics based on word frequency within documents. NMF decomposes the TF-IDF matrix to obtain latent topics. Topic2Vec and BERTopic represent more recent advancements with a focus on semantics. Topic2Vec extends the Word2Vec model to embed topics, facilitating the exploration of semantic relationships between documents. Meanwhile, BERTopic leverages BERT embeddings and the HDBSCAN clustering algorithm to identify topics.

Zero-shot Text Classification: State-of-the-art zero-shot text classifiers typically follow textual entailment (Yin et al., 2019; Pàmies et al., 2023). Therefore, we choose three entailment models: (1)

Table 2: Label Space Coverage Comparison. Top2Vec and BERTopic generate topics with multiple keywords. The predicted label is determined by selecting the top-ranking keyword based on each model's setting.

Model	AAPD	Reuters	RCV1-V2	DBPedia	Amazon
LDA	29.17	14.44	10.67	21.81	15.44
NMF	22.92	15.56	9.71	30.20	15.82
Top2Vec	33.33	17.77	21.35	31.87	16.38
BERTopic	25.00	20.00	7.76	33.89	18.08
X-MLClass	**67.35**	**25.55**	**46.60**	**53.02**	**21.66**

bart-large-mnli exclusively trained on the MNLI dataset, (2) **deberta-v3-large-all** trained on 33 datasets reformatted into the universal NLI format, and (3) **xlm-roberta-large-xnli** fine-tuned on the XNLI dataset. We apply these models using the HuggingFace Transformer pipeline, with a hypothesis template "*This example is {label}*".

5.3 Evaluation Metrics

Label Space Quality: We employ an automatic evaluation metric, **coverage**, to quantify the alignment between the ground-truth (GT) label space and the predicted label space. A ground-truth label is deemed "covered" if it can be matched to a predicted label with a similarity score surpassing a predefined threshold. For this study, we compute the similarity scores using the `all-MiniLM-L6-v2` model from HuggingFace Sentence-Transformers. In our evaluation, we set this threshold to 0.6 as it aligns the best with GPT4 and human evaluation on whether the predicted label has the same meaning as the ground-truth label.

The coverage score is computed as follows:

$$\text{Coverage} = \frac{1}{N}\mathbb{G}\left(\mathbb{I}\left(C^{\text{pred}}, C^{\text{GT}}\right)\right)$$

Here N is the total number of topics in the GT label set, $C^{\text{GT}}(C^{\text{pred}})$ denotes the set of ground-truth (predicted) labels. $\mathbb{I}$ is an indicator that returns 1 if the similarity score between the GT labels and predicted labels for the documents exceeds the threshold (0.6). $\mathbb{G}$ represents the bipartite graph maximum match algorithm.

Classification Accuracy: Because of the large label space, multi-label text classification typically employs the rank-based evaluation metric precision at k, i.e., P@k. It captures the percentage of true labels among top-k score labels and is used for performance comparison. P@k can be defined as:

$$\text{P@k} = \frac{1}{N}\sum_{i=1}^{N}\frac{C_i^{r_k} \cap L_i}{\min(k, |L_i|)}$$

where L_i and C_i denote the true labels and predicted labels for document D_i, $|L_i|$ is the number of true labels for D_i, and r_k is the k-th highest predicted label. We follow the same similarity threshold in the coverage score to decide if the true label and the predicted label are the same.

5.4 Implementation Details

We implement X-MLClass using the `llama-2-13b-chat` LLM. The chunk size is uniformly set to 50 across all datasets, ensuring a consistent approach. To maintain precision, we introduce a human-in-the-loop element. Specifically, we request human input to determine the word count constituting a single label. To optimize efficiency, a small subset is selected, aiming for a task completion time of less than 10 minutes per human contributor.

In configuring LDA and NMF, we align the number of topics with our approach. For Top2Vec and BERTopic, which employ HDBSCAN as a clustering method, specifying an exact number of topics is not feasible. However, to maintain consistency, we ensure that these methods generate clusters neither exceeding nor falling below 10 in comparison to our label number.

