

TaxoAgent: Utility-Aligned Policy Optimization for Hierarchical Semantic Classification

Zhiming Lin^{1,†}, Yihao Zhong^{2,†}, Liwei Hou³, Saisai Hu⁴, Shaoyan He⁵,
Rongru Chen¹, Xuan Yang⁶, Siyue Liu^{7,*}

¹Independent Researcher ²New York University ³National University of Singapore

⁴Pace University ⁵Ant Group Co., Ltd. ⁶Hellobike ⁷PhotonPay

[†]Equal contribution. **Correspondence:** Siyue Liu (chenlingbaba@163.com)

Abstract

Hierarchical semantic classification is essential for taxonomy-driven applications, yet existing classifiers often optimize flat label likelihoods and therefore weakly control parent-child validity, semantic error severity, coarse-to-fine coherence, and out-of-scope rejection. This paper aims to align LLM-based classifiers with the full structured utility of hierarchical prediction rather than treating taxonomy labels as independent targets. We propose TAXOAGENT, a utility-aligned policy optimization framework that formulates taxonomy prediction as a verified parent-to-child decision trajectory. Given an input, TAXOAGENT first predicts coarse parent labels, expands them into fine-grained child decisions, and verifies the resulting trajectory with a composite utility that jointly measures outcome correctness, hierarchy consistency, semantic proximity, and trajectory coherence; group-relative policy optimization then reinforces high-utility trajectories over low-utility alternatives. Across IOS-APP-ZH, Google-APP-EN, RCV1, WOS, and CLINC150, TAXOAGENT consistently improves Micro-F1 and Macro-F1, reduces hierarchy inconsistency, strengthens long-tail classification, and improves OOS-F1 by up to 3.68 points over the strongest prior baseline. These results show that treating taxonomies as structured decision spaces offers a practical and extensible path toward more reliable, adaptable, and utility-aware semantic classifiers.

1 Introduction

Hierarchical semantic classification is a fundamental problem in many real-world taxonomy systems, including news categorization, scientific document indexing, intent understanding, and mobile application classification (Golde et al., 2026). Unlike flat classification, it requires models to assign labels that are not only individually correct but also

structurally consistent under parent-child dependencies. This setting is practically important because taxonomy errors often propagate to downstream search, recommendation, moderation, and analytics systems, where confusing a fine-grained category across branches can be substantially more costly than making a local sibling-level mistake. Beyond taxonomy-driven applications, automated classification and risk assessment are also central to security-sensitive settings, including explainable network-traffic anomaly detection and large-scale cloud vulnerability prioritization (Shao, 2026a,b).

Existing research has made substantial progress by incorporating taxonomy information into hierarchical classifiers and text-label representations. Neural HTC models capture parent-child dependencies through local-global prediction (Wehrmann et al., 2018), hierarchy-aware graph modeling (Zhou et al., 2020), information maximization (Deng et al., 2021), label semantic matching (Chen et al., 2021), hierarchy-aware prompting (Wang et al., 2022b), and sub-hierarchy sequence generation (Im et al., 2023). Complementary contrastive methods further shape the representation space by pulling hierarchically related samples or labels closer (Wang et al., 2022a; Zhu et al., 2024; Guo et al., 2025). While these methods confirm the value of taxonomic structure and semantic label relations, they mainly treat hierarchy as an auxiliary representation, architectural bias, decoding format, or contrastive signal. Consequently, their objectives remain only partially aligned with the real utility of hierarchical prediction, where label correctness, parent-child validity, semantic error severity, and coarse-to-fine decision consistency must be evaluated jointly.

Large language models offer a promising alternative because they can interpret label names, descriptions, and task instructions using broad seman-

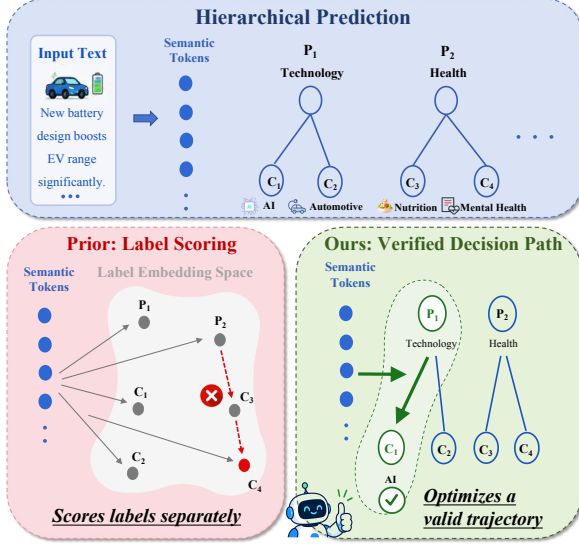


Figure 1: Prior methods score labels independently in the embedding space, which can produce semantically plausible but hierarchy-invalid predictions. In contrast, TAXOAGENT formulates hierarchical classification as a verified parent-to-child decision path, optimizing a structurally valid and semantically coherent trajectory.

tic knowledge. Recent LLM-based classification methods enrich taxonomies with generated class-indicative features (Zhang et al., 2025b) or improve zero-shot hierarchical classification through prompt ensembling (Xia et al., 2025). However, prompting alone provides limited control over hierarchy consistency, fine-grained decision calibration, and rejection behavior. This limitation is especially visible in open-set or out-of-scope scenarios, where the model must decide not only which known label to assign but also whether the input belongs to the taxonomy at all. Existing OOS intent detection methods improve rejection boundaries through benchmark design, boundary augmentation, contrastive feedback, or small-large model collaboration (Larson et al., 2019; Sahu et al., 2023; Zhang et al., 2025a; Chen and Wang, 2025), but they are usually studied outside hierarchical taxonomies and do not directly address the coupling between rejection and parent-child validity.

These observations reveal a central gap: hierarchical semantic classification should be viewed not merely as label scoring, representation learning, or prompt-based prediction, but as a structured decision problem whose utility is non-decomposable. The core challenge is to align model behavior with the actual desiderata of taxonomy prediction: selecting semantically appropriate labels, respecting hierarchy constraints, maintaining coarse-to-fine

decision coherence, and rejecting inputs that fall outside the known taxonomy. This paper therefore asks: *How can an LLM-based classifier be aligned with the full structured utility of hierarchical semantic classification rather than with isolated label likelihoods?*

To answer this question, we propose TAXOAGENT, a utility-aligned policy optimization framework for hierarchical semantic classification. As shown in Fig. 1, TAXOAGENT converts taxonomy prediction into an explicit coarse-to-fine decision process and optimizes the model according to verifiable structured utility rather than ordinary flat supervision. By aligning the policy with both semantic and structural aspects of taxonomy prediction, TAXOAGENT improves not only exact classification accuracy but also hierarchy consistency, long-tail robustness, and out-of-scope rejection.

Our contributions are summarized as follows:

(i) We identify a mismatch between conventional HTC objectives and the true utility of taxonomy prediction, where correctness, hierarchy validity, semantic severity, and decision coherence must be evaluated jointly.

(ii) We introduce TAXOAGENT, which formulates hierarchical prediction as a coarse-to-fine decision process and optimizes the policy using verifiable utility signals tailored to hierarchical semantic classification.

(iii) Experiments on app-domain taxonomies, standard HTC benchmarks, and OOS intent detection show that TAXOAGENT consistently improves Micro-F1, Macro-F1, hierarchy consistency, long-tail label performance, and out-of-scope rejection over strong hierarchical, contrastive, and LLM-based baselines.

2 Related Work

Taxonomy-constrained hierarchical classification. Hierarchical text classification predicts labels under parent-child dependencies, with RCV1 and Web of Science as widely used benchmarks (Lewis et al., 2004; Kowsari et al., 2017) and pretrained encoders such as BERT providing strong representations (Devlin et al., 2019). Prior methods incorporate hierarchy through architectures or objectives, including local-global classifiers (Wehrmann et al., 2018), hierarchy-aware graph modeling (Zhou et al., 2020), text-label information maximization (Deng et al., 2021), and label-semantic matching (Chen et al., 2021). Recent work further ex-

plores hierarchy-aware prompting, sub-hierarchy generation, and tree-isomorphism encoding (Wang et al., 2022b; Im et al., 2023; Zhu et al., 2023). Although these methods improve structural awareness, they mostly treat taxonomies as auxiliary representations, decoding constraints, or supervised regularizers. In contrast, TAXOAGENT treats the taxonomy as the decision space itself and optimizes a parent-to-child trajectory with a utility that jointly captures correctness, hierarchy validity, semantic alignment, and decision coherence.

Hierarchy-aware semantic representation learning. A complementary line of work improves HTC by shaping the semantic geometry of texts and labels (Xue and Gao, 2026). HGCLR injects hierarchy into contrastive text encoding (Wang et al., 2022a), HILL constructs hierarchy-preserving positives (Zhu et al., 2024), HMCL extends hierarchy-aware contrastive learning to mobile app classification (Guo et al., 2025), and HBM introduces a hierarchy-aware margin loss to encode label relations (Kim et al., 2024). Recent adjacent work also studies long-tailed multi-label prediction and hierarchical, contrastive data valuation (Xiao et al., 2026b,a), complementing label-geometry methods with imbalance- and utility-aware training signals. These methods learn hierarchy-sensitive embeddings, especially for fine-grained labels, but still mainly optimize separability or classification likelihood and do not explicitly distinguish mild semantic near-misses from severe cross-branch errors. TAXOAGENT addresses this limitation by using semantic prototypes as both verifiers and reward signals, encouraging predictions that are correct, semantically local, and parent-consistent.

