Structured Confidence–Guided Online Adaptation for LLM-based Multi-Label Classification

Pengyu Xu, Jingren Hou, Liping Jing*, Jian Yu
Beijing Key Laboratory of Traffic Data Mining and Embodied Intelligence
Beijing Jiaotong University, Beijing, China
pengyu@bjtu.edu.cn

Abstract

Large language models (LLMs) enable zero-shot and few-shot multi-label text classification via in-context learning, yet most approaches perform static inference and degrade under streaming test data due to distribution shift and long-tail labels. We study online test-time adaptation for LLM-based multi-label generation without any parameter updates, and identify two bottlenecks: (1) standard generation probabilities provide unreliable confidence because they ignore label competition at key decoding branches; (2) naive confidence-based caching overfits to frequent and easy examples, reducing label coverage and diversity. We propose **SCOTTA**, a structured confidence–guided online adaptation framework. SCOTTA introduces Label-set Local Likelihood Ratio (L3R), a label-level confidence measure that compares a target label against its valid competitors at critical decision positions. Using L3R as a unified signal, SCOTTA maintains an in-context exemplar cache via streaming submodular maximization, balancing label coverage, semantic diversity, and sample quality under a fixed context budget. Across four benchmarks, SCOTTA consistently improves Micro-F1 and Macro-F1 over strong LLM and non-LLM baselines, with the largest gains on long-tail labels.

1 Introduction

Multi-Label Text Classification (MLTC) assigns one or more semantic labels to a text and underpins applications such as information retrieval, legal intelligence, and biomedical text analysis (Chang et al., 2021; Li et al., 2025a; Saha et al., 2025). Recent Large Language Models (LLMs) enable MLTC via in-context learning (ICL), offering strong zero-shot and few-shot performance without parameter updates (Brown et al., 2020; Sarkar et al., 2023; Zhu and Zamani, 2024). However, most LLM-based methods remain *static* at test time, processing instances independently and ignoring the fact that real-world test data often arrives as a stream with evolving distributions (Ma, 2024; Lin et al., 2024).

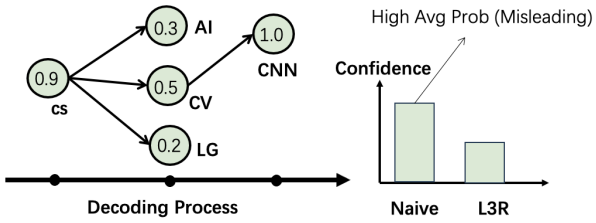

Figure 1: Illustration of the *confidence illusion* in generative multi-label classification. (Left) In label generation, most tokens follow shared prefixes with near-deterministic probabilities, while true competition among labels occurs only at a few critical branching positions. (Right) Naive confidence based on aggregated token likelihood is dominated by easy tokens and overestimates prediction reliability, whereas Label-set Local Likelihood Ratio (L3R) focuses on the decisive branching node and yields more faithful confidence estimates.

Online Test-Time Adaptation (OTTA) is a natural fit for ICL-based LLM inference, since it can leverage unlabeled test streams through experience accumulation rather than fine-tuning (Liu et al., 2024; Xia et al., 2025; Li et al., 2025b). Yet, extending OTTA to *LLM-based multi-label generation* faces two obstacles.

First, effective online adaptation requires reliable confidence estimation, yet standard sequence likelihood and token probability aggregation are misleading for multi-label generation (Malinin and Gales, 2021; Ma et al., 2025). Many labels share long prefixes and diverge only at a few critical decoding branches, where genuine competition occurs. As illustrated in Figure 1, aggregating token probabilities is dominated by near-deterministic structural tokens and fails to capture uncertainty at these decisive label-selection steps. This *con-*

* Corresponding author.

Findings of the Association for Computational Linguistics: ACL 2026, pages 16671–16686

fidence illusion systematically inflates prediction reliability, leading unreliable pseudo-labels to be repeatedly reused and ultimately undermining online adaptation.

Second, OTTA relies on caching predicted exemplars as demonstrations; naive confidence-based or first-in-first-out (FIFO) strategies easily collapse into frequent-label and easy-sample dominance, yielding semantic redundancy, poor label coverage, and degraded long-tail generalization (Li et al., 2025b; Ma, 2024).

We propose **S**tructured **C**onfidence-guided **O**nline **T**est-**T**ime **A**daptation (**SCOTTA**) to improve LLM-based MLTC without any parameter updates. SCOTTA treats confidence as a *structured decision signal* that jointly governs (i) which predictions are reliable enough to reuse and (ii) which reliable exemplars are most valuable to retain under a limited context budget. To this end, we introduce Label-set Local Likelihood Ratio (L3R), a label-level confidence metric that compares a target label against its valid competitors at key decoding positions, mitigating pseudo-confidence induced by shared prefixes and near-deterministic completions. Building on this signal, we formulate exemplar cache maintenance as a streaming submodular maximization problem (Badanidiyuru et al., 2014; Kirchhoff and Bilmes, 2014; Ji et al., 2024), balancing label coverage, semantic diversity, and sample quality to prevent cache degeneration over time.

Our contributions are:

- We propose SCOTTA, an OTTA framework for LLM-based multi-label classification that improves inference *without* fine-tuning.
- We introduce L3R, a decoding-consistent confidence measure that explicitly models label competition during generation.
- We maintain in-context exemplars via streaming submodular maximization to achieve balanced, diverse, and high-quality caches under a fixed budget.
- Experiments on four benchmarks (MOVIE, AAPD, RCV1, and StackExchange) show consistent Micro-/Macro-F1 improvements, with particularly strong gains on long-tail labels and extreme label spaces.

2 Method

2.1 Overall Framework

We study Online Test-Time Adaptation (OTTA) for LLM-based Multi-Label Text Classification (MLTC). Let the label set be

$$\mathcal{Y} = \{y^{(1)}, \ldots, y^{(K)}\}. \tag{1}$$

At test time, instances arrive sequentially as $\{x_t\}_{t=1}^{T}$. For each x_t, the LLM produces a multi-label prediction $\hat{\mathcal{Y}}(x_t) \subseteq \mathcal{Y}$ and a confidence score $C(x_t)$, while keeping all model parameters frozen.

SCOTTA performs OTTA by continuously reusing high-quality predictions as in-context demonstrations under a bounded cache. It consists of two coupled modules:

Label-set Local Likelihood Ratio. We compute label-level confidence by explicitly modeling label competition during generation, and aggregate it into an instance-level confidence $C(x_t)$ to identify reliable predictions for reuse.

Submodular Memory Bank (SMB). We maintain a bounded memory bank of exemplars via streaming submodular maximization, jointly balancing label coverage, semantic diversity, and sample quality to prevent degeneration toward frequent or redundant examples.

As illustrated in Figure 2, for each incoming x_t, SCOTTA (i) retrieves a small set of relevant exemplars from S_{t-1} and injects them into the prompt, (ii) generates $\hat{\mathcal{Y}}(x_t)$ and computes $C(x_t)$, and (iii) if $C(x_t)$ is sufficiently high, updates the cache to obtain S_t. This closed loop enables stable, continual adaptation in streaming test data without any parameter updates.

