Leveraging Text-to-Text Transformers as Classifier Chain for Few-Shot Multi-Label Classification

Quang Anh Nguyen[1,2,3], **Nadi Tomeh**[2], **Mustapha Lebbah**[1,2], **Thierry Charnois**[2], **Hanene Azzag**[2],

[1]DAVID Lab, University of Versailles, Paris-Saclay University,
[2]LIPN, CNRS UMR 7030, Université Sorbonne Paris Nord, [3]Regulatory IA, Groupe BPCE
{quang-anh.nguyen, mustapha.lebbah}@uvsq.fr,
{nadi.tomeh, thierry.charnois,hanene.azzag}@lipn.univ-paris13.fr

Abstract

Multi-label text classification (MLTC) is an essential task in NLP applications. Traditional methods require extensive labeled data and are limited to fixed label sets. Extracting labels with large language models (LLMs) is more effective and universal, but incurs high computational costs. In this work, we introduce a distillation-based T5 generalist model for zero-shot MLTC and few-shot fine-tuning. Our model accommodates variable label sets with general domain-agnostic pretraining, while modeling dependency between labels. Experiments show that our approach outperforms baselines of similar size on three few-shot tasks. Our code is available at repository.

1 Introduction

Multi-label text classification (MLTC) powers numerous NLP applications, yet supervised systems remain brittle when adapting to a *new domain* with limited supervised data, while domain-specific annotation is costly. Despite strong zero- and few-shot capabilities (Lan et al., 2024; Zhu and Zamani, 2024; Tabatabaei et al., 2025), LLMs are expensive to deploy (Park et al., 2024) and sensitive to prompting (Zhuo et al., 2024; Peskine et al., 2023). Most few-shot methods still train multiple one-vs-all heads: attention-based (Chalkidis et al., 2020), prompt-based (Schick and Schütze, 2021a), or contrastive (Tunstall et al., 2022), etc. Since each label is hard-coded into the parameters, these models cannot be pre-trained once and reused, and they ignore inter-label dependencies.

Smaller encoder–decoder LMs, e.g. T5 (Raffel et al., 2020), can instead emit labels sequentially, capturing label dependencies. Early variants like SGM (Yang et al., 2018), EncT5 (Liu et al., 2022; Kementchedjhieva and Chalkidis, 2023), improve fully supervised MLTC, but they (i) leave label semantics unused and (ii) suffer from the order bias inherited from classifier chains (Read et al., 2021).

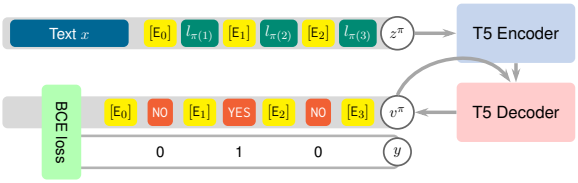

Figure 1: Modeling MLTC as text-to-text with T5. Notations: labels ($l_.$), sentinel tokens ([E_.]).

Order-agnostic training (Tsai and Lee, 2019), DEformer (Alcorn and Nguyen, 2021), or OTSeq2Set (Cao and Zhang, 2022) reduce this bias yet still deploy with an arbitrary order, while dynamic order learners such as DLOL (Li et al., 2024) add inference latency.

Our approach (section 2) trains a T5 in a three-stage procedure, to jointly embed texts and labels, learn a permutation-marginal set scorer, and finally decode with a single chain. Label semantics are encoded within the input sequence with no additional parameters. Unlike order-agnostic or dynamic methods, our approach balances between order-invariance and inference efficiency. Our method yields state-of-the-art results on three few-shot MLTC benchmarks while preserving single-sequence inference latency, bridging set-invariant theory and practical few-shot deployment.

2 Sequence-to-Set T5 Chain

We cast MLTC as *sequence–to–set* prediction with a T5 encoder–decoder (section 2.1 and figure 1) and propose a three-stage training procedure (section 2.2 and figure 2).

2.1 Formulation

Notation. Let $L = \{l_1, \ldots, l_m\}$ be the candidate label set. Let x be the text to classify and y its label indicator, where $y(l_j) = 1$ if and only if l_j is a correct label of x. Let $\pi \in \mathfrak{S}_m$ a permutation over L, or **chain**. The input sequence assembles x and

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 26941–26950
November 4-9, 2025

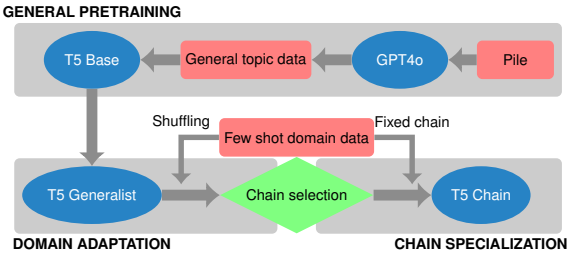

Figure 2: Overall pipeline of the training process of T5 Chain (section 2).

all labels ordered by π:

$$z^{\pi} = x\ [\mathsf{E}_1]\, l_{\pi(1)}\ [\mathsf{E}_2]\, l_{\pi(2)} \ldots [\mathsf{E}_m]\, l_{\pi(m)}, \quad (1)$$

where sentinel tokens $[\mathsf{E}_j]$ act as hard anchors, demarcating each multi-token label. The encoder sees text-label and label-label interactions in a single forward pass. The output v^{π} is a sequence of YES/NO tokens and sentinel markers, where v^{π}_{2j} is YES if $y(l_{\pi(j)}) = 1$ and NO otherwise. This formulation encapsulates the input text, label descriptors, and the label indicator within two sequences for text-to-text models (figure 1).