LLM-based classification and open-set rejection. Large language models can leverage label names, descriptions, and instructions for classification without task-specific architectures. Related work on structured generation has developed autonomy-oriented taxonomies for agentic Text-to-SQL, decomposed reasoning for interpretable, robust prediction, and human-feedback integration for unified layout generation (Su et al., 2026; Zhao et al., 2026; Guan et al., 2026; Lu et al., 2025a). Graph-aligned enrichment and logic-consistent structured knowledge reasoning further connect LLM predictions to explicit relational constraints (Li et al., 2025b,a). Related cross-domain work explores LLM-guided semantic fusion for asynchronous time-series forecasting, structured attention for

medical-image segmentation, and controlled diagnosis of video temporal-logical reasoning (Zhang et al., 2026a,b; Kwan et al., 2026). Adjacent work on LLM agents further studies structured agent distillation, role-aware context routing with structured memory, and persistent verifier signals for memory governance (Liu et al., 2026b, 2025; Wang et al., 2026). Prototype-based OOD detection and retrieval-diverse multi-turn intent understanding provide related evidence that semantic geometry and context selection matter for robust classification (Peng et al., 2026; Lin et al., 2026). Prototype-aligned soft prompts and task-conditioned LoRA further address continual adaptation under non-stationarity (Xiao and Hou, 2026; Xiao et al., 2026c). Complementary efficiency work prunes sparse mixture-of-experts LLMs without retraining (Zhou et al., 2025). Across adjacent vision and robotics tasks, semantic side information has similarly been used through language anchors for road anomaly detection (Tian et al., 2023), noun prototypes for task-oriented instance segmentation (Li et al., 2022), and LLM-derived context for motion prediction (Zheng et al., 2024). Fine-grained visual recognition has likewise used LLM-derived semantic homology to distinguish closely related categories (Liu et al., 2026a). Related 3D work studies collaborative reasoning and elastic semantic anchoring for unified understanding and generation (Yu et al., 2025b, 2026c). Adjacent VLA studies explore informative-frame selection and asymmetric generalist-VLM adaptation for embodied tasks (Yu et al., 2026a,b). For HTC, TELEClass and HiEPS improve taxonomy use through LLM-generated features and prompt ensembling (Zhang et al., 2025b; Xia et al., 2025), but prompting lacks a principled objective for hierarchy consistency and calibrated rejection. Complementary calibration work preserves predictive uncertainty in quantized language models through target-aware data selection (Yang et al., 2026). Distribution-optimized data synthesis provides another route to expanding LLM knowledge boundaries (Li et al., 2026). OOS intent detection methods improve robustness, boundary learning, and intent separation on benchmarks such as CLINC150 (Larson et al., 2019; Zhang et al., 2022; Sahu et al., 2023; Zhang et al., 2025a; Chen and Wang, 2025; Zawbaa et al., 2024), alongside broader task-oriented OOD and anomaly-detection advances (Lu et al., 2025b; Zhang et al., 2026c), yet this literature does not couple rejection with parent-child valid-

ity. Related post-training work learns hierarchical temperature policies from LLM internal states and elicits inference-time reasoning through reinforcement query refinement (Zhou et al., 2026b,a). Building on work in supervised fine-tuning, reward-based post-training, fine-grained preference optimization, progress-aware policy learning, evidence-anchored multimodal RLVR, and latent trajectory selection (Ouyang et al., 2022; Shao et al., 2024; Xue et al., 2026b,a; Shen et al., 2025; Liu et al., 2026c; Hu et al., 2026; Yu et al., 2025a), TAXOAGENT optimizes verifiable taxonomy-specific utilities for structured prediction and OOS rejection.

3 Method

We propose TAXOAGENT, a utility-aligned policy optimization framework for hierarchical semantic classification. The central idea is to treat hierarchical prediction as a structured decision problem over a taxonomy graph rather than as independent label scoring. Given an input text, the policy generates a coarse-to-fine decision trajectory: it first predicts parent labels, then predicts child labels conditioned on the parent decision, and finally verifies the structured output. The trajectory is scored by a non-decomposable utility that jointly evaluates label correctness, hierarchy validity, semantic alignment, and decision coherence. The LLM policy is optimized with group-relative policy optimization so that high-utility trajectories receive higher likelihood than low-utility trajectories sampled for the same input.

Figure 2 illustrates the TaxoAgent pipeline. The taxonomy graph defines the structural decision space. The LLM policy produces a multi-step hierarchical decision trajectory. A structural verifier evaluates parent-child validity, while a semantic verifier evaluates text-label and label-label alignment. The resulting utility is used as the reward for policy optimization.

3.1 Hierarchical Classification as Structured Decision Making

Let the taxonomy be a rooted hierarchical graph $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of taxonomy labels and $\mathcal{E}$ contains directed parent-child edges. Each label $\ell \in \mathcal{V}$ has a depth $d(\ell)$, and $\rho(\ell)$ denotes its immediate parent when ℓ is not a root-level label. For a label ℓ , let $\text{Anc}(\ell)$ denote the set of all required ancestors of ℓ in the taxonomy. The two-level case used by the app-domain taxonomies

is a special case where $\mathcal{V} = P \cup C$ and $\mathcal{E} \subseteq P \times C$.

Given an input text x , the ground-truth hierarchical assignment is $Y \subseteq \mathcal{V}$. A prediction $\tilde{Y} \subseteq \mathcal{V}$ is hierarchy-consistent if every predicted non-root label includes all of its required ancestors:

$$\forall \ell \in \tilde{Y}, \quad \text{Anc}(\ell) \subseteq \tilde{Y}. \quad (1)$$

For two-level taxonomies, this reduces to the parent-child constraint $\forall c \in \tilde{Y}^c, \rho(c) \in \tilde{Y}^p$.

The policy is trained on raw hierarchical predictions before structural projection. This allows hierarchy violations to be observed and penalized during training. At inference time, we apply a deterministic closure projection that adds missing ancestors and removes labels outside the taxonomy vocabulary, as described in Appendix A.3.

3.2 Hierarchical Decision Policy

Hierarchical classification requires coupled parent and child decisions. A one-step classifier hides this dependency and treats the final label set as a flat output. TaxoAgent instead makes the decision path explicit: the policy first selects parent labels, then selects child labels under the induced semantic context, and finally verifies the structured prediction.

Formulation. We parameterize the classifier as an LLM policy π_θ . For input x and taxonomy $\mathcal{G}$, the policy generates a trajectory $\tau = (a^p, a^c, a^v)$, where a^p is the parent decision, a^c is the child decision, and a^v is the verification decision. The trajectory probability is factorized as

$$\pi_\theta(\tau \mid x, \mathcal{G}) = \pi_\theta(a^p \mid x, \mathcal{G}) \cdot \pi_\theta(a^c \mid x, \mathcal{G}, a^p) \cdot \pi_\theta(a^v \mid x, \mathcal{G}, a^p, a^c). \quad (2)$$

The final raw prediction $\tilde{Y} = (\tilde{Y}^p, \tilde{Y}^c)$ is parsed from a^v . The exact output schema and decoding format are given in Appendix A.1.

3.3 Semantic Prototypes and Verifier

Not all classification errors have the same severity. A prediction close to the correct category in the taxonomy is less harmful than a cross-branch prediction with unrelated semantics. TaxoAgent therefore uses a semantic verifier to assign graded rewards based on label semantics.

Prototype space. Let f_ϕ be a frozen text encoder. For input x and label $\ell \in \mathcal{P} \cup \mathcal{C}$, we define normalized representations

$$h_x = \frac{f_\phi(x)}{\|f_\phi(x)\|_2}, \quad e_\ell = \frac{q_\ell}{\|q_\ell\|_2}, \quad (3)$$

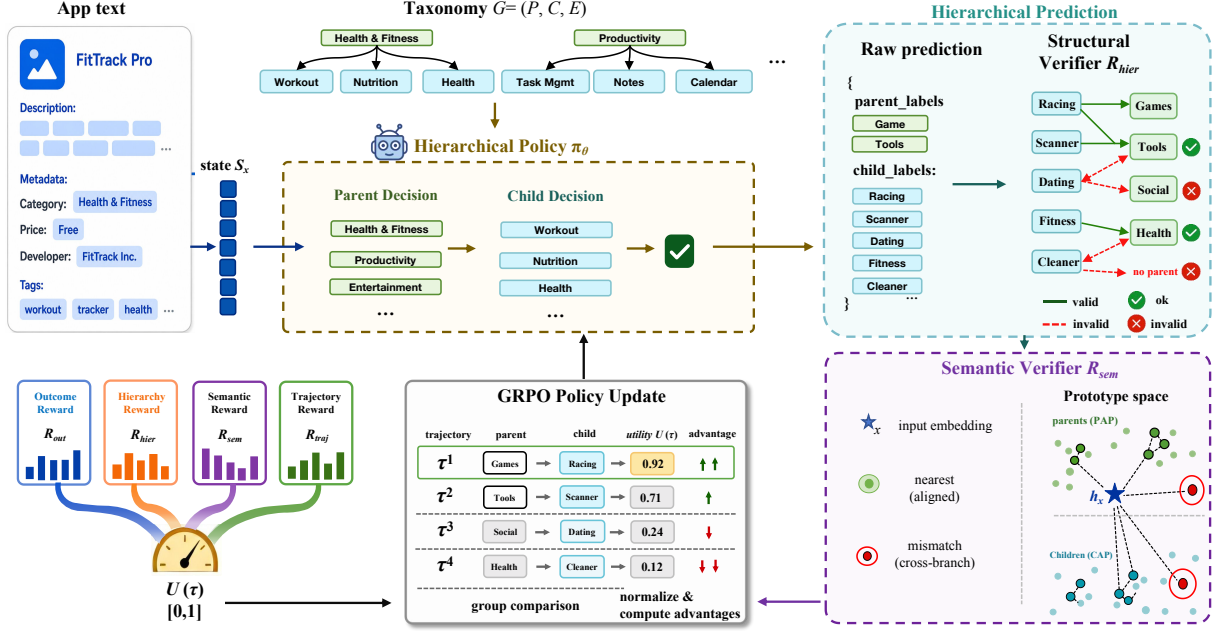


Figure 2: Pipeline of TAXOAGENT illustrated with an app-text example. Given app text and a taxonomy graph, the policy generates a coarse-to-fine parent–child prediction trajectory. The raw prediction is evaluated by a structural verifier for taxonomy validity and a semantic verifier for prototype-space alignment. Outcome, hierarchy, semantic, and trajectory rewards are combined into a utility score, which is used by GRPO to reinforce high-utility hierarchical predictions.

where h_x is the text embedding, e_ℓ is the label prototype, and q_ℓ is the unnormalized prototype constructed from label names and training instances. The construction of q_ℓ is specified in Appendix A.2.