2.2 Multi-label Generation

LLM-based MLTC typically predicts labels *generatively*: the model outputs label names token by token conditioned on the input, rather than scoring each label independently. Let each label $y^{(k)} \in \mathcal{Y}$ be a token sequence

$$y^{(k)} = \left(y_1^{(k)}, \ldots, y_{T_k}^{(k)}\right), \qquad y_t^{(k)} \in \mathcal{V}, \tag{2}$$

where $\mathcal{V}$ is the vocabulary. Given an input x and the current prefix $z_{<t}$, the LLM defines a next-token distribution

$$p_t(v) = p(v \mid x, z_{<t}), \qquad v \in \mathcal{V}. \tag{3}$$

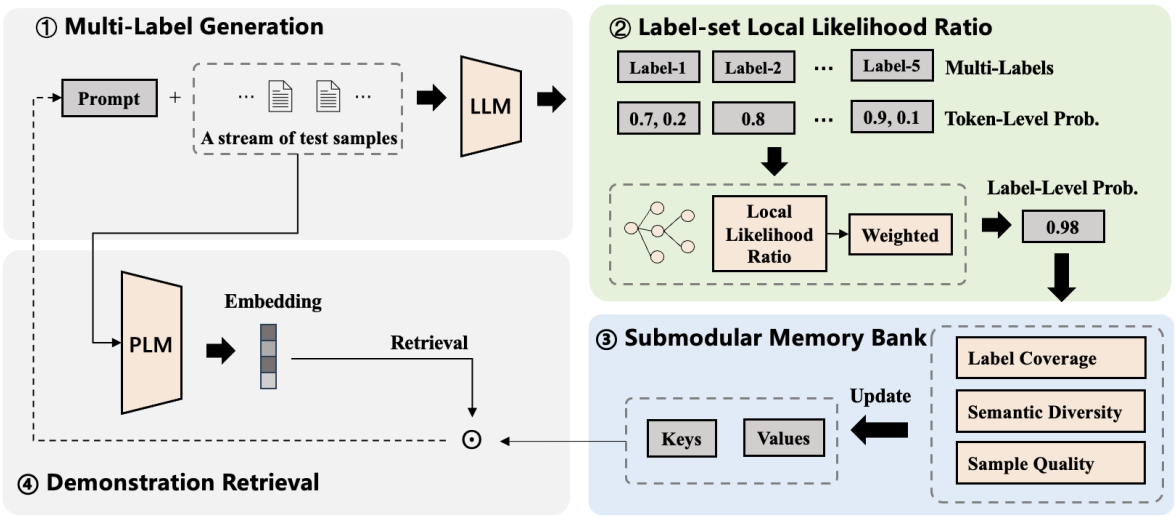

Figure 2: Overview of the proposed SCOTTA framework for LLM-based multi-label classification under online test-time adaptation. (1) Given a stream of unlabeled test instances, an LLM performs multi-label generation via in-context prompting. (2) For each predicted label, SCOTTA computes a structured label-level confidence using the Label-set Local Likelihood Ratio (L3R), which aggregates token-level probabilities while explicitly modeling local competition among valid label candidates at critical decoding positions. (3) Predictions with reliable confidence are stored in a bounded submodular memory bank, which is updated online to jointly balance label coverage, semantic diversity, and sample quality. (4) For each incoming test instance, semantically relevant exemplars are retrieved from the memory bank and injected as in-context demonstrations, enabling continual adaptation over the test stream without any parameter updates.

A key property of this decoding process is that *label competition is highly localized*. Many labels share long prefixes, so most positions are near-deterministic (e.g., common namespace tokens), while genuine discrimination happens only at a few branching points where valid labels diverge. Consequently, sequence likelihood, perplexity, or naive averages of token probabilities can be dominated by shared-prefix and deterministic-completion tokens, producing *structural pseudo-confidence* that does not reflect the model's true preference among competing labels.

This motivates confidence modeling that explicitly focuses on the critical decision positions induced by the label set, which we develop next via Label-set Local Likelihood Ratio (L3R).

2.2.1 An Illustrative Example of Multi-label Generation

Multi-label generation induces *localized* competition. Consider three labels $\{\texttt{cs.AI}, \texttt{cs.CV.CNN}, \texttt{cs.LG}\}$ for an input x about convolutional neural networks. During decoding, these labels share the prefix `cs.`, so early tokens are near-deterministic and provide little discriminative signal. The *true* decision occurs at the branching token after `cs.`, where the model must choose among valid continuations:

$$p(\texttt{CV} \mid x, \texttt{cs.}) = 0.50,$$
$$p(\texttt{AI} \mid x, \texttt{cs.}) = 0.30,$$
$$p(\texttt{LG} \mid x, \texttt{cs.}) = 0.20.$$

After selecting a branch (e.g., `cs.CV`), subsequent completion tokens (e.g., `.CNN`) are often again near-deterministic. This illustrates why confidence should focus on a few branching positions where valid labels compete, rather than on overall sequence likelihood dominated by shared prefixes and deterministic continuations.

2.2.2 Label Competition from a Generative Perspective

The example above highlights a central property of generative multi-label prediction: *label competition is sparse and position-specific*. Most decoding steps correspond to shared prefixes or deterministic completions and thus carry little discriminative information, while only a few branching positions determine which valid label path is selected.

However, many existing confidence measures rely on sequence likelihood, perplexity, or uniform aggregations of token probabilities. These measures conflate informative branching decisions with

uninformative tokens, and therefore fail to distinguish between (i) predictions supported by clear preferences at true decision points and (ii) predictions whose high probability arises mainly from shared prefixes or deterministic continuations. In online test-time adaptation, such *structural pseudo-confidence* can cause unstable or incorrect predictions to be repeatedly reused, leading to error accumulation over time.

2.2.3 From Generation Mechanism to Confidence Modeling

Motivated by this observation, we view multi-label prediction as a *structured decoding process under label-set constraints*. From this perspective, confidence should reflect how strongly the model prefers a target label path over other valid label paths *at the branching positions where competition actually occurs*, rather than the overall generation probability.

This principle motivates the Label-set Local Likelihood Ratio (L3R) introduced next. L3R directly compares the probability of a target label token against its competing label tokens at local decision positions, yielding a decoding-consistent and interpretable confidence signal tailored to generative multi-label classification.

2.3 Label-set Local Likelihood Ratio

We propose **Label-set Local Likelihood Ratio (L3R)** to measure label-level confidence for generative MLTC by explicitly comparing a target label path against its *valid competitors* at the few decoding positions where real competition occurs.

