MADE: A Living Benchmark for Multi-Label Text Classification with Uncertainty Quantification of Medical Device Adverse Events

Raunak Agarwal, Markus Wenzel, Simon Baur,
Jonas Zimmer, George Harvey, Jackie Ma
Department of Artificial Intelligence
Fraunhofer Heinrich Hertz Institute
Berlin, 10587, Germany
Correspondence: jackie.ma@hhi.fraunhofer.de

Abstract

Machine learning in high-stakes domains such as healthcare requires not only strong predictive performance but also reliable uncertainty quantification (UQ) to support human oversight. Multi-label text classification (MLTC) is a central task in this domain, yet remains challenging due to label imbalances, dependencies, and combinatorial complexity. Existing MLTC benchmarks are increasingly saturated and may be affected by training data contamination, making it difficult to distinguish genuine reasoning capabilities from memorization. We introduce MADE, a living MLTC benchmark derived from medical device adverse event reports and continuously updated with newly published reports to prevent contamination. MADE features a long-tailed distribution of hierarchical labels and enables reproducible evaluation with strict temporal splits. We establish baselines across more than 20 encoder- and decoder-only models under fine-tuning and few-shot settings (instruction-tuned/reasoning variants, local/API-accessible). We systematically assess entropy-/consistency-based and self-verbalized UQ methods. Results show clear trade-offs: smaller discriminatively fine-tuned decoders achieve the strongest head-to-tail accuracy while maintaining competitive UQ; generative fine-tuning delivers the most reliable UQ; large reasoning models improve performance on rare labels yet exhibit surprisingly weak UQ; and self-verbalized confidence is not a reliable proxy for uncertainty. Our work is publicly available at https://hhi.fraunhofer.de/aml-demonstrator/made-benchmark.

1 Introduction

Strong predictive performance is not sufficient for the adoption of machine learning (ML) models in high-stakes settings such as healthcare (Lekadir et al., 2025; Reddy et al., 2021) where human oversight is paramount (Shneiderman, 2020). In this regard, uncertainty quantification (UQ) is essential for reliable ML systems (Ojha et al., 2025) because it can flag doubtful or ambiguous cases for human re-examination.

Multi-label text classification (MLTC) is central to patient categorization, clinical coding, and incident reporting, etc. Developing reliable MLTC systems presents practitioners with several challenges: MLTC requires selecting multiple labels from a typically much larger set, which leads to a combinatorial problem that scales exponentially with the label space size. Real-world MLTC data can be characterized by severe inter- and intra-class imbalances: a few common conditions comprise the majority of examples, while safety-critical conditions reside in the long tail. Models must learn to disentangle correlated signatures without becoming biased toward frequent classes. Further, labels often co-occur and are hierarchically interdependent, violating the assumption of label independence.

Crucially, existing MLTC benchmarks are increasingly ill-suited for evaluating frontier large language models (LLMs). Traditional datasets are static, can be saturated, suffer from data contamination due to their inclusion in LLM pre-training corpora, or lack the label imbalance and interdependence of real-world environments.

These challenges leave practitioners with open questions: Which model architecture is most suitable for this complexity? Can a specialized, small model solve the task, or is a larger foundation model required? Which learning paradigm (fine-tuning vs. in-context learning) yields the best trade-off between performance for frequent and rare classes? How reliable are the resulting predictions? We address these questions by introducing an uncontaminated benchmark and systematically evaluating models across architectures, learning paradigms, and UQ methods.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 46308–46328

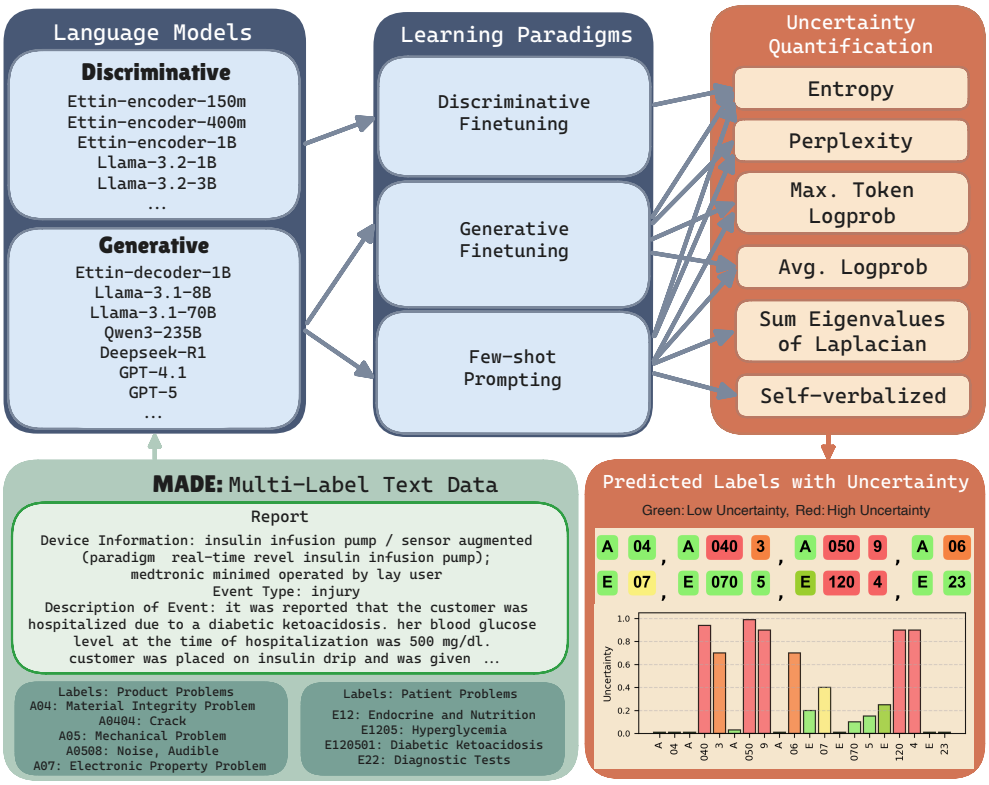

Figure 1: *Top:* Overview of the benchmarking setup, encompassing discriminative and generative language models, learning paradigms (discriminative or generative fine-tuning and few-shot prompting), and uncertainty quantification (UQ) approaches. *Bottom, left:* Multi-label text classification of medical device adverse events, each annotated with hierarchical product and patient problem labels. *Bottom, right:* UQ quality is evaluated (for of each model, learning paradigm and UQ method).

1.1 Related work

Multi-label text classification. Early approaches to MLTC relied on models like Naïve Bayes, support vector machines (Wang and Manning, 2012), or recurrent neural networks. With the advent of transformer-based language models (Vaswani et al., 2017), encoder-only (Huang et al., 2021) and decoder-only (Ma et al., 2025; Galke et al., 2025) architectures served for this task. Controlled comparisons between generative and discriminative fine-tuning for decoder models and head-to-head comparisons between encoder- and decoder-only models under matched conditions are rare. Weller et al. (2025) addresses this with a suite of matched encoder- and decoder-only models ('Ettin') trained on the ModernBERT architecture (Warner et al., 2024) across model sizes, but do not evaluate on MLTC, leaving this comparison open. Decoder-only LLMs support zero-shot (inference without labeled data) and few-shot prompting (in-context learning; conditioning on a few labeled exemplars without updating model weights), which has been compared with fine-tuning on a variety of tasks where medical expertise is required, with differing results (e.g., Nori et al., 2023; Maharjan et al., 2024; Labrak et al., 2024). Across broader benchmarks, the advantage of fine-tuning vs. prompting varies by task, data regime, and model (Chen et al., 2025; Wu et al., 2025). For MLTC, performance is sensitive to e.g. label semantics, output constraints, and thresholding strategies, motivating controlled comparisons.

Benchmarking datasets for multi-label text classification span diverse domains, including scientific literature (Kowsari et al., 2017, Yang et al., 2018, Fallah et al., 2022, Chen et al., 2022; Schopf et al., 2023), newswire (Lewis, 1997), finance-related user posts (Maia et al., 2021), patents (Tang et al., 2020), clinical notes for ICD coding (Johnson et al., 2016; Mullenbach et al., 2018), legislative documents (Steinberger et al., 2012, Boella et al., 2013, Chalkidis et al., 2019, Bocchi et al., 2024), and more. Overlap between benchmark content and LLM pre-training corpora raises concerns about contamination that can inflate zero-/few-shot performance (Jacovi et al., 2023; Oren et al., 2024; Zhu et al., 2024; Li et al., 2024; Deng et al., 2024; Xu et al., 2024). Heavy reuse can lead to benchmark saturation via overfitting and implicit adapta-

tion, which motivates the continuous introduction of novel datasets to assess generalization. Typical benchmarks are scraped from online repositories and rely on distant or weak supervision (e.g., tags/metadata as labels), often with limited documentation, introducing label noise and hampering reproducibility. Multiple sources, pre-processing choices, copyright limitations, and dataset versions further complicate fair comparisons.

Uncertainty quantification (UQ). For discriminative models, entropy-based UQ measures are well understood (Sensoy et al., 2021; Mucsányi et al., 2024; Baur et al., 2026). In contrast, UQ for generative models is more complex: token-level probabilities, consistency across stochastic generations, and self-verbalized confidence introduce distinct challenges. However, estimating uncertainty in LLM benchmarks is of great interest (Bean et al., 2025) and recent works (e.g., Ye et al., 2024; Vashurin et al., 2025a) cover different natural language processing tasks, but not yet MLTC. Token- or information-based uncertainty metrics, which in various forms rely on the log-probabilities of an LLM's outputs, have been surveyed and formalized by Shorinwa et al., 2025; Fomicheva et al., 2020. Among those of interest are: entropy of the top-n log-probabilities, perplexity, average or maximum token log-probability. Consistency-based UQ metrics measure the output variability of a model under stochasticity, and have been widely used for LLMs as a way to capture epistemic uncertainty (Xiao et al., 2025). Lin et al., 2024 suggested an effective approach to derive a consistency-based uncertainty score through the sum of eigenvalues of a graph Laplacian. Vashurin et al., 2025b proposed the combination of information- and consistency-based metrics as an effective strategy to improve uncertainty estimation, motivating our approach of integrating token-level uncertainty with consistency measures. Self-verbalized uncertainty, in which a model outputs a confidence score corresponding to its prediction (Kadavath et al., 2022; Tian et al., 2023; Harsha Tanneru et al., 2024), offers a complementary approach to directly capture model-reported uncertainty, especially when log-probabilities are not available. Dong et al., 2025 review confidence calibration for imbalanced data.

We focus on information-based and consistency-based metrics due to their complementary strengths, and additionally test self-verbalized uncertainty given its practical appeal when log-probabilities are unavailable.