We distinguish text–label similarity from label–label similarity:

$$s_x(\ell) = \frac{1 + h_x^\top e_\ell}{2}, \quad s_{\ell, \ell'} = \frac{1 + e_\ell^\top e_{\ell'}}{2}. \quad (4)$$

Both scores lie in $[0, 1]$ because the embeddings are normalized. The semantic verifier uses $s_{\ell, \ell'}$ during training to compare predicted labels against ground-truth labels, and uses $s_x(\ell)$ during inference to score text–label compatibility.

3.4 Utility-Aligned Reward

Purpose. Hierarchical decision quality is non-decomposable. A prediction is useful only when it is correct, structurally valid, semantically plausible, and coherent across the parent-to-child decision path. TaxoAgent therefore optimizes a trajectory-level utility rather than independent label losses.

Outcome reward. Let $F_1(A, B) = 2|A \cap B|/(|A| + |B| + \epsilon)$ be the set F1 score. Let $\text{Pair}(S) = \{(\rho(c), c) : c \in S\}$ map a child set

to its parent–child pairs. The outcome reward is

$$R_{\text{out}}(\tilde{Y}, Y) = \omega_p F_1(\tilde{Y}^p, Y^p) + \omega_c F_1(\tilde{Y}^c, Y^c) + \omega_{pc} F_1(\text{Pair}(\tilde{Y}^c), \text{Pair}(Y^c)), \quad (5)$$

where $\omega_p, \omega_c, \omega_{pc} \geq 0$ and $\omega_p + \omega_c + \omega_{pc} = 1$. The first two terms evaluate parent and child accuracy, while the third term evaluates correct parent–child pair assignment.

Hierarchy reward. The structural verifier evaluates whether each predicted child is supported by its predicted parent set:

$$R_{\text{hier}}(\tilde{Y}; \mathcal{G}) = 1 - \frac{\sum_{\tilde{c} \in \tilde{Y}^c} \mathbb{I}[\rho(\tilde{c}) \notin \tilde{Y}^p]}{|\tilde{Y}^c| + \epsilon}. \quad (6)$$

This reward is computed on the raw prediction before structural projection.

Semantic reward. For a predicted label ℓ and a target label set B , define the semantic distance as $d(\ell, B) = 1 - \max_{b \in B} s_{\ell, b}$. The parent-aware and child-aware semantic penalties are

$$D_p = \frac{1}{|\tilde{Y}^p| + \epsilon} \sum_{\tilde{p} \in \tilde{Y}^p} d(\tilde{p}, Y^p),$$

$$D_c = \frac{1}{|\tilde{Y}^c| + \epsilon} \sum_{\tilde{c} \in \tilde{Y}^c} \left(1 + \kappa \mathbb{I}[\rho(\tilde{c}) \notin Y^p]\right) \cdot d(\tilde{c}, Y^c). \quad (7)$$

Here, D_p measures coarse-level semantic mismatch, D_c measures fine-grained mismatch, and $\kappa \geq 0$ increases the penalty for cross-parent child errors. The semantic reward is

$$R_{\text{sem}}(\tilde{Y}, Y) = \exp\left(-\frac{\alpha D_p + (1 - \alpha) D_c}{\tau_{\text{sem}}}\right), \quad (8)$$

where $\alpha \in [0, 1]$ balances parent-level and child-level penalties, and $\tau_{\text{sem}} > 0$ is the semantic temperature.

Trajectory coherence reward. Let $\tilde{Y}_{\text{step}}^p$ be the parent set emitted in the first decision step, and let $\tilde{Y}^c$ be the child set in the final raw prediction. The trajectory coherence reward is

$$R_{\text{traj}}(\tau; \mathcal{G}) = \frac{1}{|\tilde{Y}^c| + \epsilon} \sum_{\tilde{c} \in \tilde{Y}^c} \mathbb{I}[\rho(\tilde{c}) \in \tilde{Y}_{\text{step}}^p]. \quad (9)$$

This term rewards child decisions that remain supported by the earlier parent decision, and therefore measures coarse-to-fine decision consistency rather than final-label validity alone.

Composite utility. The final trajectory utility is

$$U(\tau; x, Y, \mathcal{G}) = \lambda_{\text{out}} R_{\text{out}} + \lambda_{\text{hier}} R_{\text{hier}} + \lambda_{\text{sem}} R_{\text{sem}} + \lambda_{\text{traj}} R_{\text{traj}}, \quad (10)$$

where all λ coefficients are non-negative and sum to one. All reward components lie in $[0, 1]$, so $U(\tau; x, Y, \mathcal{G}) \in [0, 1]$.

3.5 Group-Relative Policy Optimization

Supervised fine-tuning increases the likelihood of reference labels but does not directly optimize hierarchy validity, semantic distance, and trajectory coherence. TaxoAgent instead samples multiple trajectories for the same input and updates the policy according to their relative utilities.

Optimization. For each training instance (x_i, Y_i) , we sample G trajectories $\{\tau_i^g\}_{g=1}^G$ from the old policy $\pi_{\theta_{\text{old}}}$. Each trajectory receives reward $r_i^g = U(\tau_i^g; x_i, Y_i, \mathcal{G})$. The group-relative advantage and clipped objective are

$$A_i^g = \frac{r_i^g - \mu_i}{\sigma_i + \epsilon}, \quad \mu_i = \frac{1}{G} \sum_{g=1}^G r_i^g, \quad \sigma_i^2 = \frac{1}{G} \sum_{g=1}^G (r_i^g - \mu_i)^2. \quad (11)$$

$$\mathcal{J}_{\text{GRPO}}(\theta) = \frac{1}{\sum_{i,g} T_{i,g}} \times \sum_{i,g,t} \min(r_{i,g,t} A_i^g, \bar{r}_{i,g,t} A_i^g) - \beta D_{\text{KL}}\left(\pi_{\theta}(\cdot | x_i, \mathcal{G}) \parallel \pi_{\text{ref}}(\cdot | x_i, \mathcal{G})\right),$$

$$\text{where } \bar{r}_{i,g,t} = \text{clip}(r_{i,g,t}(\theta), 1 - \epsilon, 1 + \epsilon). \quad (12)$$

Here, $T_{i,g}$ is the length of trajectory τ_i^g , $r_{i,g,t}(\theta)$ is the token-level probability ratio between π_{θ} and $\pi_{\theta_{\text{old}}}$ at step t , ϵ is the clipping coefficient, β controls KL regularization, and π_{ref} is the reference policy. The policy parameters are updated by maximizing $\mathcal{J}_{\text{GRPO}}(\theta)$. The full training algorithm is provided in Appendix A.6.

Inference At inference time, TaxoAgent decodes one hierarchical trajectory, parses the raw prediction $\tilde{Y}$, and applies the deterministic structural projection $\Pi_{\mathcal{G}}$ defined in Appendix A.3. The final prediction is $\hat{Y} = \Pi_{\mathcal{G}}(\tilde{Y})$. This projection guarantees that every predicted child label is supported by its parent. The out-of-scope decision rule used in OOS experiments is described in Appendix A.4. Taxonomy expansion is handled by updating label prototypes and the taxonomy edge set, as described in Appendix A.5.

4 Experiments

We evaluate TAXOAGENT from four complementary perspectives: **(i)** whether utility-aligned optimization improves hierarchical classification on both app-domain and standard HTC benchmarks; **(ii)** whether TAXOAGENT remains robust under taxonomy expansion and out-of-scope rejection; **(iii)** which reward and trajectory components drive the final gains; and **(iv)** whether multi-step parent-child reasoning improves hierarchy consistency, semantic error quality, and long-tail performance.

4.1 Experimental Setup

Datasets. We evaluate TAXOAGENT on five datasets spanning industrial app categorization, standard hierarchical text classification, and out-of-scope (OOS) intent detection. IOS-APP-ZH and Google-APP-EN are two manually annotated app

datasets with two-level taxonomies. RCV1 and Web of Science (WOS) are standard HTC benchmarks, while CLINC150 is used for hierarchical intent classification and OOS rejection. Dataset statistics are summarized in Table 4; detailed dataset construction and split protocols are provided in Appendix B.1.