As human experts believe the label space magnitude of the AAPD, Reuters-21578, and RCV1-V2 datasets should be no more than 100, on these datasets, we strategically choose a subset of 3,000 documents as $\mathcal{D}_{sub}$ And it is intuitive to have more labels in Wikipedia because of its diverse article categories, therefore, 8,000 documents are sampled for DBPedia-298 as $\mathcal{D}_{sub}$. Amazon products anticipate an even wider range, so 14,000 documents are chosen as $\mathcal{D}_{sub}$.

In the label space improvement phase, by ranking the top entailment scores in ascending order, we select a subset of chunks S with comparatively lower entailment scores. To ensure consistency with the original document subset size chosen for chunk-keyphrase generation, we control the size of S proportionally. Precisely, for AAPD, Reuters-21578, and RCV1-V2 datasets, we select the top 500 examples. For DBPedia-298, the subset size is set at 1,000, and for Amazon-531, we choose 1,500 examples. When incorporating labels into the new label space, only those with a semantic similarity score lower than 55% compared to all existing labels are added. This methodology ensures a refined and relevant augmentation of the label space.

Table 3: Zero-Shot Multi-Label Text Classification Accuracy Comparison: baseline model trained on raw documents vs. our model trained on the combination of chunks and keyphrases.

Method		AAPD		Reuter		RCV1-V2		DBPedia		Amazon	
		P@1	P@3	P@1	P@3	P@1	P@3	P@1	P@3	P@1	P@3
bart-large-mnli	*w/* raw docs	0.1390	0.1497	0.0940	0.2547	0.3730	0.3367	0.6330	0.3713	0.5100	0.3827
	w/ X-MLClass	0.2743	0.2115	0.1450	0.3490	0.4530	0.3808	0.6890	0.3917	0.5620	0.4168
deberta-v3-large-all	*w/* raw docs	0.3240	0.2595	0.1290	0.3937	0.4550	0.3793	0.6410	0.3713	0.5810	0.4148
	w/ X-MLClass	0.3544	0.2733	0.0980	0.4102	0.4900	0.3883	0.6370	0.3953	0.5800	0.4170
xlm-roberta-large-xnli	*w/* raw docs	0.1330	0.1455	0.1260	0.3478	0.3270	0.3053	0.6670	0.3497	0.4760	0.3663
	w/ X-MLClass	0.2222	0.1930	0.1170	0.3837	0.4040	0.3383	0.6610	0.3767	0.5250	0.4072

5.5 Label Space Coverage Results

We present the coverage of the predicted label space in comparison to topic modeling baselines, as detailed in Table 2. Our method consistently outperforms all baseline approaches. Specifically, for the AAPD, RCV1-V2, and DBPedia-298 datasets, we achieve approximately 50% coverage of the ground-truth label space, showcasing a noteworthy increase of more than 20% compared to traditional topic modeling methods. This notable improvement can be attributed to the fact that labels generated by topic model methods are mostly keywords, while some ground-truth labels are keyphrases encompassing multiple keywords. Our method excels at predicting labels that align more closely with the ground-truth label space.

However, our model exhibits comparatively lower performance on the Reuters-21578 and Amazon-531 datasets. Regarding Reuters-21578, this discrepancy is attributable to two primary factors. Firstly, this dataset includes a higher proportion of long-tail labels compared to other datasets. Secondly, some ground-truth labels consist of abbreviations, while our model generates only the full versions, resulting in lower semantic similarity scores. For the Amazon-531 dataset, the initially generated label space by X-MLClass is only one-third of the ground-truth size. Despite adding additional labels through the label space improvement stage, the predicted label space remains less than half of the ground-truth space size, leading to a lower coverage score.

5.6 Zero-shot Text Classification Accuracy

We present the comprehensive zero-shot performance across all methods in Table 3. The results unequivocally demonstrate that our framework consistently outperforms nearly all baseline models. Notably, the P@3 scores of X-MLClass surpass those of the baseline methods across all datasets. This observation implies that training the zero-shot classifier for both the keyphrases set and the chunk set, followed by merging the results, enhances the multi-label performance. Specifically, our chunk splitting procedure increases the likelihood of finding the less dominant labels for each document, as these labels may become dominant in smaller chunks. Similarly, our approach improves the accurate prediction of tail labels by the classifier, contributing to the overall MLTC performance.