1.2 Our contributions

In this paper, we (1) **create a benchmark:** To counteract saturated benchmarks with potential pre-training exposure, we introduce a reproducible pipeline to generate a challenging evaluation suite based on medical device adverse event reports made available by the U.S. Food and Drug Administration (FDA). We release `MADE`, characterized by a long-tailed distribution of 1,154 interdependent labels across three hierarchical levels, specifically curated to test generalization limits. We dub this MLTC dataset as a 'living benchmark' because the continuous publication of new reports by the FDA will continue to enable testing of models on fresh data, avoiding potential leakage of test data into the pretraining corpora of future foundation models. (2) **Establish solid baselines:** We fine-tune more than twenty encoder- and decoder-only models in discriminative and generative settings and compare them with few-shot prompting of local and API-accessible models (including reasoning variants). (3) **Evaluate UQ capabilities:** We systematically study multiple UQ approaches (see Figure 1) — including information-based, consistency-based, and self-verbalized uncertainty—and assess their utility for sample routing or human-in-the-loop triage. Beyond providing practical guidance on model selection and UQ strategies for MLTC, our results expose critical reliability trade-offs. By releasing `MADE` alongside comprehensive baselines, we share a foundation for future research into reliable MLTC, enabling the community to evaluate the limits of increasingly capable language models.

2 Data: Medical Device Adverse Events

The FDA regularly publishes medical device adverse event reports[1] along with annotations that capture the associated product and patient problems. From this resource, we curate a large-scale text classification dataset with hierarchical multi-labels. Statistics of the dataset, hierarchical labels, and topics reported are provided in Table 1, Figure 2, and Figure A.1 (in Appendix A.6). We collect files from 2015 (previous reports lack the necessary product problem labels) until mid-2025. From each report, we extract the event description, relevant metadata (event type and device information), and the product and patient problem labels.

While FDA coders assign the IMDRF labels, the consistency of this annotation process has not been

[1] https://open.fda.gov

Total number of samples	488,273
Training set *(2015–2023)*	298,825
Validation set *(1–6/2024)*	71,271
Test set *(7/2024 – 6/2025)*	118,177
Truncated test set	10,288
Average tokens (cl100k_base)	∼370
Average labels per sample	8.79
Unique labels	1,154
Hierarchy levels of labels	3
Minimum occurrences per label	5

Table 1: Summary statistics of MADE.

formally assessed through inter-annotator agreement studies. Labels may therefore reflect annotator variability or systematic biases in the reporting pipeline. As operational annotations, they may not meet the standards of clinically validated ground truth.

We map these labels from FDA terms for each product and patient problem (see Figure 2) to hierarchical codes provided by the International Medical Device Regulatory Forum (IMDRF, 2025). Samples missing IMDRF codes for one or more terms are excluded. To leverage the hierarchy of the IMDRF terminology, each code is up-propagated to include all ancestor codes, as determined from official IMDRF annexes. This approach recognizes that FDA annotators typically use the most specific (leaf) codes and ensures that models are not penalized for predicting valid parent codes. Label mapping and propagation yield two sets of hierarchical labels per report, which are flattened into a union of target labels. To avoid introducing unseen labels into the test set, we freeze the label taxonomy at December 2023, and discard labels introduced thereafter. We remove extremely rare labels (<5 instances), yielding a final set of 1,154 labels following a long-tail distribution. (see Figure 3).

We deduplicate reports based on the event description, retaining only the first occurrence, thus reducing the dataset to 5.5 million samples, and further downsample by applying HDBSCAN (McInnes et al., 2017) on embeddings of event descriptions, selecting cluster representatives, and retaining a high proportion of rarer labels; all events labeled death are included. Data from 2015 to 2023 serve for training, the first half of 2024 for testing, and July 2024 to June 2025 for testing. We compare these periods with the models' knowledge cutoff dates in Appendix A.2. To limit inference costs while maintaining statistical validity, we create a representative 'truncated test set' via stratified sampling (cf. Table 1) and perform all evaluations on this split. In summary, our pipeline transforms raw FDA adverse event reports into an IMDRF-compliant, hierarchically-labeled, temporally-split dataset tailored for a challenging MLTC benchmark. Because the FDA releases new reports quarterly, researchers can evaluate future models on data guaranteed to post-date their training, providing ongoing contamination-free evaluation rather than a one-time static benchmark.

See Appendix A.1 for data availability.

3 Experimental setup

3.1 Fine-tuning and in-context learning

Discriminative training. We fine-tune Llama 3.2-1B/-3B, and 3.1-8B base models (Grattafiori et al., 2024), with a classification head in the final layer. We also fine-tune Ettin models (150M, 400M, 1B; Weller et al., 2025), which are based on the ModernBERT (Warner et al., 2024) architecture.

We experiment with a hierarchical loss where the label space is partitioned by taxonomy level and a separate binary cross-entropy loss is computed at each level, with the total loss being their unweighted sum. Details are reported in the Appendix A.6.

We use AdamW with cosine scheduler, batch size 512, 20 epochs, warmup ratio 0.1, context length 512, and fixed maximum gradient norm 0.01. Learning rate is tuned in [2e-4, 5e-4]. Per-label classification thresholds are selected to maximize F1 on the validation set independently for each label.

Generative training. We fine-tune Llama 3.2-1B, 3.2-3B, 3.1-8B and 3.1-70B, and Ettin decoder variants (400M, 1B) to generate class labels as tokens. Each model is trained for 4 epochs. An improvement for a larger number of epochs was not observed. We compare full fine-tuning with LoRA (Hu et al., 2022) for selected models.

We use AdamW (Loshchilov and Hutter, 2019) with cosine learning-rate decay (Loshchilov and Hutter, 2016), warmup ratio 0.1, batch size 64, context length 512, and maximum generation length 50 tokens. Learning rate (3e-5 to 2e-4) and maximum gradient norm (1.0–10.0) are tuned on the validation set. For LoRA experiments, we use rank 16–64 and alpha 32–64, targeting all attention (Q, K, V, O) and MLP (gate, up, down) projection layers.

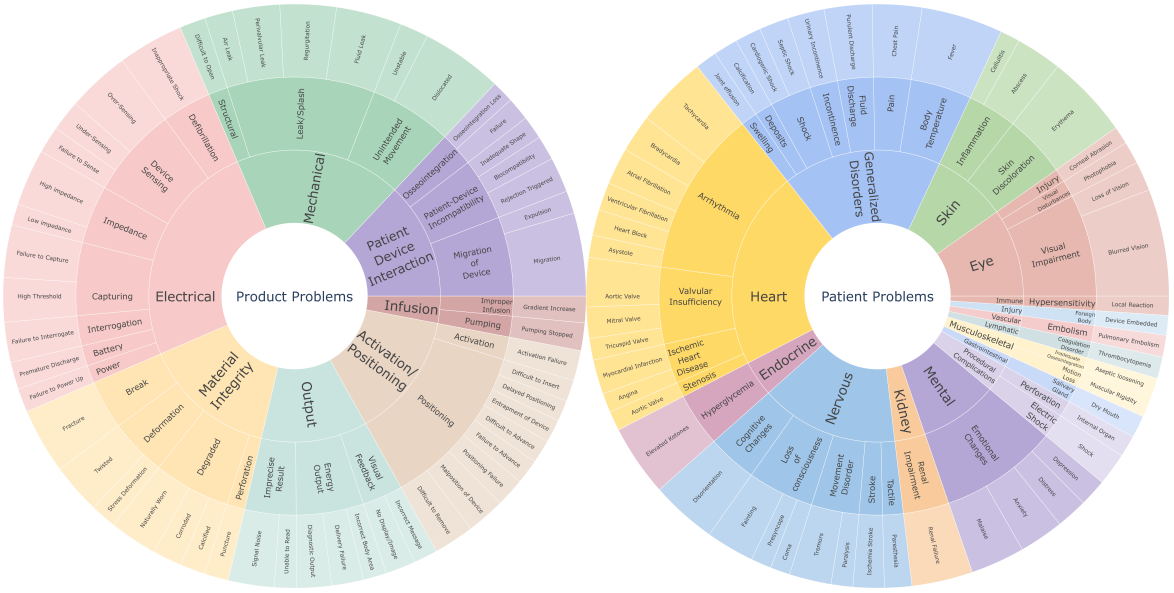

Figure 2: Product and patient problems are the hierarchical multi-labels of MADE. The outer ring shows the fifty most frequent product or patient problems in the test set, grouped by their parent classes (middle ring) and grandparent classes (inner ring).

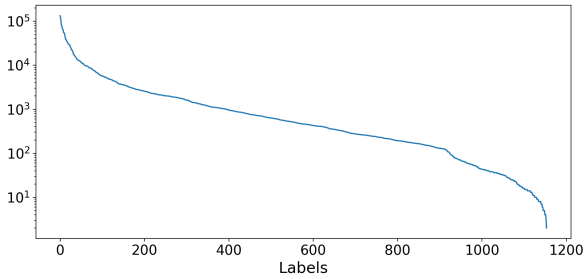

Figure 3: Log-scaled distribution of label frequencies illustrating severe imbalance and the pronounced long-tail pattern.

Few-shot prompting/in-context learning. We locally host Llama 3.2-3B, 3.1-8B, and 3.1-70B, DeepSeek-R1 (37B active, 671B total; DeepSeek-AI et al., 2025), Qwen3 (4B, 30B, 235B; Yang et al., 2025), Kimi K2 (32B activated; 1T total; Kimi-Team et al., 2025), gpt-oss-120b (5.1B active, 117B total; OpenAI et al., 2025), GLM-4.5-Air (12B active, 106B total; GLM-4.5-Team et al., 2025), and Llama-3.3-Nemotron-49B-v1.5 (Bercovich et al., 2025). GPT-4.1 (OpenAI, 2025b) and GPT-5 (OpenAI, 2025a) are accessed via API. Among API-accessed models, only GPT-4.1 – but not GPT-5 – provides log probabilities, which are required for our UQ investigations. GPT-5 is run with medium reasoning effort and does not allow modifying temperature. Knowledge cutoff dates for all the models are discussed in Appendix A.2. The prompt includes task instructions, the label list, and ten kNN-retrieved training examples (see Appendix A.7). A pre-trained model is used to embed the reports for retrieval ('bioclinical-modernbert-base-embeddings', NeuML, 2025). Each retrieved example is truncated to 10000 characters (about 2300 tokens). Unless otherwise stated, all outputs are generated in a greedy decoding setup ($temperature = 0, top\_p = 1, top\_k = -1$).

Software. All experiments use PyTorch 2.7.1, Transformers 4.56.0, and vLLM 0.10.1.

Hardware. Details about the computing infrastructure are reported in Appendix A.3.

3.2 Methods for uncertainty quantification

For discriminative models, uncertainty is computed from the output probability distributions with per-label entropy across classes. For generative models, we quantify uncertainty with two complementary approaches: (i) information-level uncertainty U_{info} is measured with token-based metrics (entropy, improbability, avg-log-probability, perplexity), while (ii) consistency-based uncertainty U_{cons} measures variation across multiple stochastic forward passes via the sum of eigenvalues of the graph Laplacian (Lin et al., 2024). Multiplying U_{info} and U_{cons} yields the combined uncertainty U_{combined}. We obtain self-verbalized uncertainty U_{self} by prompting the model to express its own confidence. Appendix A.4 contains full metric definitions and details.