Evaluation metrics. For hierarchical classification, we report Micro-F1 and Macro-F1. For OOS detection, we report OOS-F1. We also report hierarchy inconsistency rate (HIR), which measures whether predicted child labels are unsupported by their predicted parents. Lower HIR indicates better structural compliance. Formal metric definitions are given in Appendix B.2.

Baselines. We compare TAXOAGENT with encoder-based HTC models, hierarchy-aware contrastive methods, LLM-enhanced taxonomy and prompting methods, direct LLM prompting baselines, and representative OOS intent detection methods. Unless otherwise stated, TAXOAGENT uses Qwen3-14B as the base policy model (Yang et al., 2025). The full baseline list and implementation details are provided in Appendix B.3. Unless otherwise specified, all F1 and accuracy values are reported in percentage.

4.2 Main Results

Table 1 shows that TAXOAGENT achieves the best results on all datasets and metrics. It improves over the strongest prior methods by +2.49/ +2.55, +0.66/ +1.69, +1.46/ +1.79, and +0.76/ +1.98 points in Micro-F1/Macro-F1 on IOS-APP-ZH, Google-APP-EN, RCV1, and WOS, respectively. The larger and consistent Macro-F1 gains indicate better robustness to rare and fine-grained hierarchical labels. Moreover, TAXOAGENT clearly outperforms direct LLM prompting on the app-domain datasets, suggesting that utility-aligned policy optimization is more effective than direct prompting for structured hierarchical classification.

Adaptation to Evolving Taxonomies. Real-world taxonomies continually evolve as new parent categories and fine-grained child labels emerge. We therefore test whether TAXOAGENT can adapt with minimal or no retraining by expanding each app taxonomy with one new parent category and multiple child labels, using over 300 domain-specific apps, and comparing against direct LLM baselines. Table 2 shows that TAXOAGENT achieves the best parent- and child-level accuracy on both datasets,

with especially large gains at the fine-grained child level. This suggests that separating taxonomy structure, semantic prototypes, and policy optimization enables effective adaptation to new hierarchical labels without full retraining.

Out-of-Scope Detection. We further evaluate TAXOAGENT on CLINC150 to test whether it can reject semantically related OOS intents while maintaining in-scope accuracy. Following a domain-level in-domain OOS setting, we use the *Banking* and *Credit Cards* domains, treating part of each domain’s intents as in-scope and the held-out intents as OOS examples; details are provided in Appendix B.4. As shown in Table 3, TAXOAGENT achieves the best OOS performance, improving over FCSLM by +2.17 ACC, +3.64 OOS-Banking-F1, and +3.68 OOS-CreditCards-F1. These gains show that the semantic verifier and structured trajectory better separate fine-grained in-scope intents from related but out-of-scope ones.

4.3 Ablation Studies

Figure 3 summarizes the main ablations on reward design, trajectory reasoning, joint optimization, and hierarchy consistency. Overall, each component contributes to the final performance: utility-aligned RL improves over the raw Qwen3-14B baseline, while parent-aware and child-aware semantic penalties further boost Micro-F1 and Macro-F1. Replacing the full coarse-to-fine trajectory with single-step decoding reduces accuracy and increases hierarchy violations, showing that parent selection, child expansion, and verification directly improve structured decision quality. Joint parent-child optimization also outperforms independent optimization, confirming the non-decomposable nature of hierarchical utility. Finally, TAXOAGENT achieves the lowest HIR among LLM baselines, indicating that explicit hierarchy-aware utility optimization is more reliable than direct prompting or post-hoc correction alone.

4.4 Semantic Error Distance Analysis

Semantic error quality. A stronger hierarchical classifier should make not only fewer errors but also milder errors, since semantically local confusions are less harmful than cross-branch mistakes. We compute the semantic error distance of false-positive labels on IOS-APP-ZH and Google-APP-EN and visualize the resulting distributions with ridgeline plots; implementation details are provided in Appendix B.5. As shown in Figure 4,

Table 1: Main hierarchical classification results on four datasets. All values are percentages. Best results are shown in bold, and second-best results are underlined.

Model	IOS-APP-ZH		Google-APP-EN		RCV1		WOS	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
<i>Encoder-based and hierarchical neural models</i>								
BERT (Devlin et al., 2019)	43.41	23.94	77.04	75.44	86.26	67.35	86.26	80.58
HiAGM (Zhou et al., 2020)	—	—	—	—	85.58	67.93	86.04	80.19
HTCInfoMax (Deng et al., 2021)	69.79	26.18	81.28	56.37	85.53	67.09	86.30	79.97
HiMatch (Chen et al., 2021)	—	—	—	—	86.33	68.66	86.70	81.06
HMCN (Wehrmann et al., 2018)	—	—	—	—	87.52	70.39	86.45	80.91
HMCN + HMCL (Wehrmann et al., 2018; Guo et al., 2025)	—	—	—	—	<u>87.92</u>	<u>71.36</u>	86.90	81.07
<i>Hierarchy-aware contrastive learning methods</i>								
HGCLR (Wang et al., 2022a)	42.32	25.88	60.73	55.07	86.49	68.31	87.11	81.20
HGCLR (title+abstract) (Wang et al., 2022a)	45.99	28.53	61.90	56.93	87.22	69.89	—	—
HILL (Zhu et al., 2024)	71.02	21.74	80.87	53.53	87.31	70.12	87.28	<u>81.77</u>
HILL (title+abstract) (Zhu et al., 2024)	<u>74.86</u>	23.54	<u>81.84</u>	54.78	87.70	70.96	—	—
<i>LLM-enhanced taxonomy and prompting methods</i>								
TELEClass (Zhang et al., 2025b)	74.37	28.45	79.91	<u>79.01</u>	—	—	<u>89.17</u>	80.91
HiEPS (ChatGPT-3.5) (Xia et al., 2025)	41.89	27.66	76.33	75.01	—	—	65.41	55.11
TaxMorph (Golde et al., 2026)	74.19	26.89	74.03	76.02	—	—	87.04	80.20
<i>Direct LLM prompting methods</i>								
ChatGPT-4o (OpenAI, 2024)	37.42	27.12	73.40	70.20	—	—	79.03	71.95
DeepSeek-3.2 (DeepSeek-AI, 2025)	39.55	<u>32.66</u>	75.80	73.10	—	—	81.02	72.92
TAXOAGENT	77.35	35.21	82.50	80.70	89.38	73.15	89.93	83.75
Δ over best prior	↑ 2.49	↑ 2.55	↑ 0.66	↑ 1.69	↑ 1.46	↑ 1.79	↑ 0.76	↑ 1.98

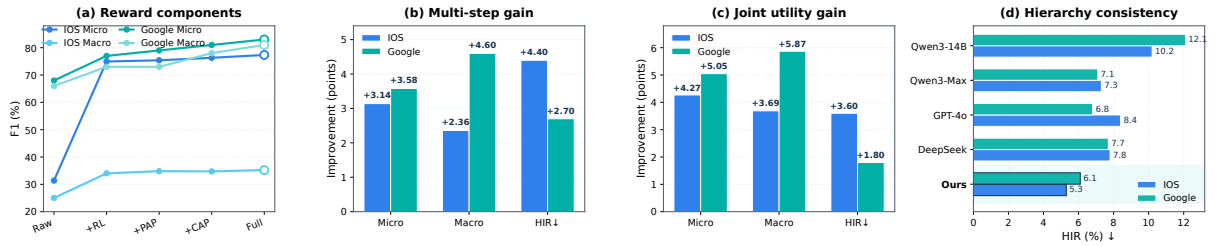


Figure 3: Ablation analysis of TAXOAGENT on IOS-APP-ZH and Google-APP-EN. (a) Reward component ablation shows the cumulative benefit of utility-aligned RL and semantic penalties. (b) Multi-step trajectory reasoning improves both prediction accuracy and hierarchy consistency over single-step decoding. (c) Joint parent–child optimization outperforms independent optimization, validating the non-decomposable nature of hierarchical utility. (d) TAXOAGENT achieves the lowest hierarchy inconsistency rate among LLM baselines.

Table 2: Performance under evolving taxonomy scenarios. We report parent-level and child-level accuracy on newly expanded app taxonomies. Best results are shown in bold, and second-best results are underlined.

Model	IOS-APP-ZH		Google-APP-EN	
	Parent Acc.	Child Acc.	Parent Acc.	Child Acc.
<i>Direct LLM baselines</i>				
Qwen3-14B	92.2	81.4	95.2	86.4
Qwen3-Max	95.1	84.9	97.2	86.90
ChatGPT-4o	95.5	82.1	<u>98.5</u>	<u>89.1</u>
DeepSeek-3.2	<u>95.70</u>	<u>85.7</u>	97.4	87.3
TAXOAGENT	98.3	94.1	98.9	95.7
Δ over best prior	↑ 2.6	↑ 8.4	↑ 0.4	↑ 6.6

TAXOAGENT produces the strongest left-shifted distributions, indicating that its wrong predictions remain closer to the correct label semantics. On IOS-APP-ZH, TAXOAGENT reduces the mean er-

ror distance from 0.266 to 0.148 and increases the near-miss rate from 51.6% to 82.0% compared with the strongest prior baseline. On Google-APP-EN, it reduces the mean error distance from 0.213 to 0.120 and increases the near-miss rate from 65.9% to 88.2%. These results suggest that the proposed utility not only improves exact matching but also reshapes severe cross-branch errors into semantically local confusions, explaining the gains in Macro-F1 and hierarchy consistency.