Competition set induced by the label set. For a candidate label $y^{(k)} = \big(y_1^{(k)}, \ldots, y_{T_k}^{(k)}\big)$, define the prefix at position t as

$$\mathrm{pref}_t^{(k)} = \big(y_1^{(k)}, \ldots, y_{t-1}^{(k)}\big). \tag{4}$$

Only labels sharing the same prefix are feasible competitors at t:

$$\mathcal{A}_t^{(k)} = \Big\{\, j \;\Big|\; \mathrm{pref}_t^{(j)} = \mathrm{pref}_t^{(k)} \,\Big\}. \tag{5}$$

Local likelihood ratio at a decision position. Let the LLM next-token distribution be $p_t(v) = p(v \mid x, z_{<t})$. We define the target-path and competing-path probabilities as

$$p_t^{\text{self}} = p_t\Big(y_t^{(k)}\Big), \tag{6}$$

$$p_t^{\text{comp}} = \sum_{\substack{j \in \mathcal{A}_t^{(k)} \\ j \neq k}} p_t\Big(y_t^{(j)}\Big). \tag{7}$$

The local preference for $y^{(k)}$ at t is captured by

$$\mathrm{LLR}_t^{(k)} = \log \frac{p_t^{\text{self}} + \epsilon}{p_t^{\text{comp}} + \epsilon}, \tag{8}$$

where $\epsilon > 0$ is a smoothing constant.

Emphasizing informative positions. Shared-prefix or near-deterministic completion tokens should contribute less. We quantify how "competitive" a position is by normalizing

$$q_t^{\text{self}} = \frac{p_t^{\text{self}}}{p_t^{\text{self}} + p_t^{\text{comp}} + \epsilon}, \qquad q_t^{\text{comp}} = 1 - q_t^{\text{self}}, \tag{9}$$

and computing the binary entropy

$$\mathcal{J}_t^{(k)} = -\sum_{c \in \{\text{self},\text{comp}\}} q_t^c \log q_t^c. \tag{10}$$

We then assign a softmax weight over positions:

$$w_t^{(k)} = \frac{\exp\Big(\alpha \mathcal{J}_t^{(k)}\Big)}{\sum_{i=1}^{T_k} \exp\Big(\alpha \mathcal{J}_i^{(k)}\Big)}, \tag{11}$$

where $\alpha > 0$ controls the emphasis on high-information decision points.

L3R confidence. Finally, the label-level confidence is the weighted sum of local likelihood ratios:

$$\mathrm{L3R}\Big(y^{(k)} \mid x\Big) = \sum_{t=1}^{T_k} w_t^{(k)} \cdot \mathrm{LLR}_t^{(k)}. \tag{12}$$

L3R is *decoding-consistent*: it aggregates evidence mainly from positions where competing labels truly branch, suppressing pseudo-confidence dominated by shared prefixes or deterministic completions.

2.4 Submodular Memory Bank

While L3R identifies reliable predictions, effective online test-time adaptation further requires *organizing* these predictions into a compact and informative experience set. Simply retaining the most confident samples or using FIFO updates leads to degeneration toward frequent labels and redundant examples. We therefore maintain a **Submodular Memory Bank (SMB)** that selects exemplars as a *set-level optimization* problem under a fixed budget.

Problem formulation. At time step t, let

$$\mathcal{U}_t = \{(x_i, \hat{\mathcal{Y}}_i, q_i)\}_{i=1}^{t} \tag{13}$$

be the stream of candidate exemplars with confidence $q_i = C(x_i)$. The goal is to maintain a memory bank $S_t \subseteq \mathcal{U}_t$ with $|S_t| \leq B$ that maximizes a monotone submodular objective

$$F(S) = \lambda_1 F_{\text{cov}}(S) + \lambda_2 F_{\text{div}}(S) + \lambda_3 F_{\text{qual}}(S). \tag{14}$$

Objective components. *(i) Label coverage* encourages balanced exposure of labels, especially long-tail ones:

$$F_{\text{cov}}(S) = \sum_{l \in \mathcal{L}} \frac{1}{\sqrt{f_l(S) + \epsilon}} \min\left(1, \sum_{i \in S} \mathbf{1}[l \in \hat{\mathcal{Y}}_i]\right), \tag{15}$$

where $f_l(S)$ is the frequency of label l in S.

(ii) Semantic diversity reduces redundancy by selecting representative samples in embedding space:

$$F_{\text{div}}(S) = \sum_{u \in \mathcal{U}_t} \max_{i \in S} s(u, i), \tag{16}$$

with $s(\cdot, \cdot)$ denoting cosine similarity.

(iii) Sample quality preserves reliability by favoring high-confidence predictions:

$$F_{\text{qual}}(S) = \sum_{i \in S} q_i. \tag{17}$$

Online update. As samples arrive sequentially, we adopt an online greedy update. For a new candidate i, define the marginal gain

$$\Delta(i \mid S) = F(S \cup \{i\}) - F(S). \tag{18}$$

If $|S| < B$, we insert i; otherwise, we replace the element with the smallest contribution if $\Delta(i \mid S \setminus \{j\}) > 0$. This strategy provides a constant-factor approximation guarantee for monotone submodular maximization under a cardinality constraint.

Adaptive inference with SMB. For a test input x_t, we retrieve the k nearest exemplars from S_{t-1} in embedding space and inject them as in-context demonstrations. As the stream progresses, SMB continuously evolves, yielding a compact, balanced, and high-quality memory that stabilizes adaptation—particularly for long-tail labels and distribution shifts.

Dataset	N	L	L_{avg}	N_{avg}
MOVIE	117,352	27	2.15	105.46
AAPD	55,840	54	2.41	163.00
RCV1	804,418	103	3.24	124.00
StackExchange	49,447	12,892	2.43	88.94

Table 1: Statistics of the evaluated datasets.

2.5 Demonstration Retrieval and Prompting

Given the current memory bank S_{t-1}, SCOTTA retrieves a small set of exemplars to construct the in-context demonstrations for the incoming instance x_t. We encode x_t and each exemplar text x_i into embeddings (using the same sentence encoder as in SMB) and perform k-NN search within S_{t-1} by cosine similarity. The retrieved exemplars are ranked by similarity and formatted as `(input` $\rightarrow$ `label-set)` pairs, which are concatenated into the prompt before querying the LLM. To respect a fixed context budget, we keep at most k exemplars and truncate the oldest exemplars if the prompt exceeds the token limit. This retrieval step is lightweight and training-free, while SMB ensures that S_{t-1} remains balanced and non-redundant over the stream.

3 Experiments

3.1 Experimental Setup

Datasets and Evaluation Metrics We evaluate SCOTTA on four public multi-label text classification benchmarks that span small/medium label spaces and an extreme multi-label setting: MOVIE (27 labels) Arevalo et al. (2017), AAPD (54) Yang et al. (2018), RCV1 (103) Lewis et al. (2004), and StackExchange (SE; 12,892) Xia et al. (2013); Tang et al. (2019). This suite covers diverse domains (movie plots, paper abstracts, news, and QA posts) and varying degrees of long-tail imbalance, allowing us to test the robustness of online test-time adaptation across different label cardinalities and label-frequency distributions. Table 1 reports dataset statistics, where N is the number of instances, L the number of labels, L_{avg} the average labels per instance, and N_{avg} the average instances per label.