3.3 Evaluation of uncertainty scoring

We assess UQ quality using selective prediction. Prediction Rejection Rate (PRR, Malinin and Gales, 2021; Vashurin et al., 2025b) quantifies the effectiveness with which uncertainty scores

identify unreliable predictions. Model outputs are ranked according to the uncertainty scoring. Predictions with the highest uncertainty are progressively removed, starting from the most uncertain. The improvement of the overall correctness due to this rejection is measured with the Jaccard score. Performance gains (relative to a random ranking) are compared against an oracle ranking, which represents the theoretical upper bound. Figure 5 illustrates the intuition of PRR and how it is derived. As complementary measure, we compute the Spearman correlation (ρ) between uncertainty scores and per-sample correctness. A negative correlation indicates that a higher uncertainty aligns with a lower correctness, which is desirable. A positive correlation suggests that incorrect predictions often occur with high confidence, which is unfavorable.

Finally, we evaluate calibration using positive-class expected calibration error (ECE_+), a variant of expected calibration error (ECE, Lichtenstein et al., 1977; Dawid, 1985) computed only on positive instances. Standard ECE is unreliable in our setting due to severe label imbalance and the multi-label structure of our data: for many medium, tail, and extreme-tail classes, it yields values below 1% despite frequent missed predictions. This issue is exacerbated for LLM-based models, where log-probabilities are available only for positive predictions, requiring zero confidence to be assumed for negative predictions and artificially inflating calibration scores for rare labels. Conditioning ECE on class labels has been proposed to address shortcomings of ECE (Nixon et al., 2019). Therefore, we report ECE_+ (Appendix A.5), which can be interpreted as a measure of under-confidence.

4 Results and Discussion

We evaluate predictive performance (macro F1, Jaccard J) and uncertainty quantification (PRR, Spearman ρ, ECE_+) across four paradigms: (i) discriminative fine-tuning of encoders/decoders, (ii) generative decoder fine-tuning, (iii) kNN-based ten-shot prompting with instruction-tuned or (iv) thinking/reasoning models. Labels are grouped by training-set frequency: head (>1%), medium (0.1–1%), tail (0.01–0.1%), and extreme tail (<0.01%). Figure 4 summarizes all results per paradigm with individual metrics shown in Table 2.

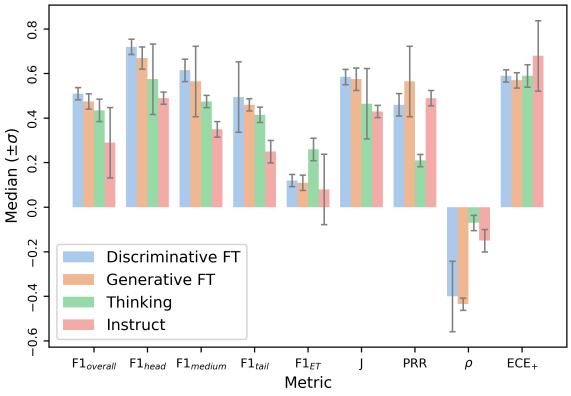

Figure 4: Overview of predictive performance and uncertainty quantification results per paradigm.

4.1 Multi-label predictive performance

Fine-tuning. Llama-3.1-8B-Base yields the highest macro F1 overall (0.54) and leads in head (0.74), medium (0.64), and tail (0.53) classes, when discriminatively fine-tuned. Smaller LLaMA models (Llama-3.2-3B/-1B) and encoders (Ettin-1B/400m/150m) trail slightly, consistent with their reduced capacity, but Ettin-1B-Encoder remains competitive given its size. In a generative setting, the much larger Llama-3.1-70B-Base achieves similar results. Notably, discriminative fine-tuning outperforms generative fine-tuning of similarly-sized decoders (Wilcoxon signed-rank test, $n = 5$ matched pairs; $p \leq 0.05$). Ablations of full vs. parameter-efficient fine-tuning and of instruction-tuned vs. base models are shown in Appendix A.6.

Prompting. For ten-shot in-context learning, Qwen3-235B-A22B-Instruct achieves the strongest macro F1 (overall 0.44, extreme tail 0.24), with only GPT-4.1 matching performance on tail classes. Reasoning models excel on the extreme tail (ET): GPT-5 attains the highest ET macro F1 (0.34) and ties the best overall macro F1 (0.54) with Llama-3.1-8B-Base (discriminative). Among open-weight reasoning models, Qwen3-235B-A22B-Thinking is strongest (overall 0.49, tail 0.48, ET 0.33). Gain on rare labels for all prompt-based models comes with a trade-off: head-class performance is consistently below that of the best fine-tuned decoders.

Overall. Discriminative decoder fine-tuning (Llama-3.1-8B-Base) remains the best for head–tail accuracy and consistently outperforms generative training. Prompt-based reasoning models (Qwen3-235B, DeepSeek-R1, GPT-5) dominate ET classes and can match overall F1 but lag on head classes (Mann-Whitney U test, independent groups: $n = 9$

Paradigm/model	**Macro F1** ↑					**J** ↑	**PRR** ↑	ρ ↓	**ECE$_+$** ↓
Number of classes →	Overall *1154*	Head *144*	Medium *481*	Tail *348*	ET *181*				
Discriminative fine-tuning	**0.51**±.03	**0.72**±.02	**0.62**±.03	**0.5**±.03	0.12±.02	**0.59**±.02	0.46±.05	-0.40±.04	0.59±.03
Llama-3.1-8B-Base	**0.54**	**0.74**	**0.64**	**0.53**	0.12	**0.62**	0.47	-0.40	0.58
Llama-3.2-3B-Base	0.51	0.72	0.62	0.49	0.11	0.59	0.46	-0.41	0.59
Llama-3.2-1B-Base	0.51	0.71	0.60	0.48	**0.14**	0.58	**0.52**	**-0.42**	0.60
Ettin-1B-Encoder	0.53	0.73	0.63	0.51	0.13	0.61	0.46	-0.40	**0.56**
Ettin-400m-Encoder	0.51	0.72	0.61	0.50	0.12	0.58	0.44	-0.36	0.59
Ettin-150m-Encoder	0.46	0.68	0.56	0.44	0.07	0.55	0.38	-0.30	0.64
Generative fine-tuning	0.48±.04	0.67±.03	0.57±.04	0.46±.04	0.11±.03	0.58±.06	**0.57**±.03	**-0.44**±.08	**0.57**±.04
Llama-3.1-70B-Base	**0.53**	**0.73**	**0.62**	**0.51**	**0.16**	**0.61**	0.55	-0.27	**0.49**
Llama-3.1-8B-Base	0.50	0.70	0.59	0.48	0.12	0.59	**0.63**	-0.30	0.52
Llama-3.2-3B-Base	0.48	0.67	0.57	0.46	0.12	0.58	0.60	**-0.46**	0.57
Llama-3.2-1B-Base	0.43	0.63	0.52	0.39	0.10	0.45	0.54	-0.44	0.60
Ettin-1B-Decoder	0.47	0.67	0.56	0.46	0.10	0.57	0.56	-0.43	0.57
Ettin-400m-Decoder	0.44	0.66	0.54	0.42	0.07	0.54	0.57	-0.44	0.60
Prompting – instruct	0.29±.15	0.49±.16	0.35±.16	0.25±.16	0.08±.09	0.43±.17	0.49±.15	-0.15±.23	0.68±.14
Llama-3.1-70B-Instruct	0.30	0.50	0.35	0.25	0.08	0.43	**0.60**	-0.15	0.68
Llama-3.1-8B-Instruct	0.08	0.28	0.09	0.03	0.01	0.22	0.20	0.26	0.78
Qwen3-235B-A22B-Instruct	**0.44**	**0.60**	**0.48**	**0.42**	**0.24**	0.49	0.56	**-0.34**	**0.56**
Qwen3-30B-A3B-Instruct	0.22	0.48	0.27	0.14	0.05	0.43	0.54	0.05	0.59
Qwen3-4B-Instruct	0.29	0.49	0.35	0.25	0.09	0.41	0.49	-0.27	0.68
Kimi-K2-Instruct	0.09	0.18	0.11	0.06	0.01	0.07	0.28	0.08	0.97
GPT-4.1	0.43	0.59	0.47	**0.42**	0.22	**0.57**	0.45	-0.31	0.60
Prompting – thinking	0.44±.05	0.58±.05	0.48±.05	0.42±.06	**0.26**±.07	0.47±.04	0.21±.10	-0.07±.04	0.59±.07
Llama-3.3-Nem.-49B-v1.5	0.42	0.57	0.46	0.38	0.19	0.46	0.21	-0.03	0.59
Qwen3-235B-A22B-Think.	0.49	0.62	0.52	0.48	0.33	0.48	**0.34**	**-0.09**	**0.45**
Qwen3-30B-A3B-Think.	0.45	0.58	0.49	0.44	0.28	0.47	0.08	-0.07	0.56
Qwen3-4B-Thinking	0.38	0.53	0.42	0.36	0.2	0.43	0.21	-0.02	0.63
DeepSeek-R1-0528	0.48	0.62	0.51	0.47	0.30	0.50	0.24	**-0.09**	0.50
GLM-4.5-Air	0.42	0.56	0.46	0.39	0.24	0.44	0.24	**-0.09**	0.62
gpt-oss-120b	0.40	0.57	0.45	0.38	0.15	0.45	0.05	0.00	0.63
GPT-5	**0.54**	**0.68**	**0.58**	**0.53**	**0.34**	**0.57**	NA	NA	NA

Table 2: Predictive performance (macro F1, J) and UQ (PRR, ρ, ECE$_+$) per learning paradigm and model. Median results ($\pm\sigma$) are listed per paradigm, see also Figure 4). Macro F1 is reported for head, medium, tail, and extreme-tail (ET) classes. For discriminative fine-tuning, per-label thresholds were selected on the validation set. For generative models, PRR corresponds to the respective best U_{info} metric (see Section 3.3). Bold marks the best model within each paradigm; underlining indicates the overall best. Arrows show whether higher or lower values are preferable. Evaluation used the truncated test set (n = 10,288). Llama-3.1-8B-Base (discriminative) delivers the strongest predictive performance next to GPT-5, while Llama-3.2-3B-Base (generative) achieves the best UQ with respect to PRR and ρ, and Llama-3.2-1B-Base (discriminative) obtains the best ECE$_+$.

U_{info} **metric**	**Gen. FT**	**Instruct**	**Thinking**
Avg. Log-Prob.	**0.54** ± 0.05	0.37±0.25	0.18 ± 0.12
Entropy	**0.58** ± 0.03	0.45 ±0.15	0.19 ± 0.12
Improbability	**0.54** ± 0.05	0.43 ± 0.15	0.17 ± 0.12
Max. Log-Prob.	**0.52** ± 0.08	0.41 ± 0.16	0.18 ± 0.12
Perplexity	**0.54** ± 0.06	0.37 ± 0.25	0.18 ± 0.11

Table 3: Average PRR ($\pm\sigma$) across U_{info} metrics for each generative paradigm. Entropy-derived U_{info} performs strongest for finetuned and instruct paradigms. For thinking models, U_{info} choice has only a minor influence on PRR.

vs. 12; $p \leq 0.05$). Within prompting, reasoning variants consistently outperform their instruct counterparts (see Qwen3).