4.5 Long-tail Label Gain Analysis

Long-tail performance. Macro-F1 gains are meaningful only if they come from rare and fine-grained labels rather than marginal improvements on frequent labels. We compute the per-label F1 gain of TAXOAGENT over the strongest prior base-

Table 3: OOS detection results on CLINC150. ACC denotes in-scope classification accuracy. OOS-F1 evaluates rejection of held-out in-domain OOS intents. Best results are shown in bold, and second-best results are underlined.

Model	ACC	OOS-Banking-F1	OOS-CreditCards-F1
<i>Fine-tuned and open-set baselines</i>			
RoBERTa	22.29	—	—
PromptMix	74.27	—	—
CUC	75.63	—	—
FCSLM	<u>90.18</u>	<u>84.29</u>	<u>82.71</u>
<i>Direct LLM baseline</i>			
Qwen3	58.92	75.82	61.92
TAXOAGENT	92.35	87.93	86.39
Δ over best prior	$\uparrow 2.17$	$\uparrow 3.64$	$\uparrow 3.68$

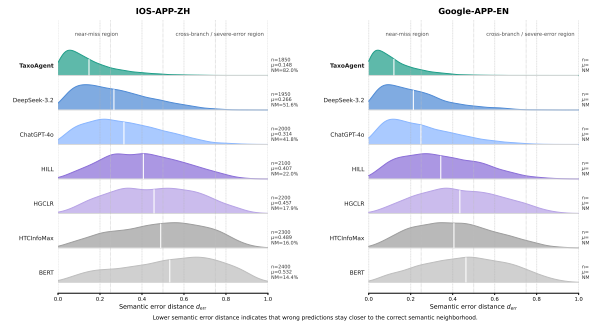


Figure 4: Semantic error distance on IOS-APP-ZH and Google-APP-EN. Lower values indicate milder mistakes. TAXOAGENT yields the lowest mean distance and highest near-miss rate, showing that its errors are more semantically local. Protocol details are provided in Appendix B.5.

line and group child labels by training frequency; details are provided in Appendix B.7. As shown in Figure 5, most high-gain labels are concentrated in the low-frequency region. On IOS-APP-ZH, tail labels improve by +8.7 F1 points on average, compared with +4.1 for medium-frequency labels and +1.6 for head labels; on Google-APP-EN, the corresponding gains are +7.9, +3.6, and +1.2. This confirms that TAXOAGENT’s Macro-F1 improvement is primarily driven by better recognition of rare labels, suggesting that semantic prototypes and hierarchy-aware rewards transfer useful signals from taxonomy structure to underrepresented categories.

5 Conclusion

This paper addressed the mismatch between conventional hierarchical classification objectives and the structured utility required by real taxonomy prediction. We introduced TAXOAGENT, a utility-

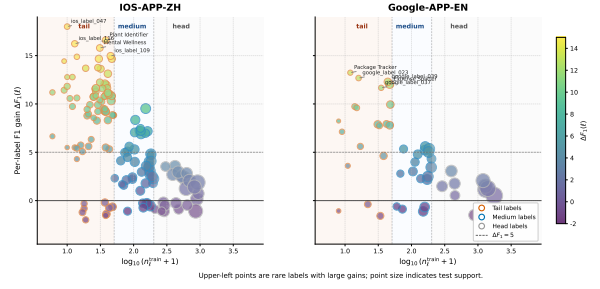


Figure 5: Long-tail label gain analysis on IOS-APP-ZH and Google-APP-EN. Each point denotes a child label, with the x -axis showing training frequency and the y -axis showing F1 gain over the strongest prior baseline. TAXOAGENT obtains the largest gains on rare labels, indicating stronger long-tail classification. Protocol details are provided in Appendix B.7.

aligned policy optimization framework that formulates hierarchical semantic classification as a verified coarse-to-fine decision process and optimizes LLM policies with signals that jointly capture correctness, hierarchy validity, semantic locality, and trajectory coherence. Across app-domain taxonomies, standard HTC benchmarks, evolving-taxonomy settings, and OOS intent detection, our results show that utility-aligned optimization improves not only predictive accuracy but also the reliability and quality of hierarchical decisions. These findings suggest that hierarchical classifiers should be trained and evaluated as structured decision systems rather than as flat label predictors augmented with taxonomy constraints.

Limitations

This work focuses on hierarchical semantic classification settings where the taxonomy structure and label descriptions are available during training and inference. Although this assumption matches practical taxonomy-driven applications, future work may study settings where taxonomies are partially observed, noisy, or updated continuously by external users. Our experiments cover app-domain taxonomies, standard hierarchical text classification benchmarks, and intent classification with out-of-scope detection, but the framework has not yet been evaluated on all taxonomy domains, such as biomedical coding, legal classification, or multi-lingual industrial taxonomies. Extending the evaluation to these domains would provide a broader understanding of how utility-aligned hierarchical decision making behaves under different label granularities and annotation conventions.

References

- Haibin Chen, Qianli Ma, Zhenxi Lin, and Jiangyue Yan. 2021. Hierarchy-aware label semantics matching network for hierarchical 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 4370–4379.
- Peng Chen and Bang Wang. 2025. On collaborating small and large models for few-shot intent detection. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 13937–13953.
- DeepSeek-AI. 2025. DeepSeek-V3.2: Pushing the Frontier of Open Large Language Models. *arXiv preprint arXiv:2512.02556*.
- Zhongfen Deng, Hao Peng, Dongxiao He, Jianxin Li, and Philip S Yu. 2021. Htcinfomax: A global model for hierarchical text classification via information maximization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3259–3265.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.
- Jonas Golde, Nicolaas Paul Jedema, RaviKiran Krishnan, and Phong Le. 2026. Hierarchical text classification with llm-refined taxonomies. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 214–228.
- Shixing Guan, Xuan Hua, Jiahao Lin, and Luyan Zhang. 2026. Internalizing Decomposed Reasoning: Reverse-Distilled Fine-Tuning for Interpretable and Robust Text-to-SQL. In *International Conference on Intelligent Computing*, pages 534–544. Springer.
- Jiawei Guo, Yang Xiao, Weipeng Huang, and Guangyuan Piao. 2025. Enhancing text-based hierarchical multilabel classification for mobile applications via contrastive learning. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 4476–4486.
- Ruina Hu, Chen Wang, Lai Wei, Jionghao Bai, Bin Yu, Weiran Huang, Kai Wang, and Yue Wang. 2026. *Attend to Evidence: Evidence-Anchored Spatial Attention Supervision for Multimodal RLVR*. *Preprint*, arXiv:2605.30912.
- SangHun Im, GiBaeg Kim, Heung-Seon Oh, Seongung Jo, and Dong Hwan Kim. 2023. *Hierarchical Text Classification as Sub-Hierarchy Sequence Generation*. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(11):12933–12941.
- 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.
- Kamran Kowsari, Donald E Brown, Mojtaba Heidarysafa, Kiana Jafari Meimandi, Matthew S Gerber, and Laura E Barnes. 2017. Hdltext: Hierarchical deep learning for text classification. In *2017 16th IEEE international conference on machine learning and applications (ICMLA)*, pages 364–371. IEEE.
- Hohin Kwan, Hongyu Li, Ray Zhang, Manyuan Zhang, Xianghao Kong, Anyi Rao, Jiahao Xie, and Si Liu. 2026. *Video-MME-Logical: A Controlled Diagnostic Benchmark for Video Temporal-Logical Reasoning*. *Preprint*, arXiv:2606.27828.
- Stefan Larson, Anish Mahendran, Joseph J. Peper, Christopher Clarke, Andrew Lee, Parker Hill, Jonathan K. Kummerfeld, Kevin Leach, Michael A. Laurenzano, Lingjia Tang, and Jason Mars. 2019. An evaluation dataset for intent classification and out-of-scope prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1311–1316.
- 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(Apr):361–397.
- Pengfei Li, Beiwen Tian, Yongliang Shi, Xiaoxue Chen, Hao Zhao, Guyue Zhou, and Ya-Qin Zhang. 2022. TOIST: Task Oriented Instance Segmentation Transformer with Noun-Pronoun Distillation. *Advances in Neural Information Processing Systems*, 35:17597–17611.
- Songze Li, Yarong Lan, Zhongpu Bo, Zhaoyang Wang, Zhiqiang Liu, Yuan Yuan, Chengtao Gan, Menghao Qian, Enpei Niu, Xiaoke Guo, Yuanxiang Liu, Zhaoyan Gong, Xiangjin Hu, Liangyurui Liu, Jingdian Lu, Lei Liang, Jun Zhou, Huajun Chen, and Wen Zhang. 2026. Scaling LLM Knowledge Boundaries via Distribution-Optimized Synthesis. *arXiv preprint arXiv:2606.23271*.
- Songze Li, Zhiqiang Liu, Zhaoyan Gong, Xiaoke Guo, Zhongpu Bo, Zhengke Gui, Lei Liang, Huajun Chen, and Wen Zhang. 2025a. Last Layer Logits to Logic: Empowering LLMs with Logic-Consistent Structured Knowledge Reasoning. *arXiv preprint arXiv:2511.07910*.
- Songze Li, Zhiqiang Liu, Zhengke Gui, Huajun Chen, and Wen Zhang. 2025b. Enrich-on-Graph: Query-Graph Alignment for Complex Reasoning with LLM Enriching. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 7672–7692.