Permutation-marginal likelihood. Ideally, prediction of a label set based on sequence generation induces the Janossy mixture (Murphy et al., 2019):

$$\log p_\theta(y \mid x) = \frac{1}{m!} \sum_{\pi \in \mathfrak{S}_m} \sum_{j=1}^{m} \log p_\theta\left(v^{\pi}_{2j} \mid z^{\pi}, v^{\pi}_{<2j}\right) \quad (2)$$

which is *permutation-invariant* and reduces to a classical chain when the sum collapses to one order. In practice, we Monte-Carlo approximate (2) by randomly sampling π; section 2.2 explains how the three training stages leverage this fact.

2.2 Shuffle-then–Select Training

Stage 1: Generalist pre-training (GP)

Vanilla T5 is pretrained for span-corruption; we first teach it to classify topics in an *order-agnostic* manner. In our work, we employ the `t5-v1.1-base`, which has 220M parameters.

Data creation We sample 10,000 texts from the PILE (Gao et al., 2020), a large-scale and diverse corpus. Each sampled text is then processed using GPT-4o, to extract relevant topics. The resulting dataset consists of 9,789 texts, each associated with several topical tags, among a pool $\mathcal{T}$ of 63,852 unique topics.

Dataset	D	L	D/L	L/D	W/D
Eurlex	45k	21	6949	3.24	583
AAPD	53840	53	2429	2.39	163
Reuters	7769	90	106	1.23	130

Table 1: Summary of datasets: training documents (D), number of labels (L), average documents per label (D/L), average labels per document (L/D), average document length (W/D).

Training For each text instance x during training, 80% positive labels are randomly sampled, while an equal number of negative labels are drawn from the topic pool. With these label candidates L_x and a random order $\pi \in \mathfrak{S}_{|L_x|}$, we minimize the autoregressive LM loss:

$$\mathcal{L}^{\mathrm{GP}} = -\mathbb{E}_{x, L_x, \pi} \sum_{t} \log p_\theta(v^{\pi}_t \mid z^{\pi}, v^{\pi}_{<t}), \quad (3)$$

which amounts to a π-SGD estimation of (2). The resulting **T5 Generalist** acquires a general understanding of the topic classification task in an order-invariant, domain-agnostic manner.

Stage 2: Order-marginal domain adaptation (DA)

We fine-tune with BCE objective that only scores YES/NO token positions:

$$\mathcal{L}^{\mathrm{DA}} = -\mathbb{E}_{(x,y),\pi} \sum_{j=1}^{m} \log q_\theta\left(v^{\pi}_{2j} \mid z^{\pi}, v^{\pi}_{<2j}\right), \quad (4)$$

where q_θ is the softmax of p_θ restricted to {YES, NO}. Label shuffling induced by π persists, keeping the model permutation-robust, while familiarizing it with domain topics.

Stage 3: Chain specialisation (CS)

For each chain in a set of random chains we compute the average BCE loss equation (4) on the validation set and keep the best chain $\pi^{\star}$. With this fixed order we continue optimization:

$$\mathcal{L}^{\mathrm{CS}} = -\mathbb{E}_{(x,y)} \sum_{j=1}^{m} \log q_\theta\left(v^{\pi^{\star}}_{2j} \mid z^{\pi^{\star}}, v^{\pi^{\star}}_{<2j}\right). \quad (5)$$

Removing permutation noise polishes conditional dependencies along $\pi^{\star}$ while keeping inference as cheap as decoding one sequence.

2.3 Inference

At test time, the chain $\pi^{\star}$ is reused. The model autoregressively generates $v^{\pi^{\star}}$ under

	Eurlex			AAPD			Reuters		
Method	mF1	MF1	IF1	mF1	MF1	IF1	mF1	MF1	IF1
NSP (BERT)	12.24	9.82	9.65	8.81	8.15	8.77	**5.68**	5.67	3.47
PET (RoBERTa)	24.57	21.60	24.27	8.71	8.09	8.68	3.57	5.01	3.71
T5 Generalist	**42.39**	**31.22**	**41.85**	**16.73**	**15.46**	**16.61**	5.54	**14.06**	**5.49**

Table 2: Performance comparison in zero-shot setup. Best results are highlighted in bold.