Following previous work (Zhang et al., 2021; Xia et al., 2025), we randomly sample 10,000 instances from each dataset for evaluation. We report Micro-F1 and Macro-F1 scores. Micro-F1 reflects overall performance dominated by frequent labels, whereas Macro-F1 computes the average F1 score

Methods	MOVIE		AAPD		RCV1		SE	
	mi-F1	ma-F1	mi-F1	ma-F1	mi-F1	ma-F1	mi-F1	ma-F1
RoBERTa	26.32	21.59	31.16	29.00	32.92	26.24	23.95	15.81
SimCSE	23.90	19.61	17.60	16.62	35.62	28.11	27.31	18.62
MPNet	27.88	21.51	33.31	30.64	44.10	42.59	35.02	25.99
GPT-3.5	62.67	43.43	34.61	28.78	59.85	54.62	53.24	44.07
GPT-4o	68.74	52.39	47.96	40.13	63.45	57.42	60.12	47.97
Qwen2.5	52.05	37.60	41.54	30.46	56.43	52.44	48.38	31.91
Qwen3	51.76	37.61	40.36	31.21	60.60	56.62	49.00	33.06
ICXML	64.46	54.58	35.88	28.86	62.41	56.54	61.97	49.30
PESCO	32.76	22.53	34.24	27.77	47.99	44.23	42.48	30.88
PIEClass	36.68	27.44	37.53	27.50	49.98	45.48	43.48	34.07
SCOTTA (Ours)	**71.86**	**59.15**	**53.17**	**44.49**	**65.56**	**61.49**	**65.88**	**51.30**

Table 2: Main results on four multi-label text classification benchmarks.

across labels and is more sensitive to rare labels, aligning with our focus on robustness to long-tailed distributions under streaming test data.

Baselines We compare against representative baselines from four groups.

(1) Embedding-based zero-shot matching casts prediction as text–label retrieval using RoBERTa (Liu et al., 2019), SimCSE (Gao et al., 2021), and MPNet (Song et al., 2020) sentence embeddings.

(2) Static LLM inference directly generates the label set using a fixed prompt with no test-time adaptation. We include GPT-3.5 (`gpt-3.5-turbo-0125`) (OpenAI, 2024a), GPT-4o (OpenAI, 2024b), and two open-source backbones, Qwen2.5-7B-Instruct (Yang et al., 2024) and Qwen3-8B-Instruct (Yang et al., 2025).

(3) LLM-based structured multi-label inference includes ICXML (Zhu and Zamani, 2024), which follows a candidate generation and re-ranking pipeline to handle large label spaces.

(4) Pseudo-labeling/self-training methods (PESCO (Wang et al., 2023) and PIEClass (Zhang et al., 2023)) are included as training-based references.

3.2 Main Results

Table 2 reports the performance of SCOTTA and all baselines on four benchmarks using Micro-F1 and Macro-F1. Without any parameter updates, SCOTTA consistently outperforms both non-LLM methods and static LLM inference across all datasets.

Compared with the strongest static backbone (GPT-4o), SCOTTA yields clear improvements in both metrics. On MOVIE, SCOTTA improves Micro-F1 and Macro-F1 by 3.12 and 6.76 points, respectively, indicating that online adaptation is beneficial even with a small label space. On AAPD, the gains increase to 5.21 (Micro-F1) and 4.36 (Macro-F1), showing that accumulated test-time experience effectively benefits medium- and low-frequency labels. Similar trends are observed on RCV1.

On the extreme multi-label dataset StackExchange, which contains 12,892 labels, SCOTTA maintains a stable advantage over GPT-4o, improving Micro-F1 by 5.76 points and Macro-F1 by 3.33 points. This demonstrates that the proposed approach scales well to very large label spaces and remains robust under severe long-tail distributions.

Across all datasets, the improvements in Macro-F1 are consistently larger than those in Micro-F1. This suggests that SCOTTA is particularly effective at mitigating head-label dominance and enhancing long-tail generalization, validating the benefit of combining structured confidence estimation with principled exemplar selection in online test-time adaptation.

3.3 Ablation Study

We conduct ablation studies to analyze the contribution of each component in SCOTTA under two LLM backbones, GPT-3.5 and GPT-4o. The results are shown in Table 3 and Table 4.

Introducing L3R alone yields consistent but relatively limited improvements over static LLM inference. This indicates that explicitly modeling label competition during generation improves the

Variants	MOVIE		AAPD		RCV1		SE	
	mi-F1	ma-F1	mi-F1	ma-F1	mi-F1	ma-F1	mi-F1	ma-F1
LLM (GPT-3.5)	62.67	43.43	34.61	28.78	59.85	54.62	53.24	44.07
+ SMB	65.44	49.41	39.54	34.37	61.98	58.14	58.03	46.46
+ L3R	63.52	43.90	35.79	29.67	59.55	54.89	56.72	44.83
L3R + SMB (Ours)	**66.79**	**51.19**	**40.82**	**36.14**	**63.96**	**60.69**	**59.00**	**48.40**

Table 3: Ablation results with **GPT-3.5**.

Variants	MOVIE		AAPD		RCV1		SE	
	mi-F1	ma-F1	mi-F1	ma-F1	mi-F1	ma-F1	mi-F1	ma-F1
LLM (GPT-4o)	68.74	52.39	47.96	40.13	63.45	57.42	60.12	47.97
+ SMB	70.51	56.37	51.89	43.72	63.58	59.94	64.91	49.36
+ L3R	69.59	52.86	49.14	41.02	63.15	57.69	63.60	47.73
L3R + SMB (Ours)	**71.86**	**59.15**	**53.17**	**44.49**	**65.56**	**61.49**	**65.88**	**51.30**

Table 4: Ablation results with **GPT-4o**.

reliability of confidence estimation and suppresses some erroneous predictions, but is not sufficient to substantially change the overall inference behavior when used in isolation.

In contrast, submodular memory bank (SMB) brings more pronounced gains, especially in terms of Macro-F1. By explicitly encouraging label coverage and semantic diversity, SMB avoids cache degeneration toward frequent or redundant examples, which is crucial in the online test-time adaptation setting. The effect is particularly noticeable on datasets with imbalanced label distributions.

The best performance is achieved when L3R and SMB are combined. Across both backbones and all datasets, the full model consistently outperforms variants that include only one component. This demonstrates a clear synergy between structured confidence modeling and submodular exemplar selection: L3R provides reliable candidates for reuse, while SMB ensures that the accumulated experience remains balanced and informative.

Finally, the observed trends are consistent under both GPT-3.5 and GPT-4o, despite their differences in scale and capability. This suggests that SCOTTA is largely model-agnostic and that its effectiveness does not rely on a specific LLM architecture.

3.4 Streaming Test Performance

To further verify the effectiveness of SCOTTA under streaming test-time adaptation, we report *online* performance on AAPD by evaluating the cumulative Micro-F1 and Macro-F1 every 500 test instances, up to 10,000 instances in total. At time step t, SCOTTA performs inference on x_t using demonstrations retrieved from the current memory bank S_{t-1}, and updates the cache if the prediction confidence exceeds the predefined threshold. In contrast, static LLM inference relies on a fixed prompt and remains unchanged throughout the test stream.

Figure 3 visualizes the streaming performance of different methods as the number of observed test instances increases. While static inference remains nearly flat over time, SCOTTA exhibits a clear and consistent upward trend as more test samples are processed.