- Zhiming Lin, Canran Xiao, and Kai Zhao. 2026. Beyond more context: Retrieval diversity boosts multi-turn intent understanding. In *Proceedings of the ACM Web Conference 2026*, pages 2320–2329.
- Hai Liu, Yu Song, Tingting Liu, Hao Zheng, Lin Chen, Zhaoli Zhang, and You-Fu Li. 2026a. [HomLLM: Exploiting Semantic Homology Relationship for Fine-Grained Bird Image Classification via Large Language Models](#). *IEEE Transactions on Neural Networks and Learning Systems*, 37(3):1221–1235.
- Jun Liu, Zhenglun Kong, Peiyan Dong, Changdi Yang, Tianqin Li, Yanyue Xie, Yifan Gong, Xuan Shen, Pu Zhao, Hao Tang, Geng Yuan, Wei Niu, Wenbin Zhang, Xue Lin, Dong Huang, and Yanzhi Wang. 2026b. Structured agent distillation for large language model agents. In *Proc. of the 25th International Conference on Autonomous Agents and Multi-agent Systems*, pages 3676–3685.
- Jun Liu, Zhenglun Kong, Changdi Yang, Fan Yang, Tianqi Li, Peiyan Dong, Joannah Nanjey, Hao Tang, Geng Yuan, Wei Niu, Wenbin Zhang, Pu Zhao, Xue Lin, Dong Huang, and Yanzhi Wang. 2025. Rcrouter: Efficient role-aware context routing for multi-agent llm systems with structured memory. *arXiv preprint arXiv:2508.04903*.
- Yuanzhe Liu, Jingyuan Zhu, Yuchen Mo, Gen Li, Xu Cao, Jin Jin, Yifan Shen, Zhengyuan Li, Tianjiao Yu, Wenzhen Yuan, Fangqiang Ding, and Ismini Lourentzou. 2026c. PALM: Progress-Aware Policy Learning via Affordance Reasoning for Long-Horizon Robotic Manipulation. *arXiv preprint arXiv:2601.07060*.
- Shuo Lu, Yanyin Chen, Wei Feng, Jiahao Fan, Fengheng Li, Zheng Zhang, Jingjing Lv, Junjie Shen, Ching Law, and Jian Liang. 2025a. Uni-layout: Integrating human feedback in unified layout generation and evaluation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 7709–7718.
- Shuo Lu, Yingsheng Wang, Lijun Sheng, Lingxiao He, Aihua Zheng, and Jian Liang. 2025b. Out-of-distribution detection: A task-oriented survey of recent advances. *ACM Computing Surveys*, 58(2):1–39.
- OpenAI. 2024. GPT-4o System Card. *arXiv preprint arXiv:2410.21276*.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.
- Ningkang Peng, Qianfeng Yu, Xiaoqian Peng, Linjing Qian, Yafei Liu, Canran Xiao, Xinyu Lu, Tingyu Lu, Zhichao Zheng, and Yanhui Gu. 2026. How to achieve prototypical birth and death for ood detection? *arXiv preprint arXiv:2603.15650*.
- Gaurav Sahu, Olga Vechtomova, Dzmitry Bahdanau, and Issam Laradji. 2023. Promptmix: A class boundary augmentation method for large language model distillation. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 5316–5327.
- Wanru Shao. 2026a. Interpretable Ensemble Learning for Network Traffic Anomaly Detection: A SHAP-Based Explainable AI Framework for Embedded Systems Security. In *2026 14th International Symposium on Digital Forensics and Security (ISDFS)*, pages 1–4. IEEE.
- Wanru Shao. 2026b. [Policy-Driven Vulnerability Risk Quantification Framework for Large-Scale Cloud Infrastructure Data Security](#). *Preprint*, arXiv:2604.06252.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.
- Yifan Shen, Yuanzhe Liu, Jingyuan Zhu, Xu Cao, Xiaofeng Zhang, Yixiao He, Wenming Ye, James Rehg, and Ismini Lourentzou. 2025. Fine-Grained Preference Optimization Improves Spatial Reasoning in VLMs. *Advances in Neural Information Processing Systems*, 38:17929–17960.
- Yiyun Su, Huiying Zhu, Yu Tian, Changruo Zhao, Zujun Peng, Yuting Liu, Luyan Zhang, Liang Fan, and Baihua Li. 2026. Agentic-SQL Taxonomy: The Research of Autonomous and Interactive Text-to-SQL with LLMs. In *International Conference on Intelligent Computing*, pages 323–334. Springer.
- Beiwen Tian, Mingdao Liu, Huan-ang Gao, Pengfei Li, Hao Zhao, and Guyue Zhou. 2023. [Unsupervised Road Anomaly Detection with Language Anchors](#). In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7778–7785. IEEE.
- Haoyu Wang, Guangyuan Dong, He Liang, Zijing Zhang, Jiachen Luo, Chuang Liu, Chao Xue, and Hao Tang. 2026. [MemGuard: Persisting Verifier Signals for LLM-Agent Memory Governance](#). *arXiv preprint arXiv:2608.21867*.
- Zihan Wang, Peiyi Wang, Lianzhe Huang, Xin Sun, and Houfeng Wang. 2022a. Incorporating hierarchy into text encoder: a contrastive learning approach for hierarchical text classification. In *Proceedings of the 60th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 7109–7119.
- Zihan Wang, Peiyi Wang, Tianyu Liu, Binghuai Lin, Yunbo Cao, Zhifang Sui, and Houfeng Wang. 2022b. Hpt: Hierarchy-aware prompt tuning for hierarchical

- text classification. In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pages 3740–3751.
- Jonatas Wehrmann, Ricardo Cerri, and Rodrigo Barros. 2018. Hierarchical multi-label classification networks. In *International conference on machine learning*, pages 5075–5084. PMLR.
- 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.
- Canran Xiao, Jiabao Dou, Zhiming Lin, Zong Ke, and Liwei Hou. 2026a. From points to coalitions: Hierarchical contrastive shapley values for prioritizing data samples. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 15995–16003.
- Canran Xiao and Liwei Hou. 2026. Prototype-aligned federated soft-prompts for continual web personalization. In *Proceedings of the ACM Web Conference 2026*, pages 6743–6754.
- Canran Xiao, Chuangxin Zhao, Zong Ke, Fei Shen, and Li Shen. 2026b. Curiosity-driven cooperation for long-tailed multi-label learning. *Neural Networks*, page 109075.
- ShiLin Xiao, Tianxiang Xu, Canran Xiao, Weihao Luo, Liwei Hou, and Chuangxin Zhao. 2026c. Meta-ucf: Unified task-conditioned lora generation for continual learning in large language models. In *The Fourteenth International Conference on Learning Representations*.
- Chao Xue and Ziyuan Gao. 2026. [StructCoh: Structured Contrastive Learning for Context-Aware Text Semantic Matching](#). In *PRICAI 2025: Trends in Artificial Intelligence*, pages 300–315. Springer Nature Singapore.
- Chao Xue, Yao Wang, Mengqiao Liu, Di Liang, Xingsheng Han, Peiyang Liu, Xianjie Wu, Chenyao Lu, Lei Jiang, Yu Lu, Haibo Shi, Shuang Liang, Minlong Peng, and Flora D. Salim. 2026a. Reason only when needed: Efficient generative reward modeling via model-internal uncertainty. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 23302–23319.
- Chao Xue, Yao Wang, Mengqiao Liu, Di Liang, Xingsheng Han, Peiyang Liu, Xianjie Wu, Chenyao Lu, Lei Jiang, Yu Lu, Haibo Shi, Shuang Liang, Minlong Peng, and Flora D. Salim. 2026b. Why supervised fine-tuning fails to learn: A systematic study of incomplete learning in large language models. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 30186–30213.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chuji 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.
- Zhen Yang, Sizai Hou, Kaiwen Zheng, Yaofang Liu, Liang He, Yixuan Chen, and Kangning Cui. 2026. [Target-Aware Calibration Data Selection for Preserving Uncertainty in Quantized Language Models](#). *Preprint*, arXiv:2608.21019.
- Bin Yu, Shijie Lian, Xiaopeng Lin, Zhaolong Shen, Yuliang Wei, Changti Wu, Hang Yuan, Haishan Liu, Bailing Wang, Cong Huang, and Kai Chen. 2026a. [FrameSkip: Learning from Fewer but More Informative Frames in VLA Training](#). *Preprint*, arXiv:2605.13757.
- Bin Yu, Shijie Lian, Xiaopeng Lin, Yuliang Wei, Zhaolong Shen, Changti Wu, Yuzhuo Miao, Xinming Wang, Bailing Wang, Cong Huang, and Kai Chen. 2026b. [TwinBrainVLA: Unleashing the Potential of Generalist VLMs for Embodied Tasks via Asymmetric Mixture-of-Transformers](#). *Preprint*, arXiv:2601.14133.
- Bin Yu, Xinming Wang, Shijie Lian, Haotian Li, Changti Wu, Ruina Hu, Bailing Wang, Yuliang Wei, and Kai Chen. 2025a. [TrajSelector: Harnessing Latent Representations for Efficient and Effective Best-of-N in Large Reasoning Model](#). *Preprint*, arXiv:2510.16449.
- Tianjiao Yu, Xinzhuo Li, Yifan Shen, Yuanzhe Liu, and Ismini Lourentzou. 2025b. Core3D: Collaborative Reasoning as a Foundation for 3D Intelligence. *arXiv preprint arXiv:2512.12768*.
- Tianjiao Yu, Xinzhuo Li, Yifan Shen, Onkar Susladkar, Yuanzhe Liu, Xiaona Zhou, and Ismini Lourentzou. 2026c. ELSA3D: Elastic Semantic Anchoring for Unified 3D Understanding and Generation. *arXiv preprint arXiv:2607.06565*.
- Hossam Zawbaa, Wael Rashwan, Sourav Dutta, and Haytham Assem. 2024. Improved out-of-scope intent classification with dual encoding and threshold-based re-classification. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8708–8718.
- Fan Zhang, Shiming Fan, and Hua Wang. 2026a. TimeSAF: Towards LLM-Guided Semantic Asynchronous Fusion for Time Series Forecasting. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 26267–26284.
- Fan Zhang, Zhiwei Gu, and Hua Wang. 2026b. Decoding with Structured Awareness: Integrating Directional, Frequency-Spatial, and Structural Attention for Medical Image Segmentation. In *Proceedings of*