	Eurlex			AAPD			Reuters		
Method	mF1	MF1	IF1	mF1	MF1	IF1	mF1	MF1	IF1
Head (RoBERTa)	$70.0_{0.8}$	$34.8_{1.2}$	$69.2_{1.0}$	$41.5_{1.1}$	$7.0_{0.6}$	$32.0_{0.9}$	$55.9_{4.7}$	$2.1_{0.3}$	$48.0_{5.4}$
Head (T5 Encoder)	$47.9_{0.9}$	$11.9_{0.3}$	$47.7_{0.6}$	$39.9_{2.0}$	$5.7_{1.0}$	$30.3_{2.1}$	$61.7_{0.7}$	$56.6_{1.0}$	$56.3_{0.9}$
PET (RoBERTa)	$70.8_{0.6}$	$45.8_{2.6}$	$70.9_{0.7}$	$52.0_{1.2}$	$22.1_{2.4}$	$45.2_{2.1}$	$\mathbf{85.6}_{0.8}$	$48.6_{1.2}$	$87.6_{1.1}$
SetFit	$69.9_{0.9}$	$35.7_{2.6}$	$70.6_{0.8}$	$58.9_{0.7}$	$30.9_{1.4}$	$58.2_{1.0}$	$84.8_{0.9}$	$36.8_{2.1}$	$88.5_{0.6}$
BERT-LWAN	$69.0_{1.4}$	$35.0_{1.8}$	$68.4_{1.5}$	$45.5_{3.8}$	$11.9_{5.5}$	$37.8_{5.7}$	$74.6_{4.1}$	$11.6_{7.1}$	$73.4_{5.5}$
T5 Chain	$\mathbf{73.3}_{0.2}$	$\mathbf{56.4}_{0.4}$	$\mathbf{73.8}_{0.2}$	$\mathbf{60.3}_{0.4}$	$\mathbf{42.6}_{0.6}$	$\mathbf{61.1}_{0.5}$	$84.8_{0.8}$	$\mathbf{57.7}_{2.8}$	$\mathbf{88.9}_{0.8}$

Table 3: Performance comparison in few-shot finetuning. Best results are highlighted in bold.

structural constraints (sentinel tokens at odd steps, YES/NO at even steps). The normalized $q_\theta\left(v_{2j}^{\pi^\star} = \texttt{YES} \mid z^{\pi*}, v_{<2j}^{\pi^\star}\right)$ serves directly as a relevant score between x and l_j, calculated with no extra latency, at the cost of a single sequence.

3 Experiments

3.1 Data

We experiment on three MLTC datasets summarized in table 1: **Eurlex** EU legislation (Chalkidis et al., 2019) with legislative English documents from EUR-lex[1], annotated with concepts from EuroVoc[2], we use top level labels of its hierarchy; **AAPD** ArXiv Academic Paper Dataset (Yang et al., 2018) consists of abstracts from arXiv academic papers, tagged with subject categories of arXiv's taxonomy; and **Reuters** The Reuters-21578 dataset (Apté et al., 1994), a popular dataset from the Reuters financial newswire service in 1987.

3.2 Baselines

We compare our models with the following approaches, which we adapt and train ourselves (appendix B): Head (Sun et al., 2020), PET (Schick and Schütze, 2021a,b), SetFit (Tunstall et al., 2022), BERT-LWAN (Mullenbach et al., 2018; Chalkidis et al., 2020), and NSP (Yin et al., 2019; Ma et al., 2021).

[1] https://eur-lex.europa.eu/
[2] http://eurovoc.europa.eu/

3.3 Setup and Evaluation

To simulate the few-shot setting, we sample $8|L|$ training examples. The T5 Generalist and all baselines are then fine-tuned on these data and evaluated on the original test sets. Our implementation can be found at our repository.

We report the mean and standard deviation across five data splits. As our focus is on architectural comparisons rather than thresholding (Fan and Lin, 2007; Al-Otaibi et al., 2014), we adopt a fixed threshold of 0.5. Performance is measured using F1 scores: micro (mF1), macro (MF1), and instance-based (IF1).

4 Results and Discussion

4.1 Main Results

From Table 2, T5 Generalist shows strong zero-shot performance against NSP and PET. Its high MF1 scores suggest robustness to label sparsity.

Table 3 summarizes the results for few-shot fine-tuning (additional metrics in appendix D). T5 Chain consistently outperforms all baselines across three datasets, demonstrating robustness and strong generalization. On Eurlex, it improves mF1 and IF1 by around 3 points over PET. On AAPD, it surpasses SetFit by around 2 points in both metrics. For Reuters, T5 Chain achieves the best MF1 and IF1 (57.7 and 88.9, respectively), reflecting good global performance and rare-label robustness.

Notably, T5 Chain demonstrates particularly strong macro-level performance, surpassing other methods by over 10 points in MF1 across all datasets. This is important in MLTC where label

GP	DA	CS	Eurlex	AAPD	Reuters
✓	✓	✓	68.24	55.43	78.76
	✓	✓	61.00	47.13	55.17
✓		✓	$65.15_{2.31}$	$52.06_{1.61}$	$66.42_{2.06}$
✓	✓		$66.30_{0.81}$	$53.78_{0.84}$	$74.94_{0.04}$

Table 4: F1 scores (averaging mF1, MF1, IF1) under different configurations of T5 Chain.

distribution is often long-tailed and dominated by rare labels. Additionally, low standard deviations (e.g., 0.2–0.8, except for MF1 on Reuters) highlight the method's stable performance across different few-shot splits.