Notably, the improvement is more pronounced in Macro-F1 than in Micro-F1, indicating that the evolving memory bank becomes increasingly beneficial for medium- and low-frequency labels. This behavior supports our claim that structured confidence estimation (L3R) and submodular memory maintenance (SMB) jointly prevent cache degeneration and enable the accumulation of higher-quality demonstrations over the test stream, leading to continual performance gains without any parameter updates.

3.5 Parameter Sensitivity

We analyze the sensitivity of SCOTTA to two key hyperparameters introduced in Appendix B: (i) the number of retrieved in-context demonstrations k, and (ii) the label coverage weight λ_1 in the submodular memory objective. Unless otherwise stated, all other hyperparameters follow the dataset-specific settings in Table 5.

Effect of retrieved demonstration number k. We vary k around the dataset-specific default values reported in Table 5, while keeping the memory

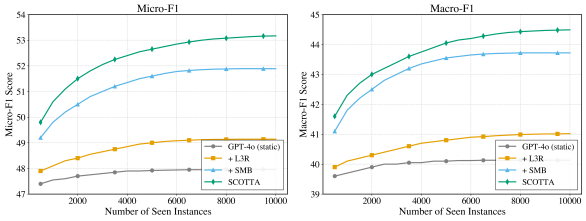

Figure 3: Streaming test-time performance on AAPD. Cumulative Micro-F1 and Macro-F1 are evaluated every 500 test instances for static LLM inference, partial variants, and SCOTTA. While static inference remains nearly flat, SCOTTA shows a steady and monotonic improvement as more test instances are observed, with particularly pronounced gains in Macro-F1, demonstrating the effectiveness of structured confidence and submodular memory maintenance under streaming adaptation.

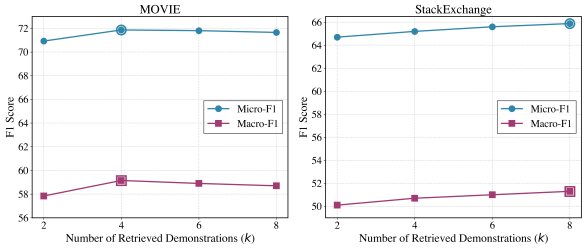

Figure 4: Sensitivity analysis of the retrieved demonstration number k. Results are reported on a small-label dataset (MOVIE) and an extreme multi-label dataset (SE), with all other hyperparameters fixed. Performance peaks near the default settings (k=4 for MOVIE and k=8 for SE), indicating a favorable trade-off between contextual sufficiency and redundancy under a fixed context budget.

capacity B and SMB weights fixed. Figure 4 visualizes the results on a small-label dataset (MOVIE) and an extreme multi-label dataset (SE).

Across both datasets, SCOTTA exhibits stable performance under moderate changes of k. On MOVIE, performance peaks at k=4, indicating that a small number of high-quality demonstrations is sufficient under limited label space. In contrast, SE benefits from a larger retrieval size, with the best performance achieved at k=8, reflecting the need for richer contextual evidence in extreme multi-label settings. In both cases, the default choices correspond to the best or near-best operating points under a fixed context budget.

Effect of label coverage weight λ_1. We further study the sensitivity to the label coverage weight λ_1 by varying it around the default values in Table 5, while keeping the ratio between the diversity weight λ_2 and the quality weight λ_3 fixed and renormalizing $\lambda_1 + \lambda_2 + \lambda_3 = 1$. Results are illustrated in Figure 5.

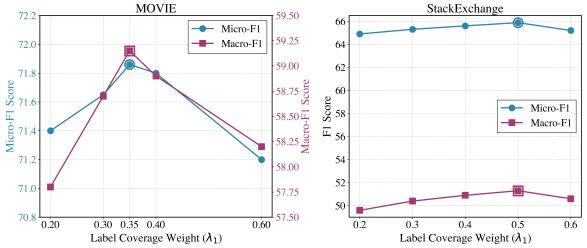

Figure 5: Sensitivity analysis of the label coverage weight λ_1 in the submodular memory objective. We vary λ_1 while keeping the ratio between the diversity and quality weights fixed and renormalizing $\lambda_1 + \lambda_2 + \lambda_3 = 1$. Optimal performance is achieved near λ_1=0.35 on MOVIE and λ_1=0.50 on SE, consistent with the heuristic guidelines described in Appendix B.1.

As shown in the figure, performance exhibits a clear unimodal trend with respect to λ_1. On MOVIE, the best results are obtained around λ_1=0.35, while on SE the optimal region shifts towards a larger value around λ_1=0.50. This behavior aligns with the intuition that datasets with larger and more long-tailed label spaces require stronger emphasis on label coverage during memory maintenance. Importantly, performance degrades gracefully when deviating from these optima, indicating that SCOTTA is not overly sensitive to precise tuning of λ_1.

Overall, SCOTTA demonstrates robust behavior under moderate variations of both k and λ_1. Performance consistently peaks near the default values derived from simple heuristics (Table 5), and remains stable across a broad parameter range. These results suggest that the proposed framework does not rely on delicate hyperparameter tuning, and that the adopted rules generalize well across datasets with different label-space scales.

4 Related Work

LLMs for Multi-Label Text Classification Early work on multi-label text classification (MLTC) mainly relied on fine-tuning discriminative encoders with specialized objectives to model label correlations (You et al., 2019; Ma et al., 2021; Zhang et al., 2021). Recently, Large Language Models (LLMs) have enabled generative MLTC through zero-shot and in-context learning paradigms (Brown et al., 2020; Sarkar et al., 2023). Subsequent studies improved performance by retrieving semantically similar demonstrations or can-

didate label sets, especially for large label spaces (Zhu and Zamani, 2024; Zhang et al., 2025; Ortego et al., 2025). However, most existing LLM-based methods operate in a *static* inference regime, assuming i.i.d. test instances and ignoring distribution shifts in streaming scenarios. In contrast, our work focuses on *online* adaptation for LLM-based multi-label generation.

Online Test-Time Adaptation Test-Time Adaptation (TTA) aims to adapt models to target distributions during inference without retraining (Ma, 2024). While classic TTA methods update model parameters (e.g., entropy minimization in TENT (Wang et al., 2021)), such approaches are often infeasible for LLMs due to computational cost or API constraints. Recent efforts therefore explore training-free adaptation via dynamic in-context demonstration selection (Zhang et al., 2022; Su et al., 2024). However, existing streaming ICL methods typically rely on FIFO buffers or greedy confidence filtering, which easily overfit frequent and easy examples, leading to poor long-tail coverage and reduced diversity (Li et al., 2025b; Bai et al., 2025). Our method instead leverages submodular optimization to maintain a bounded and balanced memory bank, explicitly accounting for coverage, diversity, and sample quality under streaming constraints (Badanidiyuru et al., 2014; Wang et al., 2025).