- the AAAI Conference on Artificial Intelligence, volume 40, pages 12421–12429.
- Feng Zhang, Wei Chen, Meng Gao, Fei Ding, Tengjiao Wang, Jiahui Yao, and Jiabin Zheng. 2025a. Clear up confusion: Iterative differential generation for fine-grained intent detection with contrastive feedback. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 2207–2221.
- Jianguo Zhang, Kazuma Hashimoto, Yao Wan, Zhiwei Liu, Ye Liu, Caiming Xiong, and Philip S Yu. 2022. Are pre-trained transformers robust in intent classification? a missing ingredient in evaluation of out-of-scope intent detection. In *Proceedings of the 4th Workshop on NLP for Conversational AI*, pages 12–20.
- Ruyi Zhang, Hongzuo Xu, Songlei Jian, Yusong Tan, Haifang Zhou, and Rulin Xu. 2026c. Angel or Devil: Discriminating Hard Samples and Anomaly Contaminations for Unsupervised Time Series Anomaly Detection. *Neural Networks*, page 108532.
- Yunyi Zhang, Ruozhen Yang, Xueqiang Xu, Rui Li, Jinfeng Xiao, Jiaming Shen, and Jiawei Han. 2025b. Teleclass: Taxonomy enrichment and llm-enhanced hierarchical text classification with minimal supervision. In *Proceedings of the ACM on Web Conference 2025*, pages 2032–2042.
- Changruo Zhao, Zujun Peng, Yu Tian, Yuting Liu, Yiyun Su, Huiying Zhu, Luyan Zhang, and Heming Zeng. 2026. Agentic-SQL Revisited: Autonomy-Based Taxonomy and Empirical Benchmark Analysis for LLM Text-to-SQL. In *International Conference on Intelligent Computing*, pages 601–612. Springer.
- Xiaoji Zheng, Lixiu Wu, Zhijie Yan, Yuanrong Tang, Hao Zhao, Chen Zhong, Bokui Chen, and Jiangtao Gong. 2024. Large Language Models Powered Context-Aware Motion Prediction in Autonomous Driving. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 980–985. IEEE.
- Jie Zhou, Chunping Ma, Dingkun Long, Guangwei Xu, Ning Ding, Haoyu Zhang, Pengjun Xie, and Gongshen Liu. 2020. Hierarchy-aware global model for hierarchical text classification. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 1106–1117.
- Yixiao Zhou, Dongzhou Cheng, Zhiliang Wu, Yi Yang, Yu Cheng, and Hehe Fan. 2026a. One Refiner to Unlock Them All: Inference-Time Reasoning Elicitation via Reinforcement Query Refinement. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 38957–38978.
- Yixiao Zhou, Yang Li, Dongzhou Cheng, Hehe Fan, and Yu Cheng. 2026b. Look Inward to Explore Outward: Learning Temperature Policy from LLM Internal States via Hierarchical RL. *arXiv preprint arXiv:2602.13035*.
- Yixiao Zhou, Ziyu Zhao, Dongzhou Cheng, Zhiliang Wu, Jie Gui, Yi Yang, Fei Wu, Yu Cheng, and Hehe Fan. 2025. Dropping Experts, Recombining Neurons: Retraining-Free Pruning for Sparse Mixture-of-Experts LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 15169–15186.
- He Zhu, Junran Wu, Ruomei Liu, Yue Hou, Ze Yuan, Shangzhe Li, Yicheng Pan, and Ke Xu. 2024. Hill: Hierarchy-aware information lossless contrastive learning for hierarchical text classification. 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 4731–4745.
- He Zhu, Chong Zhang, Junjie Huang, Junran Wu, and Ke Xu. 2023. Hitin: Hierarchy-aware tree isomorphism network for hierarchical text classification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7809–7821.

A Additional Method Details

A.1 Trajectory Format and Output Schema

For each input text x , TaxoAgent generates a three-step trajectory:

$$\tau = (a^p, a^c, a^v).$$

The parent decision a^p contains a ranked parent-label set $\tilde{Y}_{\text{step}}^p$. The child decision a^c contains a candidate child-label set conditioned on $\tilde{Y}_{\text{step}}^p$. The verification decision a^v emits the final raw prediction

$$\tilde{Y} = (\tilde{Y}^p, \tilde{Y}^c).$$

The final answer is serialized as a JSON object with two fields:

parent_labels and child_labels.

Every emitted label string is matched against the taxonomy vocabulary. Unmatched strings are discarded before reward computation. The hierarchy reward in Eq. (6) is computed after string matching and before structural projection.

A.2 Label Prototype Construction

This section gives the exact construction of the unnormalized prototype $\mathbf{q}_\ell$ used in Eq. (3). Let t_ℓ denote the textual name of label ℓ . For each child label $c \in \mathcal{C}$, define its empirical instance set as

$$\mathcal{D}_c = \{(x_i, Y_i) \in \mathcal{D}_{\text{train}} : c \in Y_i^c\}.$$

The child prototype is

$$\mathbf{q}_c = \lambda_{\text{name}} f_\phi(t_c) + \lambda_{\text{data}} \frac{1}{|\mathcal{D}_c| + \epsilon} \sum_{(x_i, Y_i) \in \mathcal{D}_c} f_\phi(x_i), \quad (13)$$

where λ_{name} and λ_{data} control the contribution of label-name semantics and empirical instance semantics.

For each parent label $p \in \mathcal{P}$, define its child set as

$$\text{Ch}(p) = \{c \in \mathcal{C} : (p, c) \in \mathcal{E}\}.$$

The parent prototype is

$$\mathbf{q}_p = \lambda_{\text{parent}} f_\phi(t_p) + (1 - \lambda_{\text{parent}}) \frac{1}{|\text{Ch}(p)|} \sum_{c \in \text{Ch}(p)} \mathbf{e}_c. \quad (14)$$

After computing $\mathbf{q}_\ell$, each label prototype is normalized as $\mathbf{e}_\ell = \mathbf{q}_\ell / \|\mathbf{q}_\ell\|_2$.

A.3 Structural Projection and Hierarchy Inconsistency Rate

During inference, TaxoAgent applies a deterministic structural projection to the raw prediction. Given $\tilde{Y} = (\tilde{Y}^p, \tilde{Y}^c)$, the projected prediction is

$$\begin{aligned} \Pi_{\mathcal{G}}(\tilde{Y}) &= (\hat{Y}^p, \hat{Y}^c), \\ \hat{Y}^c &= \{\tilde{c} \in \tilde{Y}^c : \tilde{c} \in \mathcal{C}\}, \\ \hat{Y}^p &= (\tilde{Y}^p \cap \mathcal{P}) \cup \{\rho(\hat{c}) : \hat{c} \in \hat{Y}^c\}. \end{aligned} \quad (15)$$

This projection removes invalid child strings and closes the parent set under the parent map ρ .

For dataset-level analysis, we report hierarchy inconsistency rate before projection:

$$\text{HIR} = \frac{1}{N} \sum_{i=1}^N \frac{\sum_{\tilde{c} \in \tilde{Y}_i^c} \mathbb{I}[\rho(\tilde{c}) \notin \tilde{Y}_i^p]}{|\tilde{Y}_i^c| + \epsilon}. \quad (16)$$

A lower HIR indicates better structural compliance of the raw policy output.

A.4 OOS Detection Rule

For out-of-scope evaluation, TaxoAgent uses semantic confidence after trajectory decoding. For each valid parent-child edge $(p, c) \in \mathcal{E}$, define

$$q(x, p, c) = \eta s_x(p) + (1 - \eta) s_x(c), \quad (17)$$

where $s_x(\cdot)$ is the text-label similarity in Eq. (4), and $\eta \in [0, 1]$ controls the parent-child confidence balance. The in-scope confidence is

$$q_{\max}(x) = \max_{(p, c) \in \mathcal{E}} q(x, p, c). \quad (18)$$

The prediction is rejected as OOS when

$$q_{\max}(x) < \delta_{\text{oos}}, \quad (19)$$

where δ_{oos} is selected on the validation split and fixed during test evaluation.