4.2 Ablation Studies

Q1: How do different stages of the training procedure contribute? Table 4 compares average F1 scores of different configurations, showing that the full procedure performs best overall. Removing GP causes a sharp performance drop, suggesting that foundational knowledge from diverse data is critical for few-shot downstream tasks. CS adds an extra fine-tuning step to refine the model's performance by restricting to a specific order, moderately increases final scores, while simplifying inference. The effect of DA and CS [3] is less significant than GP. Finally, omitting DA leads to a slightly larger drop, highlighting its intermediate role between GP and CS. Variation across chains decreases with DA, showing that shuffling the labels, rather than training on a fixed chain, mitigates order dependence. Nevertheless, the overall performance remains stable with a relative deviation below 3.5%, showing robustness of T5 Chain to orderings.

Q2: Do T5 models generate responses following the given label order? [4]

While this behaviour is intuitively expected, the information for such alignment is not provided: v^{π}_{2j} does not necessarily corresponds to $l_{\pi(j)}$. To verify this hypothesis, we analyse the cross-attention matrix [5], in figure 3, between output v^{π}_{2j} and the input tail $[\mathsf{E}_0]\ l_{\pi(1)}\ [\mathsf{E}_1]\ \ldots\ l_{\pi(m)}$. The observed

[3]The chain selection step (section 2.2) only makes sense after adapting to the label space (with DA) and before fixing a chain (with CS). For the last two configurations of table 4, we drop this step and instead report mean and standard deviation across chains for the study of sensitivity.

[4]Models used for Q2 and Q3 are before CS, as we intend to study these questions independently of the selected chain.

[5]Aggregated by max pooling over decoder layers and heads (to select the most significant signals) and mean pooling over 200 examples and 50 chains (we display the matrix of Eurlex for readability, other datasets show similar patterns).

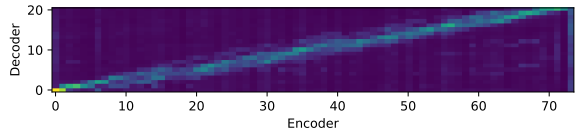

Figure 3: Cross attention weights, mean over chains and examples of Eurlex.

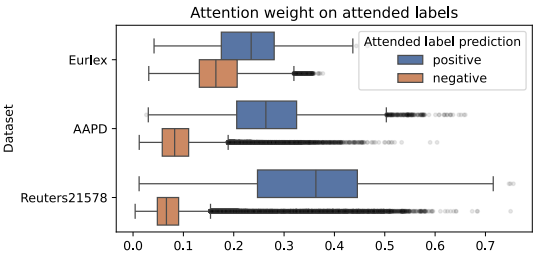

Figure 4: Distributions of decoder causal attention weights by attended label category.

highlighted diagonal suggests that the output generation attends nearly linearly to the input sequence, proving that the model truly predicts labels in the same order as the input sequence.

Q3: Do T5 models learn label dependency and exploit it in autoregressive prediction? We analyze how the models attend to previously predicted labels by collecting decoder causal attention weights over 200 examples and 50 chains. Each score represents the attention from a target label to an attended label, across 12 layers and 12 heads inside the model, shown in figure 4. Statistically, positive (YES) labels receive higher attention than negative (NO) ones. This indicates that the models leverage label dependencies during autoregressive generation.

Q4: When is modeling label dependence not beneficial? On Reuters, we notice that T5 Chain improves only marginally compared to SetFit and PET. We attribute this to sparse label correlations in Reuters. Figure 5 shows the estimated distribution of NPMI across all label pairs of the training sets (Appendix E, Bouma 2009). Observe that Eurlex and AAPD exhibit an NPMI more evenly spread out in the range $\pm$ 0.25, whereas the distribution is heavily concentrated around 0 for Reuters. This shows that in Reuters, label dependence is statistically less pronounced, explaining why the classifier chain does not gain much advantage while predicting labels sequentially.

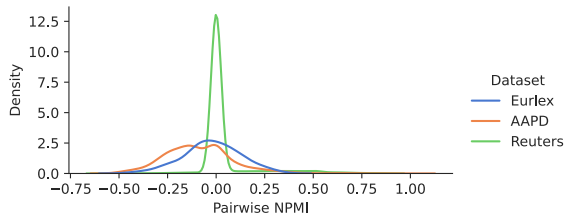

Figure 5: Distribution of pairwise normalized pointwise mutual information (NPMI) over three datasets.

5 Related Work

Beyond works reviewed in previous sections and the baselines detailed in appendix B, this section highlights complementary threads that inform our design and findings.

Set prediction and permutation invariance. A complementary approach to classifier chains treats MLTC as a *set* problem, using order-robust architectures or losses. Zaheer et al. (2017) provides a representation theorem for permutation-invariant/equivariant functions, while Lee et al. (2019) introduces attention over sets without positional bias. In structured prediction, Hungarian matching losses (Carion et al., 2020) align unordered outputs to gold sets, inspiring sequence-to-set objectives in NLP. Related *multiset* criteria marginalize or match over permutations to reduce order bias (Welleck et al., 2018). These works motivate our *shuffle-then-select* recipe (section 2): stochastic label shuffling approximates permutation marginalization, then a single chain is fixed to maintain low inference cost.