4.2 Uncertainty quantification

In practice, reliable uncertainty estimates are what allow NLP systems to be safely deployable: a model that can signal when it does not know enables downstream pipelines to defer low-confidence predictions to human review rather than acting on them automatically, which can be more valuable than marginal accuracy gains alone.

Fine-tuning. For discriminative fine-tuning, Llama-3.2-1B-Base achieves the highest PRR (0.52) and best ρ, but ranks lower on ECE$_+$, trailing the best model in this group (Ettin-1B) by 0.04. Within the Ettin family, performance scales with model capacity, with Ettin-150M performing worst and Ettin-1B best across all metrics. This trend

Model	Method	PRR:	U_{info}	U_{cons}	U_{combined}
Llama-3.1-70B-Instruct	Avg. Log-Prob.		0.59	-	0.60
	Entropy		**<u>0.60</u>**	-	**<u>0.61</u>**
	Improbability		0.58	-	**<u>0.61</u>**
	Max. Log-Prob.		0.57	-	0.60
	Perplexity		0.58	-	0.60
	Σ EigV. Laplacian		-	0.57	-
	Self-Verbalized		0.00	-	0.36
Qwen3-235B-A22B-Instruct	Avg. Log-Prob.		0.53	-	0.54
	Entropy		**0.54**	-	**0.57**
	Improbability		0.53	-	0.54
	Max. Log-Prob.		0.51	-	0.53
	Perplexity		0.53	-	0.56
	Σ EigV. Laplacian		-	0.51	-
	Self-Verbalized		0.01	-	0.28

Table 4: PRR of information-/consistency-based, and combined UQ methods for 2 top-performing (non-thinking) generative models. Best PRR in bold, best overall underlined. U_{cons} is computed from 5 samples per prompt at temperature 1.

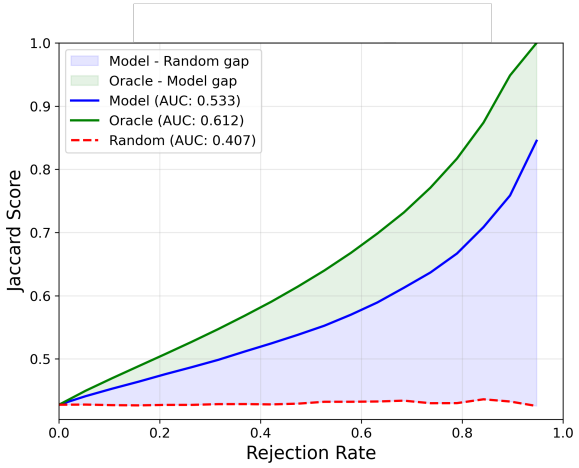

Figure 5: Illustration of the PRR for UQ evaluation: Rejection curves are shown for Llama-3.1-70B-Instruct under three strategies – uncertainty-based (blue, most uncertain predictions are rejected first), oracle (green, best case, discarding predictions achieving lowest Jaccard scores first), and random (red, baseline). PRR is the ratio of the model surplus to the oracle surplus: area where the uncertainty-based curve exceeds random divided by the area where the oracle exceeds random.

does not hold for the Llama family, where the smallest model (1B) achieves the best PRR (0.52) and ρ (-0.42). In the generative fine-tuning setting, Llama-3.1-8B-Base attains the strongest PRR (0.63)—the highest across all paradigms—and ranks second on ρ. A PRR of 0.63 means that by deferring the least confident predictions, the system can eliminate a disproportionate share of errors, retaining high-precision outputs for automated processing while routing uncertain cases to human review. While still moderately underconfident, it places second on ECE_+, only 0.03 behind Llama-3.1-70B, which achieves the best ECE_+ (0.49) in this group. Notably, generative fine-tuning yields the best median PRR, ρ, and ECE_+ across all paradigms.

Prompting. Within instruct prompting, Llama-3.1-70B achieves the strongest PRR, while Qwen3-235B-Instruct attains the best ρ and ECE_+. Performance varies strongly across models in this paradigm (σ of 0.15 for PRR and 0.23 for ρ), with some severely underperforming (e.g., Llama-3.1-8B with PRR 0.20, Kimi-K2 with PRR 0.28). Models with low PRR often exhibit higher ρ, consistent with expectations. ECE_+ also spans a wide range, from severe underconfidence (0.97, Kimi-K2-Instruct) to substantially better calibration (0.56, Qwen3-235B-Instruct). For prompted thinking models, PRR and ρ degrade substantially, with median PRR around 0.23–roughly half of fine-tuned or instruct-prompted models. Several models perform only marginally above chance (PRR 0.05–0.08), and the strongest reaches 0.34 (Qwen3-235B-Thinking). Poor alignment between uncertainty and correctness is indicated by ρ near zero or slightly negative. For applications where abstention decisions carry real costs—such as routing a case to a domain expert—this near-random uncertainty ordering makes thinking models unsuitable without further calibration, as their confidence scores provide little actionable signal for setting abstention thresholds. In contrast, this paradigm yields the best overall calibration, with an ECE_+ of 0.45 (Qwen3-235B-Thinking), outperforming instruct prompting and approaching fine-tuned models.

Overall. Generative fine-tuned models achieve the strongest UQ performance, with the best median PRR as well as ρ (Mann-Whitney U test; Gen. FT group vs. all others combined; $p \leq 0.05$) and ECE_+, and the lowest PRR variability ($\sigma \pm 0.03$). Prompted instruct models rank second in PRR but show substantial variability across models, with several failing to provide reliable uncertainty estimates. Discriminative fine-tuning performs slightly worse in PRR but yields more stable and consistent results (Levene's test, comparing the variance of PRR; $p \leq 0.05$). Across all paradigms, models are markedly underconfident; even the best global ECE_+ of 0.45 indicates substantial underconfidence. Systematic underconfidence of this magnitude means that users relying on verbalized confidence scores would consistently perceive the model as less certain than it is, potentially over-triggering manual review and reducing the efficiency gains that selective prediction is intended

to provide. Detailed ECE_+ results across label categories are shown in A.4. In contrast to fine-tuned models, thinking models fail to meaningfully quantify uncertainty in our setup. Overall, fine-tuned models—generative or discriminative—in our findings are the most reliable choice for practitioners building systems that rely on selective prediction, where the model must decide when to act autonomously versus escalate to a human reviewer. Prompted instruct models can be effective with careful model selection, while current thinking models appear unreliable and require further study. Substantial headroom for improving both selective prediction performance and calibration remains across all paradigms.

Comparison of Uncertainty Methods.

As discussed in Section 3.2, uncertainty for generative models can be estimated using token-based scores (U_{info}), consistency-based scores (U_{cons}), self-verbalized uncertainty, and their combination (U_{combined}). Table 3 reports average PRR and standard deviation across paradigms for different U_{info} variants. Across all settings, entropy over the predicted token and its alternatives performs best. While differences among U_{info} variants are moderate, we recommend entropy-based U_{info} due to its strong theoretical grounding and semantic comparability to uncertainty estimates in discriminative models. Table 4 compares U_{info}, U_{cons}, self-verbalized uncertainty, and their combinations for Llama-3.1-70B-Instruct and Qwen-235B-A22B-Instruct. U_{cons} provides reasonable uncertainty estimates but underperforms entropy-based U_{info} by 0.03 for both models. Consistent with (Lin et al., 2024), combining U_{info} and U_{cons} yields marginal improvements over entropy-based U_{info} alone, while substantially increasing computational cost due to multiple forward passes. In a deployment context, this cost-benefit tradeoff favors entropy-based U_{info} unless computational resources are abundant and even marginal uncertainty gains translate into meaningful reductions in costly human review. Self-verbalized uncertainty performs poorly for both models and should therefore be avoided.

5 Conclusion

We introduce MADE, a living, contamination-free benchmark for MLTC, and evaluate a wide range of models for predictive performance and UQ capabilities. Our results show that task-specific discriminative models achieve state-of-the-art predictive performance and competitive UQ while remaining efficient and fully controllable. Generative fine-tuning provides modest gains for underrepresented classes but enhances UQ significantly. In contrast, prompted non-thinking models exhibit highly variable predictive and UQ capabilities, highlighting the benefits of fine-tuning. Thinking models show improved prediction on the tail, but consistently fail for UQ. We observe a persistent head–tail performance trade-off across all paradigms. Fine-tuned discriminative and generative models are notably more consistent in both predictive and uncertainty performance (lower variance) than prompted instruct and thinking models. Token-entropy–based UQ emerges as the most effective UQ mechanism, while self-verbalized uncertainty performs poorly. As a living benchmark, MADE will continue to provide contamination-free evaluation as the FDA releases new quarterly reports.

Overall, MLTC remains challenging even for state-of-the-art models, with class imbalance as a primary limiting factor. Unlike established benchmarks where performance has largely plateaued, the best-performing model on our dataset achieves only a 54% Macro F1 score. This gap indicates that the task remains far from solved and that the benchmark provides substantial room for differentiation between models, including future advances and UQ methods. We believe this demonstrates that, while incremental, our dataset constitutes a concrete and useful step toward mitigating benchmark saturation rather than a claim of fully addressing it. Promising research directions for the scientific community include investigating why generative fine-tuning is beneficial for UQ, why UQ fails for thinking models, and training new reasoning models on the dataset using reinforcement learning with verifiable rewards. Further work could explore how UQ and predictive performance on ET classes can be improved for smaller encoder models. Achieving this could position them as a superior choice by combining strong predictive performance with reliable UQ, practical flexibility, and far lower computational costs compared to LLMs.

6 Limitations

UQ for discriminative models can flag both positive and negative predictions where the model is uncertain, allowing human review. In contrast, this is not feasible for LLMs in the current set-

ting as flagging negative predictions would require prompting the model to explicitly enumerate all negative classes, after which UQ could be computed for those classes. Given the large number of classes, such generation would significantly increase computational costs, making it prohibitively expensive–an inherent drawback of LLMs and a relative advantage of discriminative models.

We jointly elicit both the classification and the model's self-verbalized uncertainty within a single prompt (see Appendix A.7.3). An alternative design would decouple these steps, first obtaining the model's prediction and then, in a separate prompt, eliciting its confidence or uncertainty estimate, e.g., via the $P(True)$ protocol of Kadavath et al., 2022. While such a two-stage procedure may yield different or more reliable uncertainty estimates, it also incurs additional computational cost. Systematically comparing single- and multi-prompt elicitation strategies–both in terms of UQ and resource usage–remains an open direction for future work.