Table 4: Label Coverage Score Improvement Results.

Dataset	Initial	After Improvement	Δ
AAPD	51.02	67.34	+16.32%
Reuters-21578	18.89	25.55	+6.66%
RCV1-V2	40.78	46.60	+5.82%
DBPedia-298	51.68	53.02	+1.34%
Amazon-531	19.96	21.67	+1.71%

5.7 Label Space Coverage Improvement

Table 4 shows that iteratively updating the label space leads to an enhancement in label coverage across all datasets. Figure 2 visually represents the incremental coverage during each iteration across AAPD and RCV1-V2 datasets. Notably, the improvement is more pronounced for datasets with smaller initial label space sizes. This finding aligns with expectations, as DBPedia-298 and Amazon-531 exhibit significantly larger label space sizes compared to the other three datasets, rendering label space improvement more challenging. Additionally, the criteria for adding new labels must align with all existing ones in the generated label space, presenting a greater challenge in expanding larger label spaces. Moreover, DBPedia-298 and Amazon-531 feature hierarchical labels, with inclusive relationships. For example, consider "health_care" and "health_personal_care", where "health_care" acts as the parent node of "health_personal_care". Despite the model suggesting they are the same label due to higher semantic similarity scores, these labels represent distinct concepts in real-world scenarios, presenting challenges in adding new labels during the label space improvement process.

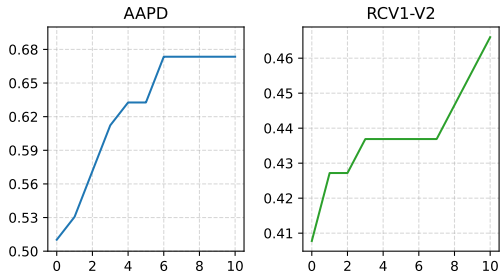

Figure 2: Improvement of Label Coverage across Iterations for the AAPD and RCV1-V2 datasets.

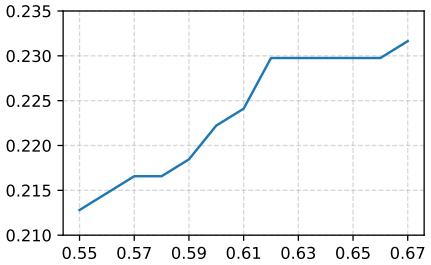

Figure 3: Improvement of Label Coverage for Amazon-531 by changing tail labels addition criterion.

5.8 Label Coverage with Human Evaluation

Our model encounters challenges in generating labels exactly matching the ground-truth label space. Consequently, there exists a possibility that, despite the same meanings, our model-generated label may not align with the ground truth using only semantic similarity score calculation. For instance, within Reuters-298, certain ground-truth labels are abbreviations, while our model generates the full-word version, leading to a lower semantic similarity score than the actual score. As shown in Table 5, the ground-truth label "acq" corresponds to our predicted label "acquisitions," possessing identical meanings, yet their semantic similarity score falls below 30%.

In the Amazon-531 dataset, many ground-truth labels consist of phrases, complicating semantic similarity score calculation. Achieving high scores requires all words in our predicted phrase to match the ground-truth label precisely. However, predicting a similar-meaning phrase with different individual words is common, leading to an overall similarity score lower than the actual score. As evident in Table 5, "electrical_safety" and "electronics_troubleshooting" are identical labels, but their semantic similarity scores are lower, treated as distinct labels in our setting.

Considering these factors, the actual coverage score of our predicted label space compared to the ground-truth label space is likely higher than the presented result in Table 2.

5.9 Ablation Study for Amazon-531 Dataset

The label space for the Amazon-531 dataset significantly surpasses that of the other datasets. To address this discrepancy and enhance label coverage, we have customized hyperparameters specifically for the Amazon-531 dataset. Using the same hyperparameter setting as the other datasets would result in a final label space that is only half the size of the ground-truth label space. In our current setting, we tailored the similarity score, which serves as the boundary for adding tail labels to the existing label space. As depicted in Figure 3, we observe that increasing the similarity score facilitates the addition of more labels to the predicted label space, leading to an improvement in the coverage score.