Label semantics and open-vocabulary classification. Some approaches reduce supervision using *textual labels*. Shen et al. (2021) performs hierarchical MLTC with class names and self-training, while Gao et al. (2023) uses label descriptions for better zero-shot transfer (*label-description tuning*). Hierarchy-aware reasoning further constrains predictions: Mao et al. (2019) combines reinforcement learning with logical rollback for zero/few-shot taxonomy traversal. Inspired by these, our work joins label texts and the input to ground predictions in lexical semantics, aiding with rare labels. Similarly, GLiNER (Zaratiana et al., 2024) also conditions on text and label names for zero-shot extraction, but differs in granularity (span vs. document) and decoding (parallel vs. our autoregressive chain).

Meta-learning and preserving zero-shot ability. Meta-learning allows rapid adaptation to new labels or domains in MLTC, complementing contrastive and prompt-based methods (Wu et al., 2019). Recent work (Chen et al., 2025) shows supervised adaptation can harm zero-shot generalization and suggests strategies to *preserve* it. Similarly, we distill a domain-agnostic *generalist* model, adapt it via label shuffling, and fix a single efficient chain, maintaining generalization while modeling instance-level dependencies.

Set-aware generation in IE and structured NLP. Recent generative information extraction treats outputs (entities, relation triplets, etc.) as *sets*, using permutations or set-matching losses to make order-agnostic decorders (Paolini et al., 2021). This reduces hallucinations and exposure errors, consistent with our observation that shuffling improves stability in few-shot MLTC.

6 Conclusion

We proposed a T5-based architecture for few-shot multi-label text classification that captures label dependencies through autoregressive generation. Our three-stage pipeline - pretraining, adaptation, and specialization - supports robust cross-domain generalization. Experiments demonstrate strong zero- and few-shot performance, while ablations confirm the model's ability to follow MLTC structures and effectively leverage label dependency information, highlighting its potential for multi-label problems.

7 Limitations

Despite its strong performance, our approach has several limitations. Firstly, the reliance on sequential label prediction introduces inference latency, particularly when dealing with long label sequences, both in computation time and memory usage. Chunking label sequences would require partition selection and loss of global dependency, and may lead to a trade-off between efficiency and performance. The effectiveness of our method also hinges on selecting an optimal label order, which currently relies on empirical evaluation. However, a reliable and generalized method for chain selection, especially with limited supervised data, is left for future research. Lastly, our method depends on the quality of the distillation data used during pretraining; denoising or enhancements of labels from the LLM could potentially boost downstream generalization of our approach.

References

Reem Al-Otaibi, Peter Flach, and Meelis Kull. 2014. Multi-label classification: A comparative study on threshold selection methods. In *First International Workshop on Learning over Multiple Contexts (LMCE) at ECML-PKDD*, volume 29.

Michael A. Alcorn and Anh Totti Nguyen. 2021. The DEformer: An order-agnostic distribution estimating transformer. In *ICML Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models*.

Chidanand Apté, Fred Damerau, and Sholom M. Weiss. 1994. Automated learning of decision rules for text categorization. *ACM Transactions on Information Systems*. To appear.

Gerlof J. Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction.

Jie Cao and Yin Zhang. 2022. Otseq2set: An optimal transport enhanced sequence-to-set model for extreme multi-label text classification.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I*, page 213–229, Berlin, Heidelberg. Springer-Verlag.

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.

Ilias Chalkidis, Manos Fergadiotis, Sotiris Kotitsas, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. An empirical study on large-scale multi-label text classification including few and zero-shot labels. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7503–7515, Online. Association for Computational Linguistics.

Si-An Chen, Hsuan-Tien Lin, and Chih-Jen Lin. 2025. Preserving zero-shot capability in supervised fine-tuning for multi-label text classification. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5699–5712, Albuquerque, New Mexico. Association for Computational Linguistics.

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.

Rong-En Fan and Chih-Jen Lin. 2007. A study on threshold selection for multi-label classification. *Department of Computer Science, National Taiwan University*, pages 1–23.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2020. The pile: An 800gb dataset of diverse text for language modeling.

Lingyu Gao, Debanjan Ghosh, and Kevin Gimpel. 2023. The benefits of label-description training for zero-shot text classification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13823–13844, Singapore. Association for Computational Linguistics.

Yova Kementchedjhieva and Ilias Chalkidis. 2023. An exploration of encoder-decoder approaches to multi-label classification for legal and biomedical text. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5828–5843, Toronto, Canada. Association for Computational Linguistics.

Mengfei Lan, Lecheng Zheng, Shufan Ming, and Halil Kilicoglu. 2024. Multi-label sequential sentence classification via large language model. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16086–16104, Miami, Florida, USA. Association for Computational Linguistics.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. 2019. Set transformer: A framework for attention-based permutation-invariant neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3744–3753. PMLR.

Jiangnan Li, Yice Zhang, Shiwei Chen, and Ruifeng Xu. 2024. Enhancing multi-label classification via dynamic label-order learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):18527–18535.

Frederick Liu, Terry Huang, Shihang Lyu, Siamak Shakeri, Hongkun Yu, and Jing Li. 2022. Enct5: A framework for fine-tuning t5 as non-autoregressive models.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Tingting Ma, Jin-Ge Yao, Chin-Yew Lin, and Tiejun Zhao. 2021. Issues with entailment-based zero-shot text classification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 786–796, Online. Association for Computational Linguistics.