Loss functions, predictive performance metrics (e.g. macro F1), and UQ metrics like U_{info} for discriminative models typically assume label independence. However, in the present task, labels exhibit dependencies due to the hierarchy. Prior work on hierarchical loss functions (e.g., Kim et al., 2024) and hierarchical variants of the F1 metric (e.g., Kiritchenko et al., 2006; Plaud et al., 2024; Lin et al., 2025) seeks to address these dependencies. In this study, we provide strong comprehensive baselines using standard metrics, and conduct preliminary experiments with hierarchical losses but more extensive work needs to be done in future work.

To capture variability, we repeat generations for two models, computing U_{cons} from five samples per prompt at temperature 1 (see Table 4). We also report descriptive statistics where feasible (see Tables 2 and 3). However, conducting multiple full training iterations (disc. and gen. finetuning) or inference runs (prompting) to capture performance variability would be computationally prohibitive.

Our analysis relies on a single newly introduced MLTC dataset, which limits the scope of generalization. We acknowledge that a single benchmark can only provide an incremental contribution and that the resulting observations may be specific to the characteristics of this dataset. Consequently, it cannot yield complete answers to the practical questions outlined in the introduction, but should instead be viewed as a starting point for broader, multi-dataset analyses. Future work could extend this approach to further unsaturated datasets to validate the findings across diverse domains. Furthermore, the dataset is restricted to English, which may constrain applicability to multilingual settings.

7 Ethical considerations

We introduce this dataset of adverse event reports primarily as a MLTC benchmark and its intended use is research. The data and associated methods may appear useful in real-world contexts, where models may help draft incident reports (in hospitals or industry) or support surveilling medical devices (in regulatory agencies). Models could automatically label reports and check human-annotated labels. Automation could speed up the prioritization of the large volume of adverse event reports, leading to faster follow-up on defective devices.

However, the reliability of the dataset's reports and labeling is not guaranteed (e.g., they are not extensively validated; see Appendix A.1). There is a possible bias because some of the labels originally come from manufacturers with potential vested interests. Reports appear to be labeled by single annotators, making inter-annotator variability unknown. Original reports have been pre-processed (e.g., partially anonymized) by the FDA. It is unclear how models trained on pre-processed reports behave when faced with actual raw reports in real-world applications. Besides, model deployment would entail risks such as: (1) Automation bias: Overreliance on model outputs, leading to diminished human oversight and possible reductions in expert staff under cost-saving arguments. (2) Missed anomalies: Models may fail to detect outlier reports of less common but severe events, or reports written in unconventional ways. (3) Adversarial incentives: Reporters might (un-/intentionally) phrase reports to influence automated decisions. Given these limitations and risks, we advise against using this dataset or the associated methods for real-world reporting, compliance, or regulatory applications. Instead, the dataset should be considered a benchmarking resource for MLTC research.

8 Acknowledgements

This work was supported by the Senate of Berlin and the European Union's Digital Europe programme under grant agreement No. 101100700 (TEF-Health).

References

Simon Baur, Wojciech Samek, and Jackie Ma. 2026. Benchmarking uncertainty and its disentanglement in multi-label chest X-ray classification. In *Uncertainty for Safe Utilization of Machine Learning in Medical Imaging*, pages 193–203, Cham. Springer Nature Switzerland.

Andrew M. Bean, Ryan Othniel Kearns, Angelika Romanou, Franziska Sofia Hafner, Harry Mayne, Jan Batzner, Negar Foroutan, Chris Schmitz, Karolina Korgul, Hunar Batra, Oishi Deb, Emma Beharry, Cornelius Emde, Thomas Foster, Anna Gausen, María Grandury, Simeng Han, Valentin Hofmann, Lujain Ibrahim, and 23 others. 2025. Measuring what matters: Construct validity in large language model benchmarks. *Preprint*, arXiv:2511.04703.

Akhiad Bercovich, Itay Levy, Izik Golan, Mohammad Dabbah, Ran El-Yaniv, Omri Puny, Ido Galil, Zach Moshe, Tomer Ronen, Najeeb Nabwani, Ido Shahaf, Oren Tropp, Ehud Karpas, Ran Zilberstein, Jiaqi Zeng, Soumye Singhal, Alexander Bukharin, Yian Zhang, Tugrul Konuk, and 117 others. 2025. Llama-Nemotron: Efficient reasoning models. *Preprint*, arXiv:2505.00949.

Lorenzo Bocchi, Camilla Casula, and Alessio Palmero Aprosio. 2024. KEVLAR: The complete resource for EuroVoc classification of legal documents. In *Proceedings of the 10th Italian Conference on Computational Linguistics (CLiC-it 2024)*, pages 66–73, Pisa, Italy. CEUR Workshop Proceedings.

Guido Boella, Luigi Di Caro, Daniele Rispoli, and Livio Robaldo. 2013. A system for classifying multi-label text into EuroVoc. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Law*, ICAIL '13, page 239–240, New York, NY, USA. Association for Computing Machinery.

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.

Qingyu Chen, Alexis Allot, Robert Leaman, Rezarta Islamaj, Jingcheng Du, Li Fang, Kai Wang, Shuo Xu, Yuefu Zhang, Parsa Bagherzadeh, Sabine Bergler, Aakash Bhatnagar, Nidhir Bhavsar, Yung-Chun Chang, Sheng-Jie Lin, Wentai Tang, Hongtong Zhang, Ilija Tavchioski, Senja Pollak, and 20 others. 2022. Multi-label classification for biomedical literature: an overview of the BioCreative VII LitCovid track for COVID-19 literature topic annotations. *Database*, 2022:baac069.

Qingyu Chen, Yan Hu, Xueqing Peng, Qianqian Xie, Qiao Jin, Aidan Gilson, Maxwell B Singer, Xuguang Ai, Po-Ting Lai, Zhizheng Wang, and 1 others. 2025. Benchmarking large language models for biomedical natural language processing applications and recommendations. *Nature communications*, 16(1):3280.

Alexander Philip Dawid. 1985. Calibration-Based Empirical Probability. *The Annals of Statistics*, 13(4):1251 – 1274.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *Preprint*, arXiv:2501.12948.

Chunyuan Deng, Yilun Zhao, Xiangru Tang, Mark Gerstein, and Arman Cohan. 2024. Investigating data contamination in modern benchmarks for large language models. 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 8706–8719, Mexico City, Mexico. Association for Computational Linguistics.

Jinzong Dong, Zhaohui Jiang, Dong Pan, Zhiwen Chen, Qingyi Guan, Hongbin Zhang, Gui Gui, and Weihua Gui. 2025. A survey on confidence calibration of deep learning-based classification models under class imbalance data. *IEEE Transactions on Neural Networks and Learning Systems*, 36(9):15664–15684.

Haytame Fallah, Patrice Bellot, Emmanuel Bruno, and Elisabeth Murisasco. 2022. Adapting transformers for multi-label text classification. In *CIRCLE (Joint Conference of the Information Retrieval Communities in Europe) 2022*.

Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. 2020. Unsupervised quality estimation for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:539–555.

Lukas Galke, Ansgar Scherp, Andor Diera, Fabian Karl, Bao Xin Lin, Bhakti Khera, Tim Meuser, and Tushar Singhal. 2025. Are we really making much progress in text classification? A comparative review. *Preprint*, arXiv:2204.03954.

GLM-4.5-Team, Aohan Zeng, Xin Lv, Qinkai Zheng, Zhenyu Hou, Bin Chen, Chengxing Xie, Cunxiang Wang, Da Yin, Hao Zeng, Jiajie Zhang, Kedong Wang, Lucen Zhong, Mingdao Liu, Rui Lu, Shulin Cao, Xiaohan Zhang, Xuancheng Huang, Yao Wei, and 152 others. 2025. GLM-4.5: Agentic, reasoning, and coding (ARC) foundation models. *Preprint*, arXiv:2508.06471.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh

Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The Llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Sree Harsha Tanneru, Chirag Agarwal, and Himabindu Lakkaraju. 2024. Quantifying uncertainty in natural language explanations of large language models. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pages 1072–1080. PMLR.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Yi Huang, Buse Giledereli, Abdullatif Köksal, Arzucan Özgür, and Elif Ozkirimli. 2021. Balancing methods for multi-label text classification with long-tailed class distribution. In *Proc. EMNLP 2021*, pages 8153–8161, Online and Punta Cana, Dominican Republic. ACL.

IMDRF. 2025. Adverse Event Terminology. `https://www.imdrf.org/working-groups/adverse-event-terminology`. [Accessed 19-11-2025].

Alon Jacovi, Avi Caciularu, Omer Goldman, and Yoav Goldberg. 2023. Stop uploading test data in plain text: Practical strategies for mitigating data contamination by evaluation benchmarks. *Preprint*, arXiv:2305.10160.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. MIMIC-III, a freely accessible critical care database. *Scientific data*, 3(1):1–9.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, and 17 others. 2022. Language models (mostly) know what they know. *Preprint*, arXiv:2207.05221.

Fabian Karl and Ansgar Scherp. 2025. HYDRA: A multi-head encoder-only architecture for hierarchical text classification. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 9292–9303, Suzhou, China. Association for Computational Linguistics.

Gibaeg Kim, SangHun Im, and Heung-Seon Oh. 2024. Hierarchy-aware biased bound margin loss function for hierarchical text classification. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 7672–7682, Bangkok, Thailand. Association for Computational Linguistics.

Kimi-Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, Zhuofu Chen, Jialei Cui, Hao Ding, Mengnan Dong, Angang Du, Chenzhuang Du, Dikang Du, Yulun Du, Yu Fan, and 150 others. 2025. Kimi K2: Open agentic intelligence. *Preprint*, arXiv:2507.20534.

Svetlana Kiritchenko, Stan Matwin, Richard Nock, and A. Fazel Famili. 2006. Learning and evaluation in the presence of class hierarchies: Application to text categorization. In *Advances in Artificial Intelligence*, pages 395–406, Berlin, Heidelberg. Springer Berlin Heidelberg.

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

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. BioMistral: A collection of open-source pretrained large language models for medical domains. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5848–5864, Bangkok, Thailand. Association for Computational Linguistics.

Karim Lekadir, Alejandro F Frangi, Antonio R Porras, Ben Glocker, Celia Cintas, Curtis P Langlotz, Eva Weicken, Folkert W Asselbergs, Fred Prior, Gary S Collins, Georgios Kaissis, Gianna Tsakou, Irène Buvat, Jayashree Kalpathy-Cramer, John Mongan, Julia A Schnabel, Kaisar Kushibar, Katrine Riklund, Kostas Marias, and 30 others. 2025. FUTURE-AI: international consensus guideline for trustworthy and deployable artificial intelligence in healthcare. *BMJ*, 388.

David D. Lewis. 1997. Reuters-21578 text categorization test collection, distribution 1.0. `https://www.daviddlewis.com/resources/testcollections/reuters21578/`. [Accessed 19-11-2025].

Yucheng Li, Frank Guerin, and Chenghua Lin. 2024. LatestEval: Addressing data contamination in language model evaluation through dynamic and time-sensitive test construction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18600–18607.