Confidence Estimation for LLM Adaptation Reliable confidence estimation is critical for self-training and online adaptation. Most prior work relies on global sequence likelihood or average token probability, which has been shown to be poorly calibrated for LLMs (Malinin and Gales, 2021; Kuhn et al., 2023). Although improved calibration techniques have been proposed, such as semantic entropy or verbalized confidence (Kuhn et al., 2023; Lin et al., 2022), they do not explicitly consider the structural properties of multi-label generation. In practice, label sequences often share long deterministic prefixes, while true uncertainty concentrates at a few critical branching positions. Our proposed **L3R** metric directly models this label-level competition by measuring local likelihood ratios at these decision points, providing a more faithful and adaptation-ready confidence signal.

5 Conclusion

We propose SCOTTA, a training-free online test-time adaptation framework for LLM-based multi-label text classification. SCOTTA combines Label-set Local Likelihood Ratio (L3R) for decoding-consistent, competition-aware confidence estimation with a Submodular Memory Bank (SMB) for maintaining a compact and informative in-context cache under a fixed budget. Experiments on four benchmarks show consistent improvements over strong baselines, with larger gains on long-tail labels and large label spaces, demonstrating stable adaptation without any parameter updates.

Limitations

SCOTTA has several limitations. (1) L3R requires access to token-level probabilities or log-probabilities, which may not be available for some LLM APIs. (2) The method adds inference overhead from confidence computation, embedding-based retrieval, and submodular cache updates, which can be costly in high-throughput streams or extremely large label spaces. (3) SMB depends on the embedding quality; misaligned embeddings may reduce retrieval and cache selection effectiveness. (4) Under severe distribution shifts, confident but incorrect predictions may still accumulate, and stronger calibration or verification signals could further improve robustness.

Acknowledgments

This work was partly supported by the National Key Research and Development Program of China under Grant 2024YFE0202900; the NSFC projects 62441614; the National Natural Science Foundation of China under Grant (62436001, 62176020); the Joint Foundation of the Ministry of Education for Innovation team (8091B042235); the Fundamental Research Funds for the Central Universities (2019JBZ110); and the State Key Laboratory of Rail Traffic Control and Safety (Contract No.RCS2023K006), Beijing Jiaotong University.

References

John Arevalo, Thamar Solorio, Manuel Montes y Gómez, and Fabio A. González. 2017. Gated multimodal units for information fusion. *ArXiv*, abs/1702.01992.

Ashwinkumar Badanidiyuru, Baharan Mirzasoleiman, Amin Karbasi, and Andreas Krause. 2014. Stream-

ing submodular maximization: massive data summarization on the fly. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, page 671–680, New York, NY, USA. Association for Computing Machinery.

Fan Bai, Hamid Hassanzadeh, Ardavan Saeedi, and Mark Dredze. 2025. LLMs are better than you think: Label-guided in-context learning for named entity recognition. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 28372–28392, Suzhou, China. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems (NeurIPS)*.

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

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Baijun Ji, Xiangyu Duan, Zhenyu Qiu, Tong Zhang, Junhui Li, Hao Yang, and Min Zhang. 2024. Submodular-based in-context example selection for LLMs-based machine translation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15398–15409, Torino, Italia. ELRA and ICCL.

Katrin Kirchhoff and Jeff Bilmes. 2014. Submodularity for data selection in statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty in large language models. In *International Conference on Learning Representations (ICLR)*.

David D. Lewis, Yiming Yang, Tony G. Rose, and Fan Li. 2004. Rcv1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5:361–397.

Ang Li, Yiquan Wu, Ming Cai, Adam Jatowt, Xiang Zhou, Weiming Lu, Changlong Sun, Fei Wu, and Kun Kuang. 2025a. Legal judgment prediction based on knowledge-enhanced multi-task and multi-label text classification. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2025 - Volume 1: Long Papers, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*, pages 6957–6970. Association for Computational Linguistics.

Yue Li, Zhixue Zhao, and Carolina Scarton. 2025b. Label set optimization via activation distribution kurtosis for zero-shot classification with generative models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 31724–31741, Suzhou, China. Association for Computational Linguistics.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. Teaching models to express their uncertainty in words. *Preprint*, arXiv:2205.14334.

Yuqi Lin, Minghao Chen, Kaipeng Zhang, Hengjia Li, Mingming Li, Zheng Yang, Dongqin Lv, Binbin Lin, Haifeng Liu, and Deng Cai. 2024. Tagclip: a local-to-global framework to enhance open-vocabulary multi-label classification of clip without training. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press.

An Liu, Zonghan Yang, Zhenhe Zhang, Qingyuan Hu, Peng Li, Ming Yan, Ji Zhang, Fei Huang, and Yang Liu. 2024. PANDA: Preference adaptation for enhancing domain-specific abilities of LLMs. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10960–10977, Bangkok, Thailand. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Jing Ma. 2024. Improved self-training for test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 23701–23710.

Marcus Ma, Georgios Chochlakis, Niyantha Maruthu Pandiyan, Jesse Thomason, and Shrikanth Narayanan. 2025. Large language models do multi-label classification differently. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 2472–2495, Suzhou, China. Association for Computational Linguistics.

Qianwen Ma, Chunyuan Yuan, Wei Zhou, and Songlin Hu. 2021. Label-specific dual graph neural network for multi-label 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 1: Long Papers)*, pages 3855–3864.

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

OpenAI. 2024a. Gpt-3.5 turbo model documentation. https://platform.openai.com/docs/models/gpt-3.5-turbo. Accessed: 2026-01-06.

OpenAI. 2024b. Hello gpt-4o. https://openai.com/index/hello-gpt-4o/. Accessed: 2026-01-06.

Diego Ortego, Marlon Rodríguez, Mario Almagro, Kunal Dahiya, David Jiménez, and Juan C. SanMiguel. 2025. Large language models meet extreme multi-label classification: Scaling and multi-modal framework. *Preprint*, arXiv:2511.13189.

Pramit Saha, Divyanshu Mishra, Felix Wagner, Konstantinos Kamnitsas, and J. Alison Noble. 2025. Incongruent multimodal federated learning for medical vision and language-based multi-label disease detection. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pages 28331–28339. AAAI Press.

Souvika Sarkar, Dongji Feng, and Shubhra Kanti Karmaker Santu. 2023. Zero-shot multi-label topic inference with sentence encoders and llms. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 16218–16233. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *arXiv preprint arXiv:2004.09297*.

Yi Su, Yunpeng Tai, Yixin Ji, Juntao Li, Yan Bowen, and Min Zhang. 2024. Demonstration augmentation for zero-shot in-context learning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14232–14244, Bangkok, Thailand. Association for Computational Linguistics.

Shijie Tang, Yuan Yao, Suwei Zhang, Feng Xu, Tianxiao Gu, Hanghang Tong, Xiaohui Yan, and Jian Lu. 2019. An integral tag recommendation model for textual content. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/IAAI'19/EAAI'19. AAAI Press.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. 2021. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations (ICLR)*.

Y. Wang et al. 2025. Streaming stochastic submodular maximization with on-demand user requests. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Yau-Shian Wang, Ta-Chung Chi, Ruohong Zhang, and Yiming Yang. 2023. PESCO: prompt-enhanced self contrastive learning for zero-shot text classification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 14897–14911. Association for Computational Linguistics.