Yuning Mao, Jingjing Tian, Jiawei Han, and Xiang Ren. 2019. Hierarchical text classification with reinforced

label assignment. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 445–455, Hong Kong, China. Association for Computational Linguistics.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. 2018. Explainable prediction of medical codes from clinical text. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1101–1111, New Orleans, Louisiana. Association for Computational Linguistics.

Ryan L. Murphy, Balasubramaniam Srinivasan, Vinayak Rao, and Bruno Ribeiro. 2019. Janossy pooling: Learning deep permutation-invariant functions for variable-size inputs.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, RISHITA ANUBHAI, Cicero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured prediction as translation between augmented natural languages. In *International Conference on Learning Representations*.

Yeonhong Park, Jake Hyun, SangLyul Cho, Bonggeun Sim, and Jae W. Lee. 2024. Any-precision llm: Low-cost deployment of multiple, different-sized llms.

Youri Peskine, Damir Korenčić, Ivan Grubisic, Paolo Papotti, Raphael Troncy, and Paolo Rosso. 2023. Definitions matter: Guiding GPT for multi-label classification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4054–4063, Singapore. Association for Computational Linguistics.

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. *Journal of Machine Learning Research*, 21(140):1–67.

Jesse Read, Bernhard Pfahringer, Geoffrey Holmes, and Eibe Frank. 2021. Classifier chains: A review and perspectives. *Journal of Artificial Intelligence Research*, 70:683–718.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021a. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Timo Schick and Hinrich Schütze. 2021b. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352, Online. Association for Computational Linguistics.

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 *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4239–4249, Online. Association for Computational Linguistics.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2020. How to fine-tune bert for text classification?

Seyed Amin Tabatabaei, Sarah Fancher, Michael Parsons, and Arian Askari. 2025. Can large language models serve as effective classifiers for hierarchical multi-label classification of scientific documents at industrial scale? In *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track*, pages 163–174, Abu Dhabi, UAE. Association for Computational Linguistics.

Che-Ping Tsai and Hung-Yi Lee. 2019. Order-free learning alleviating exposure bias in multi-label classification.

Lewis Tunstall, Nils Reimers, Unso Eun Seo Jo, Luke Bates, Daniel Korat, Moshe Wasserblat, and Oren Pereg. 2022. Efficient few-shot learning without prompts. *arXiv preprint arXiv:2209.11055*.

Sean Welleck, Zixin Yao, Yu Gai, Jialin Mao, Zheng Zhang, and Kyunghyun Cho. 2018. Loss functions for multiset prediction. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Jiawei Wu, Wenhan Xiong, and William Yang Wang. 2019. Learning to learn and predict: A meta-learning approach for multi-label classification. In *Proceedings of the 2019 Conference on Empirical Methods*

in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 4354–4364, Hong Kong, China. Association for Computational Linguistics.

Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018. SGM: Sequence generation model for multi-label classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3915–3926, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. 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 3914–3923, Hong Kong, China. Association for Computational Linguistics.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. 2017. Deep sets. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. 2024. GLiNER: Generalist model for named entity recognition using bidirectional transformer. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5364–5376, Mexico City, Mexico. Association for Computational Linguistics.

Yaxin Zhu and Hamed Zamani. 2024. ICXML: An in-context learning framework for zero-shot extreme multi-label classification. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2086–2098, Mexico City, Mexico. Association for Computational Linguistics.

Jingming Zhuo, Songyang Zhang, Xinyu Fang, Haodong Duan, Dahua Lin, and Kai Chen. 2024. ProSA: Assessing and understanding the prompt sensitivity of LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1950–1976, Miami, Florida, USA. Association for Computational Linguistics.

A General distilled dataset construction

We sample 10,000 texts from the PILE (Gao et al., 2020), a large-scale and diverse corpus widely used for pretraining LMs. Each sampled text is then processed using GPT-4o, along with the following prompt template to extract relevant topics.

```
System message:
**Objective:** Read the given passage and extract important topics in the passage.
**Format Requirements:** The output should be formatted in JSON, containing a list of topics, from the most general to the most specific.
**Extraction Details:**
- The first topic should be the discipline in which the document belongs to, the second topic should be a sub-discipline or field that the passage describe.
- The list should be from the most general topic to the most specific details, and should contain from 5 to 15 topics.
- Try not to copy keywords from the given passage. Topics should be short terms with general meaning, including at most 5 words
**Output Schema:**
<start>
{ "topics": ["Topic 1", "Topic 2", ...] }
<end>

User:
{text}
```

After filtering out invalid responses, we obtain a dataset $\mathcal{D}_{\text{Pile}}$ comprising 9,789 texts, each averaging 813 words in length, accompanied by a pool $\mathcal{T}$ of 63,852 unique topic tags, reflecting a broad and diverse range of topic categories spanning multiple domains (appendix A). On average, each topic appears in 1.66 texts. Each text is tagged with an average of 10.83 topics, mimicking the long-tail nature of labels in MLTC.