Sarah Lichtenstein, Baruch Fischhoff, and Lawrence D. Phillips. 1977. Calibration of probabilities: The state of the art. In Helmut Jungermann and Gerard De Zeeuw, editors, *Decision Making and Change in Human Affairs: Proceedings of the Fifth Research Conference on Subjective Probability, Utility, and Decision Making, Darmstadt, 1–4 September, 1975*, pages 275–324. Springer Netherlands, Dordrecht.

Sanne Lin, Flavius Frasincar, and Jasmijn Klinkhamer. 2025. Hierarchical deep learning for multi-label imbalanced text classification of economic literature. *Applied Soft Computing*, 176:113189.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2024. Generating with confidence: Uncertainty quantification for black-box large language models. *Transactions on Machine Learning Research*.

S. Lloyd. 1982. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137.

Ilya Loshchilov and Frank Hutter. 2016. SGDR: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. *Preprint*, arXiv:1711.05101.

Marcus Ma, Georgios Chochlakis, Niyantha Maruthu Pandiyan, Jesse Thomason, and Shrikanth Narayanan. 2025. Large language models do multi-label classification differently. *Preprint*, arXiv:2505.17510.

Jenish Maharjan, Anurag Garikipati, Navan Preet Singh, Leo Cyrus, Mayank Sharma, Madalina Ciobanu, Gina Barnes, Rahul Thapa, Qingqing Mao, and Ritankar Das. 2024. OpenMedLM: prompt engineering can out-perform fine-tuning in medical question-answering with open-source large language models. *Scientific Reports*, 14(1):14156.

Macedo Maia, Juliano Efson Sales, André Freitas, Siegfried Handschuh, and Markus Endres. 2021. A comparative study of deep neural network models on multi-label text classification in finance. In *2021 IEEE 15th International Conference on Semantic Computing (ICSC)*, pages 183–190.

Andrey Malinin and Mark Gales. 2021. Uncertainty estimation in autoregressive structured prediction. In *International Conference on Learning Representations*.

Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11):205.

Leland McInnes, John Healy, and James Melville. 2020. UMAP: Uniform manifold approximation and projection for dimension reduction. *Preprint*, arXiv:1802.03426.

Bálint Mucsányi, Michael Kirchhof, and Seong Joon Oh. 2024. Benchmarking uncertainty disentanglement: Specialized uncertainties for specialized tasks. *Advances in neural information processing systems*, 37:50972–51038.

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 1101–11, New Orleans, LA. Association for Computational Linguistics.

NeuML. 2025. bioclinical-modernbert-base-embeddings. `https://huggingface.co/NeuML/bioclinical-modernbert-base-embeddings`. Hugging Face model repository.

Jeremy Nixon, Michael W Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. 2019. Measuring calibration in deep learning. In *CVPR workshops*, volume 2.

Harsha Nori, Yin Tat Lee, Sheng Zhang, Dean Carignan, Richard Edgar, Nicolo Fusi, Nicholas King, Jonathan Larson, Yuanzhi Li, Weishung Liu, and 1 others. 2023. Can generalist foundation models outcompete special-purpose tuning? Case study in medicine. *arXiv preprint arXiv:2311.16452*.

Jaya Ojha, Oriana Presacan, Pedro G. Lind, Eric Monteiro, and Anis Yazidi. 2025. Navigating uncertainty: A user-perspective survey of trustworthiness of AI in healthcare. *ACM Transactions on Computing for Healthcare*, 6(3):1–32.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, and 21 others. 2025. 2 OLMo 2 furious. *Preprint*, arXiv:2501.00656.

OpenAI, :, Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K. Arora, Yu Bai, Bowen Baker, Haiming Bao, Boaz Barak, Ally Bennett, Tyler Bertao, Nivedita Brett, Eugene Brevdo, Greg Brockman, Sebastien Bubeck, and 108 others. 2025. gpt-oss-120b & gpt-oss-20b model card. *Preprint*, arXiv:2508.10925.

OpenAI. 2025a. GPT-5 system card. `https://openai.com/index/gpt-5-system-card/`. [Accessed 19-11-2025].

OpenAI. 2025b. Introducing GPT-4.1 in the API. `https://openai.com/index/gpt-4-1/`. [Accessed 19-11-2025].

Yonatan Oren, Nicole Meister, Niladri S. Chatterji, Faisal Ladhak, and Tatsunori Hashimoto. 2024. Proving test set contamination in black-box language models. In *The Twelfth International Conference on Learning Representations*.

Roman Plaud, Matthieu Labeau, Antoine Saillenfest, and Thomas Bonald. 2024. Revisiting hierarchical text classification: Inference and metrics. In *Proceedings of the 28th Conference on Computational Natural Language Learning*, pages 231–242, Miami, FL, USA. Association for Computational Linguistics.

Sandeep Reddy, Wendy Rogers, Ville-Petteri Makinen, Enrico Coiera, Pieta Brown, Markus Wenzel, Eva Weicken, Saba Ansari, Piyush Mathur, Aaron Casey, and 1 others. 2021. Evaluation framework to guide implementation of ai systems into healthcare settings. *BMJ health & care informatics*, 28(1):e100444.

Tim Schopf, Karim Arabi, and Florian Matthes. 2023. Exploring the landscape of natural language processing research. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 1034–1045, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Murat Sensoy, Maryam Saleki, Simon Julier, Reyhan Aydogan, and John Reid. 2021. Misclassification risk and uncertainty quantification in deep classifiers. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2484–2492.

Ben Shneiderman. 2020. Human-centered artificial intelligence: Reliable, safe & trustworthy. *International Journal of Human–Computer Interaction*, 36(6):495–504.

Ola Shorinwa, Zhiting Mei, Justin Lidard, Allen Z. Ren, and Anirudha Majumdar. 2025. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *ACM Comput. Surv.*

Ralf Steinberger, Mohamed Ebrahim, and Marco Turchi. 2012. JRC Eurovoc indexer JEX - a freely available multi-label categorisation tool. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 798–805, Istanbul, Turkey. European Language Resources Association (ELRA).

Pingjie Tang, Meng Jiang, Bryan (Ning) Xia, Jed W. Pitera, Jeffrey Welser, and Nitesh V. Chawla. 2020. Multi-label patent categorization with non-local attention-based graph convolutional network. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):9024–9031.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5433–5442, Singapore. Association for Computational Linguistics.

Roman Vashurin, Ekaterina Fadeeva, Artem Vazhentsev, Lyudmila Rvanova, Daniil Vasilev, Akim Tsvigun, Sergey Petrakov, Rui Xing, Abdelrahman Sadallah, Kirill Grishchenkov, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, Maxim Panov, and Artem Shelmanov. 2025a. Benchmarking uncertainty quantification methods for large language models with LM-polygraph. *Transactions of the Association for Computational Linguistics*, 13:220–248.

Roman Vashurin, Maiya Goloburda, Albina Ilina, Aleksandr Rubashevskii, Preslav Nakov, Artem Shelmanov, and Maxim Panov. 2025b. Uncertainty quantification for LLMs through minimum Bayes risk: Bridging confidence and consistency. *Preprint*, arXiv:2502.04964.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Sida Wang and Christopher Manning. 2012. Baselines and bigrams: Simple, good sentiment and topic classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 90–94, Jeju Island, Korea. Association for Computational Linguistics.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *Preprint*, arXiv:2412.13663.

Orion Weller, Kathryn Ricci, Marc Marone, Antoine Chaffin, Dawn Lawrie, and Benjamin Van Durme. 2025. Seq vs Seq: An open suite of paired encoders and decoders. *Preprint*, arXiv:2507.11412.

Jiageng Wu, Bowen Gu, Ren Zhou, Kevin Xie, Doug Snyder, Yixing Jiang, Valentina Carducci, Richard Wyss, Rishi J Desai, Emily Alsentzer, Leo Anthony Celi, Adam Rodman, Sebastian Schneeweiss, Jonathan H. Chen, Santiago Romero-Brufau, Kueiyu Joshua Lin, and Jie Yang. 2025. Bridge: Benchmarking large language models for understanding real-world clinical practice text. *Preprint*, arXiv:2504.19467.

Quan Xiao, Debarun Bhattacharjya, Balaji Ganesan, Radu Marinescu, Katya Mirylenka, Nhan H Pham, Michael Glass, and Junkyu Lee. 2025. The consistency hypothesis in uncertainty quantification for large language models. In *Proceedings of the Forty-first Conference on Uncertainty in Artificial Intelligence*, volume 286 of *Proceedings of Machine Learning Research*, pages 4636–4651. PMLR.

Ruijie Xu, Zengzhi Wang, Run-Ze Fan, and Pengfei Liu. 2024. Benchmarking benchmark leakage in large language models. *Preprint*, arXiv:2404.18824.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

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

Fanghua Ye, Mingming Yang, Jianhui Pang, Longyue Wang, Derek F. Wong, Emine Yilmaz, Shuming Shi, and Zhaopeng Tu. 2024. Benchmarking llms via uncertainty quantification. In *Advances in Neural Information Processing Systems*, volume 37, pages 15356–15385. Curran Associates, Inc.

Wenhong Zhu, Hongkun Hao, Zhiwei He, Yun-Ze Song, Jiao Yueyang, Yumeng Zhang, Hanxu Hu, Yiran Wei, Rui Wang, and Hongyuan Lu. 2024. CLEAN–EVAL: Clean evaluation on contaminated large language models. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 835–847, Mexico City, Mexico. Association for Computational Linguistics.

A Appendix

A.1 Data availability

Origin of the data: We pre-process data that are made available by the U.S. Food and Drug Administration[2] with a 'Creative Commons CC0 1.0 Universal dedication' license[3] under the terms of service expressed by openFDA[4] and the data licensing version from 2014 [5]. The reports' sources may come from manufacturers, user facilities, distributors, and voluntary sources (such as patients and physicians) as mentioned by openFDA[6]. FDA does not endorse this article.

Privacy: It is stated[7] that *'openFDA [...] does not contain data with Personally Identifiable Information about patients or other sensitive information'*. Information about the **'Responsible use of the data'** and a **'Disclaimer'** are given by openFDA. (*'Adverse event reports submitted to FDA do not undergo extensive validation or verification. Therefore, a causal relationship cannot be established between product and reactions listed in a report. While a suspected relationship may exist, it is not medically validated and should not be the sole source of information for clinical decision making or other assumptions about the safety or efficacy of a product. Additionally, it is important to remember that adverse event reports represent a small percentage of total usage numbers of a product. Common products may have a higher number of adverse events due to the higher total number of people using the product. In recent years the FDA has undertaken efforts to increase collection of adverse events. Increases in the total number of adverse events is likely caused by improved reporting.'; 'Although MDRs are a valuable source of information, this passive surveillance system has limitations, including the potential submission of incomplete, inaccurate, untimely, unverified, or biased data. In addition, the incidence or prevalence of an event cannot be determined from this reporting system alone due to potential under-reporting of events and lack of information about frequency of device use. Because of this, MDRs comprise only one of the FDA's several important postmarket surveillance data sources.'*).