Mingxuan Xia, Zhijie Jiang, Haobo Wang, Junbo Zhao, Tianlei Hu, and Gang Chen. 2025. Ensembling prompting strategies for zero-shot hierarchical text classification with large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 18189–18208, Suzhou, China. Association for Computational Linguistics.

Xin Xia, David Lo, Xinyu Wang, and Bo Zhou. 2013. Tag recommendation in software information sites. In *Proceedings of the 10th Working Conference on Mining Software Repositories (MSR)*, pages 287–296. IEEE.

An Yang et al. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

An Yang et al. 2025. Qwen3 technical report. *arXiv 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. *arXiv preprint arXiv:1806.04822*.

Ronghui You, Zihan Zhang, Ziye Wang, Suyang Dai, Hiroshi Mamitsuka, and Shanfeng Zhu. 2019. Attentionxml: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification. *Advances in Neural Information Processing Systems*, 32:5820–5830.

Jinbin Zhang, Nasib Ullah, and Rohit Babbar. 2025. Large language model as a teacher for zero-shot tagging at extreme scales. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 3465–3478, Abu Dhabi, UAE. Association for Computational Linguistics.

Qian-Wen Zhang, Ximing Zhang, Zhao Yan, Ruifang Liu, Yunbo Cao, and Min-Ling Zhang. 2021. Correlation-guided representation for multi-label text classification. In *IJCAI*, pages 3363–3369.

Renrui Zhang, Zhang Wei, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Jiao Qiao, and Hongsheng Li. 2022. Tip-adapter: Training-free adaption of clip for few-shot classification. volume abs/2207.09519.

Yunyi Zhang, Minhao Jiang, Yu Meng, Yu Zhang, and Jiawei Han. 2023. Pieclass: Weakly-supervised text classification with prompting and noise-robust iterative ensemble training. In *Proceedings of the 2023*

Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023, pages 12655–12670. 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, Mexico City, Mexico, June 16-21, 2024*, pages 2086–2098. Association for Computational Linguistics.

A Additional Experimental Details

A.0.1 Implementation Details

LLM backbones. Our main results use GPT-3.5 and GPT-4o as closed-source backbones accessed via official APIs. We additionally report open-source results using Qwen2.5-7B-Instruct and Qwen3-8B-Instruct deployed on NVIDIA A100 80GB GPUs.

Unified prompting. All LLM-based methods (static LLM inference, ICXML, and SCOTTA) share the same prompt template, label verbalization scheme, and output constraints, with identical decoding settings (e.g., temperature and maximum generation length) unless otherwise specified. This controls for prompt engineering effects and isolates algorithmic improvements.

Online test-time adaptation protocol. SCOTTA performs no parameter updates and adapts solely through in-context demonstrations. For each incoming instance, it computes sample confidence using L3R, maintains a bounded Submodular Memory Bank (SMB) with capacity B, and retrieves k nearest exemplars from the current memory as demonstrations for subsequent predictions. Detailed hyperparameter settings are provided in Appendix B.

Handling extreme label spaces. For the StackExchange (SE) dataset with an extremely large label set, we adopt a two-stage pipeline similar to ICXML for all baselines: we first perform coarse candidate retrieval to obtain a manageable label subset, and then conduct LLM-based label generation over the shortlisted candidates. Without this design, most methods are unable to handle extreme multi-label settings effectively.

B Hyperparameter Settings

This appendix details the hyperparameters used in SCOTTA and provides practical heuristics for setting them across datasets. Unless otherwise stated, we tune hyperparameters on a held-out validation split and reuse the same rules for all backbones.

B.1 Key Hyperparameters and Heuristics

SCOTTA contains two online components: (i) confidence-based filtering via L3R, and (ii) bounded cache maintenance via SMB. Accordingly, the key hyperparameters are: memory capacity B, retrieved demonstration number k, confidence threshold τ, L3R smoothing ϵ and weighting sharpness α, and SMB weights $(\lambda_1, \lambda_2, \lambda_3)$.

Memory capacity B. B controls the maximum number of exemplars stored in the Submodular Memory Bank. A larger B improves label coverage and reduces forgetting, but increases retrieval and update cost. We set B to scale with label-space size and long-tail severity:

$$B \propto \min\big(B_{\max},\ c \cdot \sqrt{L}\big), \tag{19}$$

where L is the number of labels and c is a small constant. In practice, we use smaller B for small-/medium label spaces (MOVIE/AAPD/RCV1) and a larger B for extreme multi-label (SE).

Retrieved demonstrations k. k controls the number of exemplars injected into the prompt. We choose k primarily based on the context budget and per-example prompt length. A robust rule is to keep k small (typically 4–8) to avoid prompt dilution and latency overhead. We found k slightly larger helps SE due to higher label sparsity.

Confidence threshold τ. τ determines whether a predicted instance is admitted into the memory. Rather than tuning a dataset-specific absolute value, we adopt a percentile-based heuristic on the validation stream: we set τ to keep approximately the top $p\%$ most confident instances, where p decreases with label-space size (e.g., $p \approx 30\%$ for small label spaces and $p \approx 15\%$ for extreme multi-label). We then convert this to an absolute threshold for each dataset and report the resulting τ.

L3R parameters ϵ and α. We use ϵ only for numerical stability in $\log \frac{p^{\text{self}}+\epsilon}{p^{\text{comp}}+\epsilon}$ and fix it across datasets. α controls how strongly L3R emphasizes high-entropy (competitive) branching positions. We fix α across datasets to avoid overfitting and found performance is stable within a moderate range.

SMB weights $(\lambda_1, \lambda_2, \lambda_3)$. $\lambda_1, \lambda_2, \lambda_3$ balance label coverage, semantic diversity, and sample quality. A simple and effective heuristic is: (i) increase λ_1 for long-tail/extreme label spaces to prevent head-label collapse, (ii) increase λ_2 when examples are semantically repetitive, (iii) keep λ_3 moderate to avoid purely confidence-driven memorization. We normalize weights such that $\lambda_1 + \lambda_2 + \lambda_3 = 1$.

B.2 Dataset-Specific Values

Table 5 summarizes the final hyperparameters used for each dataset. All other settings are shared: cosine similarity for retrieval, k-NN retrieval within the memory, and the same prompting format.

Hyperparameter	MOVIE	AAPD	RCV1	SE
Memory capacity B	64	256	256	1024
Retrieved demos k	4	6	6	8
Confidence threshold τ	0.80	0.72	0.70	0.62
L3R smoothing ϵ	10^{-8}	10^{-8}	10^{-8}	10^{-8}
L3R sharpness α	5	5	5	5
SMB weights $(\lambda_1, \lambda_2, \lambda_3)$	(0.35, 0.35, 0.30)	(0.40, 0.30, 0.30)	(0.40, 0.30, 0.30)	(0.50, 0.25, 0.25)

Table 5: Hyperparameter settings for SCOTTA on four datasets. B scales with label-space size; τ is chosen to retain roughly top-$p\%$ confident instances (higher p for smaller label spaces). Weights satisfy $\lambda_1 + \lambda_2 + \lambda_3 = 1$.