Popular topics with frequency greater than 0.1% include *Computer Science, Software Development, Medicine, Law, Sports, Web Development, Technology, Software Engineering, Biology, Programming, Education*, etc. On the other hand, 84% of all topics is tagged only once in the whole dataset, accounting for 50.7% of the total tags, e.g. *Add-ons, Construction Adhesives, Data-driven Decision Making, Higher Education Funding, Hydroelectric Power, Industrial Practices, Marvel, Mexican Culture, Notable individuals, Recreational Vehicles, Sin and Redemption, Trade Rumors, Workplace Issues*, etc.

The Pile corpus, which serves as the text source for our data, is publicly available under the MIT License (https://pile.eleuther.ai). For transparency, we also acknowledge that OpenAI's terms of use apply to all AI-generated annotations (https://openai.com/terms).

Each example contains a text and its positive topics. During the GP stage, we randomly drop 20% of the positive labels and sample negative labels from the pool of all topics in the dataset, so that the number of additional negative samples is equal to the number of existing positive labels. The total number of labels is limited to 100. The order of labels is also shuffled each time the same text

passes through the model T5. The training process takes about 10 hours on an A100 GPU.

B Baselines

Our models are compared with the following approaches, which we adapt and train ourselves. All baselines are trained with the sum of the BCE loss of individual labels.

Head The document is fed to *an encoder*, and the representation of the [CLS] or <\s> token is passed to $|L|$ binary classifier heads, one per label.

PET (Schick and Schütze, 2021a,b) is a prompt-based method where x is transformed into a sequence with the `[MASK]` token. A *verbalizer* maps labels to words, converting the `[MASK]` distribution into class scores.

SetFit (Tunstall et al., 2022) is a metric-based approach with two phases: finetuning a sentence transformers (ST) (Reimers and Gurevych, 2019) in a contrastive manner, then train a classifier on rich text embeddings of the ST. Here we use `paraphrase-mpnet-base-v2`[6] as ST.

BERT-LWAN LWAN (Mullenbach et al., 2018) produced label-wise document representation by learning attention blocks for each label, then further improved by BERT encoder (Devlin et al., 2019; Chalkidis et al., 2020).

NSP (Yin et al., 2019; Ma et al., 2021) show that finetuning BERT (Devlin et al., 2019) on NLI data helps with zero-shot topic classification.

C Finetuning hyperparameters

For few-shot experiments, we finetune T5 models following the methods described in section 2.2. The models are trained for a maximum of 5000 steps for DA and maximum 1000 steps for CS, with the AdamW optimizer (Loshchilov and Hutter, 2019) and a linear decay scheduler. Between DA and CS stages, the validation set is used to select the optimal chain, among $M = 50$ chains sampled randomly from permutations of labels. The training batch size is fixed at 4 due to memory limitations, and the gradient accumulation value is 4. For each dataset and each split of training data, the base learning rate is tuned in {2e-5, 5e-5, 1e-4, 2e-4}. During training, evaluation on the validation set is done every 50 steps to select the best checkpoint. Our implementation is based on `transformers` (Wolf et al., 2020) and experiments are conducted on A100 GPU. Each training step takes approximately 1.04 seconds on average.

[6] https://huggingface.co/sentence-transformers/paraphrase-mpnet-base-v2

D Extended results on few shot finetuning

Table 5 summarizes micro/macro/instance precision and recall for T5 Chain and other baselines over three studied datasets.

E Normalized Pointwise Mutual Information

Pointwise mutual information of a couple of events (x, y) is a measure of how much the actual probability of their co-occurrence $p(x, y)$ differs from the case of independence, $p(x)p(y)$ (Bouma, 2009). PMI can be normalized to $[-1, +1]$, with -1 for never occurring together, 0 for independence, and $+1$ for complete co-occurrence, giving the normalized pointwise mutual information (NPMI)

$$\mathrm{NPMI}(x, y) = \frac{\log \frac{p(x,y)}{p(x)p(y)}}{-\log p(x, y)} \tag{6}$$

Particularly in our case, for a pair of labels $l_j, l_k \in L$ of a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ of texts and label indicators, the NPMI is calculated with their occurrence ($y_i(l_j) = 1$, $y_i(l_k) = 1$):

$$\mathrm{NPMI}(l_j, l_k) = \frac{\log \frac{N \sum_i y_i(l_j) y_i(l_k)}{\sum_i y_i(l_j) \sum_i y_i(l_k)}}{-\log \frac{1}{N} \sum_i y_i(l_j) y_i(l_k)} \tag{7}$$

Figure 5 illustrates the distribution of this index over all pairs of labels for each dataset, using Gaussian kernel density estimation.