Reproducibility: We publish[8] the code repository for data preprocessing and benchmark. We make pre-processed dataset splits available[9].

A.2 Model knowledge cutoff dates

FDA publishes reports of adverse events involving medical devices *after* each quarter through openFDA. Thus, reports from the first quarter (July-September 2024) of our test data period (July 2024 to June 2025; see Section 2) were published in October 2024 or later. Consequently, knowledge cutoff dates for model pre-training until including September 2024 are expected to avoid potential test data leakage (if these data were included in the pre-training dataset of a model). Our test set should contain only data released after the knowledge cutoff dates for Llama 3.1-3 (December 2023), GPT 4.1 (June 2024), gpt-oss-120b (June 2024), and GPT-5 (September 2024). Ettin (Weller et al., 2025) was pre-trained including on the 'DOLMino mix 1124' dataset (OLMo et al., 2025, which mentions data from September 2024), among other data sources.

We have not found official information on the knowledge cutoff dates for DeepSeek-R1, Qwen3, and Kimi K2 (released in January, April, and July 2025, respectively). Llama-3.3-Nemotron-49B-v1.5 used post-training data released in July 2025.

A.3 Computing infrastructure

The computing infrastructure used included two Nvidia A100 (40GB) and up to eight Nvidia H200 and AMD Instinct MI300X accelerators.

A.4 Methods for uncertainty quantification

Uncertainty for discriminative models is quantified from the output probability vector of the MLTC model. We treat the classes as independent and compute binary entropy per class:

$$\mathbb{H}(\pi_c) = -\big[\pi_c \log(\pi_c) + (1-\pi_c)\log(1-\pi_c)\big],$$

where π_c is the predicted probability to be an instance (label 1) of class c. The information-based uncertainty for a sample is the vector of entropies across classes:

$$U_{\text{info}} = \big(\mathbb{H}(\pi_1), \mathbb{H}(\pi_2), \ldots, \mathbb{H}(\pi_C)\big).$$

[2] https://open.fda.gov/data/downloads/ → 'Medical Device Event')

[3] http://creativecommons.org/publicdomain/zero/1.0/legalcode

[4] https://open.fda.gov/terms/

[5] https://open.fda.gov/license/

[6] https://open.fda.gov/apis/device/event/

[7] https://open.fda.gov/apis/

[8] https://github.com/raunak-agarwal/made-benchmark

[9] https://huggingface.co/datasets/ragarwal/MADE-Multilabel-Benchmark

Uncertainty for generative models is quantified with two complementary scores: information-based U_{info} and consistency-based U_{cons}. Multiplying information- and consistency-based scores results in the combined uncertainty score: $U_{\text{combined}} = U_{\text{info}} \cdot U_{\text{cons}}$. We compute U_{info} with these metrics:

- Average Log-Probability of Tokens: $\text{Avg}(\pi) = -\frac{1}{L_i}\sum_j \log(\pi_{ij})$
- Entropy of Token Distribution: $H_{ij} = -\sum_{w \in D} \pi_{ij}(w) \log \pi_{ij}(w)$
- Improbability $= 1 - \prod_j \pi_{ij}$
- Maximum Log-Probability of Tokens: $\text{Max}(\pi) = \max_j (-\log(\pi_{ij}))$
- Perplexity$(\pi) = \exp\left(-\frac{1}{L_i}\sum_j \log(\pi_{ij})\right)$

The consistency-based uncertainty score U_{cons} is computed using the Laplacian of a graph derived from multiple stochastic forward passes (Lin et al., 2024). We perform $n = 5$ stochastic forward passes with temperature $t = 1$, calculate the pairwise Jaccard similarity W between the predictions, and compute the normalized Laplacian L as $L = I - D^{-1/2}WD^{-1/2}$, where I is the identity matrix and D is the degree matrix ($D = \text{diag}\left(\sum_j W_{ij}\right)$). The consistency-based uncertainty score is then $U_{\text{cons}} = \sum_k \max(0, 1-\lambda_k)$, with λ_k denoting the eigenvalues of L.

Finally, we prompt the model to self-verbalize its confidence $C_i \in [0, 1]$ for each prediction. High confidence corresponds to low uncertainty. Self-verbalized uncertainty is, thus, $U_{\text{self}} = 1 - C_i$.

A.5 Evaluation of uncertainty scores

With selective prediction, we evaluate the quality of uncertainty scores (either U_{info} or U_{combined}). Outputs of a model are rejected according to their sorted uncertainty scores. We assess whether these rejections improve the overall predictive performance.

Prediction rejection rate (PRR; illustrated in Figure 5) evaluates the effectiveness of an uncertainty scoring in prioritizing unreliable predictions:

$$\text{PRR} = \frac{\text{AUC}_{\text{uncertainty}} - \text{AUC}_{\text{random}}}{\text{AUC}_{\text{oracle}} - \text{AUC}_{\text{random}}},$$

where $\text{AUC}_{\text{uncertainty}}$ is the area under the curve (AUC) obtained by sorting the samples by uncertainty U (highest first), iteratively rejecting the top $N\%$ of the samples, and computing the average Jaccard score after each rejection threshold. $\text{AUC}_{\text{oracle}}$ is the AUC obtained by sorting samples by the Jaccard score (lowest score first) and iterative rejection – this gives us a clear upper-bound to compare against for a given rejection threshold. $\text{AUC}_{\text{random}}$ is the AUC obtained by rejecting $N\%$ random samples at each step. A PRR close to one indicates that the uncertainty scoring is nearly as good as the oracle, while a PRR near zero indicates that the scoring is not better than random.

Spearman correlation (ρ) of uncertainty with per-sample correctness. We evaluate binary correctness (0/1) for each example–label pair, compute Spearman's ρ between U and correctness separately for each label, and report the average ρ across all labels. A large negative correlation indicates that higher uncertainty coincides with lower prediction accuracy, confirming that U effectively flags unreliable predictions.

Positive class expected calibration error (ECE_+). To assess calibration specifically on the subset of samples where a label is positively present, we compute the positive class expected calibration error. For a given class c, we restrict evaluation to the positive instances of that class and bin their predicted confidences into M bins b_m. Following the formula of standard ECE, the discrepancy between empirical accuracy and predicted confidence is then

$$\text{ECE}_{+c} = \frac{1}{N_c^+}\sum_{m=1}^{M} |b_m| \left|\text{acc}(b_m)^+ - \text{conf}(b_m)^+\right|,$$

where N_c^+ is the number of samples for which class c is a true label, $\text{acc}(b_m)^+$ is the fraction of positives in bin m, and $\text{conf}(b_m)^+$ is the average predicted confidence for class c in that bin. Since we only consider samples where the ground truth label is present, $\text{acc}(b_m)^+ = 1$ for all bins, and the formula simplifies to

$$\text{ECE}_{+c} = \frac{1}{N_c^+}\sum_{m=1}^{M} |b_m| \left(1 - \text{conf}(b_m)^+\right)$$

as accuracy does not vary across bins, this further simplifies to

$$\text{ECE}_{+c} = 1 - \frac{1}{N_c^+}\sum_{i=1}^{N_c^+} p_i^c,$$

with p_i being the predicted probability of class c being present in a sample. Positive class expected

calibration error therefore measures average underconfidence of the model on true positive instances, and semantically captures how well the model's confidence aligns with correctness conditional on the class being present. The total score ECE_+ is derived by averaging over all binary class scores

$$\mathrm{ECE}_+ = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} \mathrm{ECE}_{+c}$$

where $\mathcal{C}$ is the set of all classes.

A.6 Additional results

Figure A.1 gives an overview of the topics reported in the truncated test set of the MADE dataset.

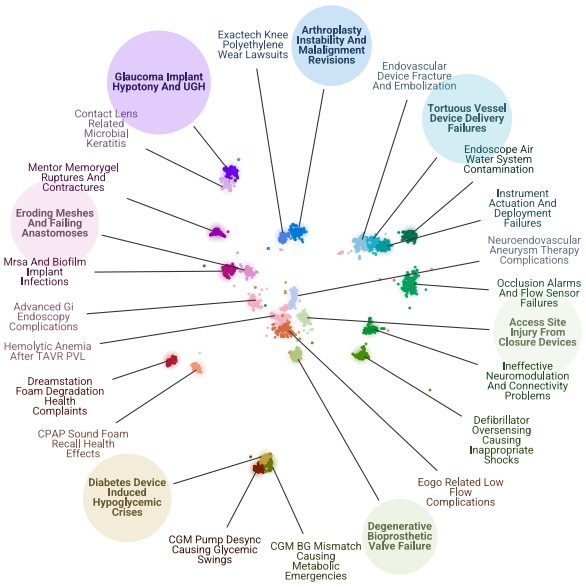

Figure A.1: The 25 most frequent topics in MADE reports (truncated test set) are identified with UMAP (McInnes et al., 2020) and K-means clustering (Lloyd, 1982) and named with GPT-5.

Table A.1 compares a hierarchical loss (HYDRA, Karl and Scherp, 2025) with standard binary cross-entropy for Ettin models. Instead of training a single classification head over the entire label space, HYDRA partitions labels by their hierarchy level and assigns a dedicated classification head to each level. Despite this, the differences between the two methods are negligible, and performance even regresses for the largest model.

Full vs. parameter-efficient generative fine-tuning using LoRA are compared in Table A.2. Full fine-tuning improves macro-F1 by 0.01 for Llama models (1B, 3B). Gains are greater for Ettin models (400M, 1B), at 0.12 and 0.09, respectively. Table A.3 compares instruction-tuned models with base models. Instruction-tuned Llama models (1B, 3B) achieve macro-F1 0.04–0.05 higher than the corresponding base variants.