B.3 Two-Stage Inference for StackExchange

For StackExchange (SE), directly generating from the full label space is impractical for many baselines. We therefore adopt a two-stage pipeline similar to prior extreme multi-label generation approaches: (1) a coarse retrieval stage selects a candidate label subset $\mathcal{Y}'$, and (2) the LLM performs constrained multi-label generation over $\mathcal{Y}'$.

Candidate size. We set the candidate size to $|\mathcal{Y}'| = 200$ for SE, which offers a good trade-off between recall and decoding cost. On the validation split, performance is stable for $|\mathcal{Y}'|$ in the range 100–300, with diminishing returns beyond 200.

Retrieval model. We use the same sentence encoder as in SMB (for semantic similarity) to retrieve candidates, and rank labels by cosine similarity between the input and label descriptions (or label names when descriptions are unavailable).

B.4 Shared Decoding and Prompt Budget

All LLM-based methods use the same decoding configuration unless explicitly stated: temperature $= 0$ for deterministic generation, and a maximum generation length of 128 tokens for label-set output. We use a fixed context budget; when the prompt would exceed the budget, we keep at most k demonstrations and truncate older/longer exemplars first. This ensures comparable inference conditions across datasets and methods.

C Prompt Templates

This appendix documents all prompts used in our experiments for LLM-based multi-label text classification. To ensure a fair comparison, **all LLM-based methods** (static LLM inference, ICXML, and SCOTTA) share the same task instruction, label verbalization, and output constraints, unless otherwise specified. SCOTTA differs only in how it *retrieves* in-context demonstrations online and how it *computes confidence* (L3R) and *maintains* a bounded cache (SMB); the prompt surface form remains consistent.

C.1 Notation and Formatting

We use the following placeholders:

- `{DATASET}`: dataset name (MOVIE, AAPD, RCV1, SE).
- `{LABEL_SPACE}`: the label set for small/medium label spaces (MOVIE/AAPD/RCV1), provided as a list.
- `{CANDIDATE_LABELS}`: a reduced candidate set for extreme label space (SE), obtained by a coarse retrieval stage (Section A.0.1).
- `{DEMONSTRATIONS}`: k retrieved exemplars from the current memory bank S_{t-1} (Section 2.5).
- `{INPUT_TEXT}`: the current test instance x_t.

All prompts enforce a **strict output format** to make evaluation deterministic: the model must output only a single line containing a comma-separated label list.

C.2 Shared Task Instruction (All Datasets)

System / role instruction (optional). We use a minimal system instruction to encourage format compliance:

> You are a helpful and precise assistant for multi-label text classification. Follow the output format exactly.

User prompt template.

> Task: Multi-label text classification.
>
> Given an input text, predict all applicable labels from the provided label set. Return labels as a comma-separated list.
>
> Rules: - Only output labels from the provided label set. - Do not output explanations. - Output exactly one line in the form: Labels: label1, label2, ...

C.3 Demonstration Block (Used by SCOTTA and Retrieval-based Baselines)

For methods that use in-context demonstrations (ICXML and SCOTTA; SCOTTA updates them online), we prepend the following block. Each demonstration is formatted as `(input → label-set)`.

Demonstration template.

> Examples:
>
> [Example 1] Text: {EX_TEXT_1} Labels: {EX_LABELS_1}
>
> [Example 2] Text: {EX_TEXT_2} Labels: {EX_LABELS_2}
>
> ...

Query template (appended after demonstrations).

> Now classify the following text:
>
> Text: {INPUT_TEXT} Labels:

C.4 Prompt for MOVIE / AAPD / RCV1 (Full Label Set Provided)

For datasets with small/medium label spaces (MOVIE, AAPD, RCV1), we provide the full label space directly in the prompt.

Full prompt template.

> Task: Multi-label text classification.
>
> Label set ({DATASET}): {LABEL_SPACE}
>
> Rules: - Only output labels from the label set above. - Do not output explanations. - Output exactly one line in the form: Labels: label1, label2, ...
>
> {DEMONSTRATIONS}
>
> Now classify the following text:
>
> Text: {INPUT_TEXT} Labels:

C.5 Prompt for StackExchange (Extreme Label Space via Two-stage Candidate Set)

As described in Section A.0.1, for StackExchange (SE; 12,892 labels), directly enumerating the full label space is impractical. We therefore adopt a two-stage setup for **all** LLM-based baselines and SCOTTA: (i) a coarse retrieval stage proposes a candidate label set, and (ii) the LLM performs constrained label generation over these candidates.

Stage-2 LLM prompt template (candidate-constrained generation).

> Task: Multi-label text classification (extreme label space).
>
> You will be given a candidate label set retrieved for this input. Predict all applicable labels from the candidate set only.
>
> Candidate label set (SE): {CANDIDATE_LABELS}
>
> Rules: - Only output labels from the candidate label set above. - Do not output explanations. - Output exactly one line in the form: Labels: label1, label2, ...
>
> {DEMONSTRATIONS}
>
> Now classify the following text:
>
> Text: {INPUT_TEXT} Labels:

C.6 Static LLM Inference vs. SCOTTA: Prompt Difference Clarification

The *instruction text and formatting constraints* are identical across static LLM inference and SCOTTA. The only difference is the `{DEMONSTRATIONS}` field:

- **Static LLM inference:** `{DEMONSTRATIONS}` is fixed (e.g., a small random or dev-selected set) and does not change over the test stream.
- **SCOTTA:** `{DEMONSTRATIONS}` is retrieved online from the evolving memory bank S_{t-1} maintained by SMB, and candidates are filtered by L3R-based confidence before being inserted into the memory.

C.7 Output Normalization (Evaluation Parsing)

To robustly parse model outputs, we apply lightweight normalization:

- Strip leading/trailing whitespace.
- Require the prefix `Labels:`.
- Split by comma, trim each label string, and discard empty strings.
- Drop any label not in the provided label set / candidate set.

This normalization is applied uniformly to all LLM-based methods.

D Algorithm

Algorithm 1: SCOTTA: Structured Confidence-Guided Online Test-Time Adaptation

Input: Unlabeled test stream $\{x_t\}_{t=1}^{T}$; memory capacity B; retrieved exemplar number k; confidence threshold τ; L3R parameters (α, ϵ); SMB weights $(\lambda_1, \lambda_2, \lambda_3)$.

Output: Predicted label sets $\{\hat{\mathcal{Y}}_t\}_{t=1}^{T}$.

$S \leftarrow \emptyset$; `// Initialize bounded memory bank`

for $t \leftarrow 1$ **to** T **do**

  Retrieve k nearest exemplars D_t from S;

  $\hat{\mathcal{Y}}_t \leftarrow \text{LLMGENERATE}(x_t, D_t)$; `// multi-label generation`

  $C_t \leftarrow \text{L3R}(x_t, \hat{\mathcal{Y}}_t; \alpha, \epsilon)$; `// structured confidence`

  if $C_t \geq \tau$ **then**

    $S \leftarrow \text{SMBUPDATE}(S, (x_t, \hat{\mathcal{Y}}_t, C_t), B; \lambda_1, \lambda_2, \lambda_3)$;

return $\{\hat{\mathcal{Y}}_t\}_{t=1}^{T}$;