Method	mP	mR	mF1	MP	MR	MF1	IP	IR	IF1
					Eurlex				
Head (RoBERTa)	**80.0**$_{1.4}$	62.3$_{1.2}$	70.0$_{0.8}$	56.7$_{3.1}$	30.5$_{1.1}$	34.8$_{1.2}$	79.4$_{1.5}$	65.7$_{1.0}$	69.2$_{1.0}$
Head (T5 Encoder)	68.7$_{2.6}$	36.7$_{0.4}$	47.9$_{0.9}$	14.3$_{1.8}$	11.9$_{0.2}$	11.9$_{0.3}$	64.8$_{1.6}$	40.4$_{0.4}$	47.7$_{0.6}$
PET (RoBERTa)	73.9$_{2.5}$	68.2$_{2.4}$	70.8$_{0.6}$	**59.0**$_{0.7}$	43.5$_{3.8}$	45.8$_{2.6}$	76.2$_{1.9}$	71.4$_{2.2}$	70.9$_{0.7}$
SetFit	79.0$_{0.9}$	62.8$_{1.2}$	69.9$_{0.9}$	58.8$_{2.4}$	31.6$_{2.0}$	35.7$_{2.6}$	**80.9**$_{0.8}$	67.0$_{1.1}$	70.6$_{0.8}$
BERT-LWAN	77.1$_{2.1}$	62.4$_{1.1}$	69.0$_{1.4}$	58.0$_{2.2}$	31.4$_{1.6}$	35.0$_{1.8}$	77.3$_{2.2}$	65.8$_{1.2}$	68.4$_{1.5}$
T5 Chain	71.7$_{0.3}$	**74.9**$_{0.3}$	**73.3**$_{0.2}$	54.9$_{0.5}$	**59.9**$_{0.6}$	**56.4**$_{0.4}$	74.8$_{0.2}$	**77.8**$_{0.3}$	**73.8**$_{0.2}$
					AAPD				
Head (RoBERTa)	**79.8**$_{2.7}$	28.1$_{1.1}$	41.5$_{1.1}$	14.6$_{2.5}$	5.8$_{0.5}$	7.0$_{0.6}$	37.5$_{0.7}$	29.4$_{0.9}$	32.0$_{0.9}$
Head (T5 Encoder)	79.6$_{3.3}$	26.7$_{2.1}$	39.9$_{2.0}$	8.9$_{1.9}$	5.1$_{1.0}$	5.7$_{1.0}$	34.7$_{2.3}$	28.3$_{2.0}$	30.3$_{2.1}$
PET (RoBERTa)	74.7$_{2.0}$	40.0$_{1.9}$	52.0$_{1.2}$	40.8$_{3.6}$	17.6$_{2.1}$	22.1$_{2.4}$	53.1$_{2.4}$	42.2$_{2.1}$	45.2$_{2.1}$
SetFit	65.9$_{0.8}$	53.3$_{0.9}$	58.9$_{0.7}$	39.0$_{1.0}$	28.9$_{1.6}$	30.9$_{1.4}$	**65.0**$_{1.2}$	56.2$_{0.9}$	58.2$_{1.0}$
BERT-LWAN	75.5$_{4.2}$	33.0$_{5.1}$	45.5$_{3.8}$	20.2$_{8.9}$	10.1$_{4.4}$	11.9$_{5.5}$	44.7$_{6.5}$	34.7$_{5.6}$	37.8$_{5.7}$
T5 Chain	59.6$_{0.7}$	**61.0**$_{0.6}$	**60.3**$_{0.4}$	**42.6**$_{1.0}$	**44.7**$_{0.7}$	**42.6**$_{0.6}$	63.1$_{0.7}$	**63.8**$_{0.6}$	**61.1**$_{0.5}$
					Reuters				
Head (RoBERTa)	**98.6**$_{0.5}$	39.1$_{4.5}$	55.9$_{4.7}$	3.4$_{0.8}$	1.8$_{0.3}$	2.1$_{0.3}$	48.5$_{5.6}$	47.9$_{5.4}$	48.0$_{5.4}$
Head (T5 Encoder)	95.3$_{0.8}$	45.7$_{0.8}$	61.7$_{0.7}$	3.0$_{0.4}$	2.1$_{0.1}$	56.6$_{1.0}$	56.6$_{1.0}$	56.2$_{0.9}$	56.3$_{0.9}$
PET (RoBERTa)	89.1$_{3.4}$	82.4$_{2.2}$	**85.6**$_{0.8}$	**59.0**$_{4.1}$	46.9$_{1.2}$	48.6$_{1.2}$	88.6$_{1.0}$	88.8$_{1.8}$	87.6$_{1.1}$
SetFit	87.6$_{1.3}$	82.3$_{0.8}$	84.8$_{0.8}$	42.2$_{2.3}$	35.7$_{2.1}$	36.8$_{2.1}$	90.2$_{0.7}$	89.2$_{0.7}$	88.5$_{0.6}$
BERT-LWAN	89.7$_{0.5}$	64.1$_{6.2}$	74.6$_{4.1}$	14.7$_{8.6}$	10.8$_{6.5}$	11.6$_{7.1}$	75.5$_{5.6}$	72.9$_{5.7}$	73.4$_{5.5}$
T5 Chain	82.3$_{1.1}$	**87.5**$_{0.7}$	84.8$_{0.8}$	58.4$_{3.0}$	**66.9**$_{2.0}$	**57.7**$_{2.8}$	**88.8**$_{0.9}$	**91.2**$_{0.8}$	**88.9**$_{0.8}$

Table 5: Performance comparison in few-shot finetuning. Scores include precision (P), recall (R), and F1-score (F1), with different views: micro (m), macro (M), or instance (I). Best results are highlighted in bold.