A.7 Prompting setup

A.7.1 System prompt

```
SYSTEM_PROMPT = """
You are an AI assistant tasked with
    classifying a medical device adverse
     event report into one or more
    categories according to the FDA
    taxonomy. Your goal is to assign all
     relevant labels to the given report
    .

The report that needs classifying is
    provided within the <classification-
    text> tag. Along with the report,
    the label definitions are provided
    within the <labels> tag. To assist
    you with the task, we also include
    10 "few-shot" examples in the <few-
    shot-examples> tag. These are past
    reports similar to the one you are
    classifying - the past reports are
    accompanied by their corresponding
    labels which were tagged by a human
    expert.

# RULES:

1. The taxonomy of labels is provided
    within the <labels> tag.
    - Labels are separated by newlines;
        a definition for the label is
        provided.
    - We are in a 3-level hierarchical
        multi-label classification
        setting - this means that when a
         child label (such as A040507)
        is selected, the parent label (
        A0405) and grandparent label (
        A04) must also be selected.
        Similarly, if a parent label (
        A0405) is selected, the
        grandparent label (A04) must
        also be selected.
    - The converse is not always true -
        selecting a parent (A0405) or
        grandparent (A04) doesn't
        necessarily mean selecting all
        its children (A040507).
    - Labels that start with "A" are
        Medical Device Problems.
    - Labels that start with "E" are
        Health Effects - Clinical Signs
        and Symptoms or Conditions.
    - The grandparent label (eg. A01) is
         the most general label, the
        parent label (eg. A0101) is the
        next most specific, and the
        child label (eg. A010101) is the
         most specific.
    - The grandparent label always has
        the letter "A" or "E" followed
        by 2 numbers, eg. A01, E01, A02,
         E02, etc.
    - The parent label always has the
        letter "A" or "E" followed by 4
```

Model	Loss	Macro F1 ↑					J ↑	PRR ↑	ρ ↓	ECE_+ ↓
		Overall	Head	Medium	Tail	ET				
Ettin-150m-Encoder	BCE	0.46	0.68	0.56	0.44	0.07	0.55	0.38	-0.30	0.64
	HYDRA	0.47	0.69	0.55	0.44	0.12	0.56	0.42	-0.41	0.63
Ettin-400m-Encoder	BCE	0.51	0.72	0.61	0.50	0.12	0.58	0.44	-0.36	0.59
	HYDRA	0.49	0.70	0.57	0.46	0.14	0.57	0.45	-0.42	0.61
Ettin-1B-Encoder	BCE	0.53	0.73	0.63	0.51	0.13	0.61	0.46	-0.40	0.56
	HYDRA	0.48	0.70	0.58	0.44	0.13	0.57	0.50	-0.43	0.53

Table A.1: Comparison of BCE vs HYDRA loss across Ettin encoder models.

Fine-tuning/model	Macro F1 ↑					J ↑	PRR ↑	ρ ↓	ECE_+ ↓
Number of classes →	Overall *1154*	Head *144*	Medium *481*	Tail *348*	ET *181*				
Full									
Llama-3.2-3B-Instruct	**0.49**	**0.68**	**0.57**	**0.48**	**0.14**	**0.58**	**0.59**	**-0.45**	**0.55**
Llama-3.2-1B-Instruct	0.47	0.67	0.56	0.45	0.12	0.57	0.58	-0.43	0.57
Ettin-1B-Decoder	0.47	0.67	0.56	0.46	0.10	0.57	0.56	-0.43	0.57
Ettin-400M-Decoder	0.44	0.66	0.54	0.42	0.07	0.55	0.56	-0.44	0.60
Parameter-efficient									
Llama-3.2-3B-Instruct	**0.48**	**0.67**	**0.57**	**0.47**	**0.11**	**0.58**	0.58	**-0.46**	**0.56**
Llama-3.2-1B-Instruct	0.45	0.66	0.54	0.44	0.08	0.56	**0.60**	-0.45	0.59
Ettin-1B-Decoder	0.38	0.63	0.48	0.33	0.014	0.52	0.58	-0.44	0.67
Ettin-400M-Decoder	0.32	0.59	0.41	0.24	0.005	0.48	0.57	-0.45	0.73

Table A.2: Full vs. Parameter-Efficient Fine-Tuning (LoRA) – Performance and UQ. Performance metrics (macro F1, Jaccard J) and UQ measures (PRR, Spearman ρ, weighted ECE) are reported for both fine-tuning strategies. All models were trained in a generative setting. Macro F1 is the primary predictive metric, with results broken down by head, medium, tail, and extreme tail (ET) classes. PRR is reported for the best U_{info} metric (see Section 3.3). While UQ results vary, full fine-tuning results often in slightly better predictive performance in comparison to the parameter-efficient version.

```
        numbers, eg. A0101, E0101, A0201
        , E0201, etc.
    - The child label always has the
        letter "A" or "E" followed by 6
        numbers, eg. A010101, E010101,
        A020101, E020101, etc.

2. There are 10 "few-shot" examples
    included in the <few-shot-examples>
    tag.
    - Each example includes a report and
        its corresponding labels.
    - The examples included in the <few-
        shot-examples> tag were chosen
        using a K-Nearest Neighbours
        algorithm which picked reports
        similar in content to the text
        which needs classifying. The
        labels shown for these examples
        may or may not overlap with the
        labels for the report inside the
        <classification-text> tag. Use
        them as contextual guidance.

3. Your goal is to classify the text
    provided within the <classification-
    text> tag.
    - Assign all labels that are
        relevant.
    - You can choose multiple labels, a
        single label, or no labels if
        none apply.
    - Always use the exact label names
        from the label list provided in
        the taxonomy under the <labels>
        tag.  Do not invent new labels
        or modify existing ones.
    - Return your output as a list of
        labels, separated by newlines.
    - Do not include any explanations,
        text, or formatting outside the
        label list.
    - If no label applies, return an
        empty list.
    - Do not invent new labels.

4. Provide your output as a list of
    labels, each on a new line. For
    example:

A04
A0405
A040507
E01
E0101

# IMPORTANT
*In your final output, you must not
    include any extra text, explanations
    , or formatting outside the label
    list. Only return the list of labels
    separated by newlines.*
"""
```

Type/model	**Macro F1** ↑					**J** ↑	**PRR** ↑	ρ ↓	**ECE**$_+$ ↓
Number of classes →	Overall *1154*	Head *144*	Medium *481*	Tail *348*	ET *181*				
Instruction-tuned									
Llama-3.2-3B-Instruct	**0.49**	**0.68**	**0.57**	**0.48**	**0.14**	**0.58**	**0.59**	**-0.45**	**0.56**
Llama-3.2-1B-Instruct	0.47	0.67	0.56	0.45	0.12	0.57	0.58	-0.43	0.59
Base									
Llama-3.2-3B-Base	**0.48**	**0.67**	**0.57**	**0.46**	**0.12**	**0.58**	**0.70**	**-0.46**	**0.59**
Llama-3.2-1B-Base	0.43	0.63	0.52	0.39	0.10	0.45	0.53	-0.44	0.63

Table A.3: Instruction-Tuned vs. Base Models: Performance metrics (macro F1, J) and UQ measures (PRR, ρ, weighted ECE) are reported for both model types, which were fine-tuned using generative objectives. Instruction-tuned models achieve slightly higher predictive performance more often, whereas the 3B-Base model excels in UQ (highest PRR, lowest ρ and ECE).

Paradigm/model	**ECE**$_+$ ↓			
Number of classes →	Head *144*	Medium *481*	Tail *348*	ET *181*
Discriminative fine-tuning				
Llama-3.1-8B-Base	0.32	0.48	0.67	0.86
Llama-3.2-3B-Base	0.34	0.50	0.68	0.86
Llama-3.2-1B-Base	0.37	0.51	0.67	**0.85**
Ettin-1B-Encoder	**0.31**	**0.47**	**0.65**	**0.85**
Ettin-400m-Encoder	0.33	0.50	0.68	0.86
Ettin-150m-Encoder	0.38	0.55	0.74	0.88
Generative fine-tuning				
Llama-3.1-70B-Base	**0.27**	**0.39**	**0.54**	**0.87**
Llama-3.1-8B-Base	0.30	0.42	0.58	0.91
Llama-3.2-3B-Base	0.37	0.47	0.61	0.91
Llama-3.2-1B-Base	0.40	0.51	0.66	0.92
Ettin-1B-Decoder	0.37	0.48	0.61	0.92
Ettin-400m-Decoder	0.39	0.51	0.65	0.95
Prompting - instruct				
Llama-3.1-70B-Instruct	0.48	0.62	0.73	0.91
Llama-3.1-8B-Instruct	0.64	0.74	0.83	0.93
Qwen3-235B-A22B-Instruct	**0.40**	**0.51**	**0.61**	**0.75**
Qwen3-30B-A3B-Instruct	**0.40**	0.53	0.65	0.83
Qwen3-4B-Instruct	0.47	0.61	0.74	0.90
Kimi-K2-Instruct	0.93	0.96	0.98	1.00
GPT-4.1	0.44	0.56	0.64	0.79
Prompting - thinking				
Llama-3.3-Nemotron-49B-v1.5	0.43	0.54	0.65	0.79
Qwen3-235B-A22B-Thinking	**0.32**	**0.40**	**0.49**	**0.63**
Qwen3-30B-A3B-Thinking	0.44	0.51	0.60	0.72
Qwen3-4B-Thinking	0.47	0.57	0.68	0.81
DeepSeek-R1-0528	0.36	0.45	0.53	0.67
GLM-4.5-Air	0.44	0.56	0.69	0.82
GPT-5	NA	NA	NA	NA

Table A.4: Positive class expected calibration errors (head, medium, tail, extreme tail) for each paradigm/model.

A.7.2 User prompt

```
USER_PROMPT = """
You are an AI assistant tasked with
    classifying a medical device adverse
     event report into one or more
    categories according to the FDA
    taxonomy. Your goal is to assign all
     relevant labels to the given report
    . The rules were provided in the
    system prompt. For the sake of
    clarity, we are repeating them here:
- Assign all the labels that are
    relevant to the report, but only if
    you are sure about it.
- You can choose multiple labels, a
    single label, or no labels if none
    apply.
- Use exact label names from the
    provided taxonomy. Do not invent or
    modify labels.
- Include parent and grandparent labels
    when selecting a child label.
- Do not include any explanations, text,
     or formatting outside the label
    list.
- If no labels apply, return an empty
    list.
- Your output should contain only the
    list of applicable labels, with each
     label on a new line. You must not
    include any extra text, explanations
    , or formatting outside the label
    list.
- Provide your output as a list of
    labels, each on a new line.
Example output:
A04
A0405
A040507
E01
E0101

Here's how to proceed:

1. First,  familiarize yourself with the
     label definitions:

<labels>
A01: Patient Device Interaction Problem
    - Problem related to the interaction
     between the patient and the device.
A0101: Patient-Device Incompatibility -
    Problem associated with the
    interaction between the patient's
    physiology or anatomy and the device
     that affects the patient and/or the
     device.
.
.

</labels>

2. Review these few-shot examples of
    similar reports and their
    corresponding labels:

<few-shot-examples>
{EXAMPLES}
</few-shot-examples>

3. Now, carefully classify the following
     report:

<classification-text>
{CLASSIFICATION_TEXT}
</classification-text>"""
```

A.7.3 Variation With Self-Verbalized Confidence

```
"""
...
- Assign all labels that are relevant.
- You can choose multiple labels, a
    single label, or no labels if none
    apply.
- Always use the exact label names from
    the label list provided in the
    taxonomy under the <labels> tag.  Do
     not invent new labels or modify
    existing ones.
- Return your output as a JSON
    dictionary with label codes as keys
    and confidence scores as values.
    Confidence scores should be floats
    between 0 and 1, reflecting your
    certainty about each prediction.
- Do not include any explanations, text,
     or formatting outside the JSON
    dictionary.
- If no label applies, return an empty
    JSON object: {}.
- Do not invent new labels.

Provide your output as a JSON dictionary
    .
For example:
{
"A04": 0.92,
"A0405": 0.74,
"A040507": 0.50,
"E01": 0.95,
"E0101": 0.32
}
# IMPORTANT
* In your final output, you must not
    include any extra text, explanations
    , or formatting outside the JSON
    dictionary. Only return the JSON
    dictionary.
"""
```