Table 5: Matching pairs between the ground-truth labels and the predicted labels through human evaluation.

Ground-truth	Predicted Label
acq	acquisitions
money-fx	monetary policy
earn	earnings
plug_play_video_games	gaming_electronics
electrical_safety	electronics_troubleshooting
teether	baby_dental_care

6 Conclusion and Future Work

We attack a novel and challenging problem, open-world MLTC with extremely weak supervision. In this scenario, only raw documents are available, lacking labeled data or a ground-truth label space. Our LLM-based framework, X-MLClass, is designed to overcome this challenge by discovering a practical label space and constructing an MLTC classifier for label prediction. Notably, it excels in identifying long-tail labels, arguably the most challenging aspect in MLTC problems. Our experiment results show that X-MLClass surpasses baselines in terms of ground-truth label coverage and exhibits higher zero-shot text classification performance compared to top-ranking models.

Despite our model's success in generating some tail labels, a considerable number of tail labels remain undiscovered. Future work should focus on refining our approach to capture more long-tail labels. Subsequent studies could explore methodologies tailored for datasets featuring significantly larger label spaces, contributing to the broader applicability of our model.

Limitations

Our work aims to discover the label space from extensive input text documents and then construct a multi-label text classifier. The most formidable challenge in this problem setting revolves around label space construction — how can we discover the labels, especially the long-tail ones? Therefore, our primary focus is on developing a novel method to address this challenge; we didn't propose any new zero-shot multi-label text classifier, since it is beyond the scope of this paper. Given that our proposed X-MLClass starts with a subset of documents, its efficacy may be limited for extremely long-tail labels (e.g., those occurring less frequently than 0.0001% of the documents). Alternatively, a considerably large subset would be required, potentially incurring significant computational costs from LLM. While our evaluation includes a diverse set of datasets, there is potential for further extension to more challenging datasets with an exceptionally large label space (e.g., over 1000 different labels are expected).

References

Charu C Aggarwal, Alexander Hinneburg, and Daniel A Keim. 2001. On the surprising behavior of distance metrics in high dimensional space. In *Database Theory—ICDT 2001: 8th International Conference London, UK, January 4–6, 2001 Proceedings 8*, pages 420–434. Springer.

Dimo Angelov. 2020. Top2vec: Distributed representations of topics. *arXiv preprint arXiv:2008.09470*.

Kush Bhatia, Himanshu Jain, Purushottam Kar, Prateek Jain, and Manik Varma. 2015. Locally non-linear embeddings for extreme multi-label learning. *arXiv preprint arXiv:1507.02743*.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Maarten De Raedt, Fréderic Godin, Thomas Demeester, and Chris Develder. 2023. Idas: Intent discovery with abstractive summarization. *arXiv preprint arXiv:2305.19783*.

Franca Debole and Fabrizio Sebastiani. 2005. An analysis of the relative hardness of reuters-21578 subsets. *Journal of the American Society for Information Science and technology*, 56(6):584–596.

Cédric Févotte and Jérôme Idier. 2011. Algorithms for nonnegative matrix factorization with the β-divergence. *Neural computation*, 23(9):2421–2456.

Ariel Gera, Alon Halfon, Eyal Shnarch, Yotam Perlitz, Liat Ein-Dor, and Noam Slonim. 2022. Zero-shot text classification with self-training. *arXiv preprint arXiv:2210.17541*.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. 2015. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic web*, 6(2):167–195.

David D Lewis, Yiming Yang, Tony Russell-Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research. *Journal of machine learning research*, 5(Apr):361–397.

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*, pages 115–124.

Ziwen Liu, Josep Grau-Bove, and Scott Allan Orr. 2022. Bert-flow-vae: A weakly-supervised model for multi-label text classification.

Julian McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. *Proceedings of the 7th ACM conference on Recommender systems*.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Marc Pàmies, Joan Llop, Francesco Multari, Nicolau Duran-Silva, César Parra-Rojas, Aitor González-Agirre, Francesco Alessandro Massucci, and Marta Villegas. 2023. A weakly supervised textual entailment approach to zero-shot text classification. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 286–296.

Chau Minh Pham, Alexander Hoyle, Simeng Sun, and Mohit Iyyer. 2023. Topicgpt: A prompt-based topic modeling framework. *arXiv preprint arXiv:2311.01449*.

Anthony Rios and Ramakanth Kavuluru. 2018. Few-shot and zero-shot multi-label learning for structured label spaces. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Conference on Empirical Methods in Natural Language Processing, volume 2018, page 3132. NIH Public Access.

Jiaming Shen, Wenda Qiu, Yu Meng, Jingbo Shang, Xiang Ren, and Jiawei Han. 2021. Taxoclass: Hierarchical multi-label text classification using only class names. In *NAAC'21: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,{NAACL-HLT} 2021*, volume 2021.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A Smith, Luke Zettlemoyer, and Tao Yu. 2022. One embedder, any task: Instruction-finetuned text embeddings. *arXiv preprint arXiv:2212.09741*.

Tianle Wang, Zihan Wang, Weitang Liu, and Jingbo Shang. 2023a. Wot-class: Weakly supervised open-world text classification. *arXiv preprint arXiv:2305.12401*.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020a. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33:5776–5788.

Zihan Wang, Dheeraj Mekala, and Jingbo Shang. 2020b. X-class: Text classification with extremely weak supervision. *arXiv preprint arXiv:2010.12794*.

Zihan Wang, Jingbo Shang, and Ruiqi Zhong. 2023b. Goal-driven explainable clustering via language descriptions. *arXiv preprint arXiv:2305.13749*.

Yuanhao Xiong, Wei-Cheng Chang, Cho-Jui Hsieh, Hsiang-Fu Yu, and Inderjit Dhillon. 2021. Extreme zero-shot learning for extreme text classification. *arXiv preprint arXiv:2112.08652*.

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*.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. *arXiv preprint arXiv:1909.00161*.

Yuwei Zhang, Zihan Wang, and Jingbo Shang. 2023. ClusterLLM: Large language models as a guide for text clustering. pages 13903–13920. Association for Computational Linguistics.

Yuwei Zhang, Haode Zhang, Li-Ming Zhan, Xiao-Ming Wu, and Albert Lam. 2022. New intent discovery with pre-training and contrastive learning. *arXiv preprint arXiv:2205.12914*.

A Prompt Templates for Dominant Label

Here is the prompt we use to find the dominant label for the selected document:

```
Which label in the label space ground-truth labels[i] is the dominant label that covers more than 50 percent of the below content? Please output the dominant label only if exists or output 'NO' if there are no dominant labels.
Documents[i]
```

B Label Space refinement with human involvement

Human experts play a crucial role in refining the label space generated by LLM. For instance, when the cosine similarity score between two labels falls between 0.55 and 0.65, indicating a certain degree of semantic similarity, human intervention is preferred to determine whether these labels are synonymous. Synonyms need to be identified and treated accordingly, with one of them being removed from the label space. However, there is also the case that these two labels may represent concepts from different scopes; for example, "health_care" and "health_personal_care." In such instances, human judgment is necessary to detect and treat them as separate labels.

Furthermore, some predicted labels may contain multiple meanings, necessitating human intervention to split them into distinct labels. For instance, if a predicted label is "computer vision and machine learning," it is evident that the label should be divided into two separate labels. These judgments require human expertise for accurate and context-aware decisions.

C Datasets Detailed Information

- **AAPD** (Yang et al., 2018) contains computer science papers. The labels are research topics.
- **Reuters**-21578 (Debole and Sebastiani, 2005) is a collection of news articles from the Reuters financial newswire service in 1987. The labels are the news topics.
- **RCV1-V2** (Lewis et al., 2004) contains categorized newswire articles by Reuters Ltd. The labels are the news topics.
- **DBPedia**-298 (Lehmann et al., 2015) are extracted from Wikipedia articles. The labels are the article categories.

- **Amazon**-531 (McAuley and Leskovec, 2013) encompasses product reviews and associated metadata. The labels are the product tags.