A.5 Taxonomy Expansion

When a new child label c_{new} is inserted under parent p_{new} , the taxonomy is updated as

$$\mathcal{C} \leftarrow \mathcal{C} \cup \{c_{\text{new}}\}, \quad \mathcal{E} \leftarrow \mathcal{E} \cup \{(p_{\text{new}}, c_{\text{new}})\}.$$

Given K seed examples $\{x_k^{c_{\text{new}}}\}_{k=1}^K$ for the new child label, its prototype is initialized as

$$\begin{aligned} \mathbf{q}_{c_{\text{new}}} &= \lambda_{\text{name}} f_\phi(t_{c_{\text{new}}}) \\ &+ \lambda_{\text{data}} \frac{1}{K + \epsilon} \sum_{k=1}^K f_\phi(x_k^{c_{\text{new}}}), \quad (20) \\ \mathbf{e}_{c_{\text{new}}} &= \frac{\mathbf{q}_{c_{\text{new}}}}{\|\mathbf{q}_{c_{\text{new}}}\|_2}. \end{aligned}$$

Algorithm 1 TaxoAgent Training with Utility-Aligned GRPO

Require: Training data $\mathcal{D}_{\text{train}}$, taxonomy $\mathcal{G}$, initial policy π_θ , reference policy π_{ref} , group size G

Ensure: Optimized policy π_θ

- 1: Construct label prototypes $\{e_\ell\}_{\ell \in \mathcal{P} \cup \mathcal{C}}$ using Appendix A.2
 - 2: **for** each optimization step **do**
 - 3: Sample a mini-batch $\{(x_i, Y_i)\}_{i=1}^B$ from $\mathcal{D}_{\text{train}}$
 - 4: **for** each instance (x_i, Y_i) **do**
 - 5: Sample trajectories $\{\tau_i^g\}_{g=1}^G$ from $\pi_{\theta_{\text{old}}}(\cdot | x_i, \mathcal{G})$
 - 6: **for** each trajectory τ_i^g **do**
 - 7: Parse raw prediction $\tilde{Y}_i^g = (\tilde{Y}_i^{p,g}, \tilde{Y}_i^{c,g})$
 - 8: Compute R_{out} by Eq. (5)
 - 9: Compute R_{hier} by Eq. (6)
 - 10: Compute R_{sem} by Eqs. (7)–(8)
 - 11: Compute R_{traj} by Eq. (9)
 - 12: Compute $r_i^g = U(\tau_i^g; x_i, Y_i, \mathcal{G})$ by Eq. (10)
 - 13: **end for**
 - 14: Compute group-relative advantages $\{A_i^g\}_{g=1}^G$ by Eq. (12)
 - 15: **end for**
 - 16: Update θ by maximizing $\mathcal{J}_{\text{GRPO}}(\theta)$ in Eq. (12)
 - 17: **end for**
 - 18: **return** π_θ
-

The parent prototype $e_{p_{\text{new}}}$ is then recomputed using Eq. (14). The policy prompt is updated with the expanded taxonomy vocabulary, while the learned policy parameters remain fixed in the taxonomy-expansion evaluation.

A.6 Training Algorithm

A.7 Implementation Details for Reward Computation

All reward components are computed on parsed label sets after taxonomy-vocabulary matching. Empty predicted sets are assigned zero set-level F1. For semantic distance $d(\ell, B)$, an empty target set gives $d(\ell, B) = 1$. The coefficients in the composite utility are selected on the validation split and fixed for test evaluation. The same validation-selected coefficients are used for all test splits of the same dataset.

A.8 Multi-Level Taxonomy Generalization

The main text introduces the hierarchical decision process using a two-level taxonomy for clarity. TAXOAGENT naturally extends to taxonomies with arbitrary depth. Let $G = (\mathcal{V}, \mathcal{E})$ be a rooted taxonomy graph and let $\text{Anc}(\ell)$ denote the set of all ancestors of label ℓ .

Multi-level structural reward. For a raw prediction $\tilde{Y} \subseteq \mathcal{V}$, the multi-level hierarchy reward is defined as

$$R_{\text{hier}}^{\text{multi}}(\tilde{Y}; G) = 1 - \frac{\sum_{\ell \in \tilde{Y}} \sum_{a \in \text{Anc}(\ell)} \mathbb{I}[a \notin \tilde{Y}]}{\sum_{\ell \in \tilde{Y}} |\text{Anc}(\ell)| + \epsilon}. \quad (21)$$

This reduces to Eq. (7) in the two-level case.

Multi-level outcome reward. Let $\mathcal{V}^{(d)}$ denote labels at depth d . We compute level-wise set F1 and path-wise F1:

$$R_{\text{out}}^{\text{multi}} = \sum_{d=1}^D \omega_d F1(\tilde{Y}^{(d)}, Y^{(d)}) + \omega_{\text{path}} F1(\text{Path}(\tilde{Y}), \text{Path}(Y)), \quad (22)$$

where $\text{Path}(Y)$ denotes the set of root-to-label paths induced by Y . The weights are non-negative and sum to one.

Multi-level trajectory coherence. For a trajectory that emits predictions level by level, the trajectory coherence reward is

$$R_{\text{traj}}^{\text{multi}} = 1 - \frac{\sum_{d=2}^D \sum_{\ell \in \tilde{Y}^{(d)}} \mathbb{I}[\rho(\ell) \notin \tilde{Y}_{\text{step}}^{(d-1)}]}{\sum_{d=2}^D |\tilde{Y}^{(d)}| + \epsilon}. \quad (23)$$

This rewards consistency between adjacent decision stages and generalizes the parent-to-child trajectory reward used for two-level app taxonomies.

RCV1 instantiation. For RCV1, we use the original four-level taxonomy and apply the multi-level reward above. We do not collapse RCV1 into a flat label space. All reported RCV1 Micro-F1 and Macro-F1 scores are computed over the original label set, while hierarchy inconsistency is evaluated before structural projection.

B Additional Experimental Setup

B.1 Dataset Details

We evaluate TAXOAGENT on five datasets that cover complementary hierarchical classification scenarios.

Table 4: Dataset statistics. IOS-APP-ZH and Google-APP-EN are app-domain datasets; RCV1 and WOS are standard HTC benchmarks; CLINC150 is used for intent classification and OOS detection.

Dataset	Level	#Labels	Train	Valid	Test
IOS-APP-ZH	2	144	8,827	2,207	11,034
Google-APP-EN	2	48	48,688	12,173	60,861
RCV1	4	103	20,833	2,316	781,265
WOS	2	141	30,070	7,518	9,397
CLINC150	2	150	13,500	4,500	4,500

Industrial app categorization. IOS-APP-ZH and Google-APP-EN are two self-collected real-world app classification datasets. IOS-APP-ZH is a Chinese app dataset collected from the Apple App Store¹ and annotated with a two-level taxonomy containing 144 labels. Google-APP-EN is an English app dataset collected from Google Play² and annotated with a two-level taxonomy containing 48 labels. Both datasets are constructed from app names and descriptions and manually annotated according to our taxonomy. These two datasets are used to evaluate whether TAXOAGENT can handle industrial hierarchical semantic classification, where labels are fine-grained, semantically overlapping, and frequently long-tailed.

Standard hierarchical text classification. RCV1 (Lewis et al., 2004) is a large-scale news classification benchmark with a four-level taxonomy and 103 hierarchical labels. Web of Science (WOS) (Kowsari et al., 2017) is an academic abstract classification dataset with a two-level taxonomy and 141 labels. These benchmarks evaluate whether TAXOAGENT generalizes beyond app-domain taxonomies to standard HTC settings.

Hierarchical intent classification and OOS detection. CLINC150 (Larson et al., 2019) contains 150 fine-grained intents organized under 10 coarse domains, together with OOS utterances. We use it to evaluate both hierarchical intent classification and the ability to reject inputs that fall outside the current taxonomy. For OOS evaluation, held-out intent classes are treated as in-domain OOS examples, making the rejection task more challenging than detecting unrelated open-domain inputs.

¹<https://developer.apple.com/app-store/product-page/>

²<https://support.google.com/googleplay/android-developer/answer/9859673>

B.2 Metric Details

For hierarchical classification, we report Micro-F1 and Macro-F1. Micro-F1 aggregates true positives, false positives, and false negatives across all labels and therefore reflects overall predictive utility, especially on frequent labels. Macro-F1 computes the unweighted average of per-label F1 scores and is more sensitive to rare and long-tail categories.

For OOS detection, we report OOS-F1 on the held-out OOS subsets. Given true positives (TP), false positives (FP), and false negatives (FN) for OOS prediction, OOS-F1 is computed as

$$\text{OOS-F1} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}. \quad (24)$$

We additionally measure hierarchy inconsistency rate (HIR), which evaluates whether the raw prediction violates parent-child constraints before any structural projection. Given a test set $\mathcal{D}$, raw predictions $\tilde{Y}_i = (\tilde{Y}_i^p, \tilde{Y}_i^c)$, and parent map $\rho(\cdot)$, HIR is defined as

$$\text{HIR} = 100 \times \frac{\sum_{i \in \mathcal{D}} \sum_{\tilde{c} \in \tilde{Y}_i^c} \mathbb{I}[\rho(\tilde{c}) \notin \tilde{Y}_i^p]}{\sum_{i \in \mathcal{D}} |\tilde{Y}_i^c| + \epsilon}. \quad (25)$$

Lower HIR indicates better hierarchy compliance.

B.3 Baseline Details

We compare TAXOAGENT with five groups of baselines.

Encoder-based and hierarchical neural models. We include standard and hierarchical neural classifiers, including BERT (Devlin et al., 2019), HiAGM (Zhou et al., 2020), HMCN (Wehrmann et al., 2018), HTCInfoMax (Deng et al., 2021), HiMatch (Chen et al., 2021), and HMCL (Guo et al., 2025). These methods represent conventional HTC approaches that inject hierarchy through architectures, graph modeling, local-global prediction, or hierarchy-aware objectives.

Hierarchy-aware contrastive learning methods. We compare with HGCLR (Wang et al., 2022a) and HILL (Zhu et al., 2024), which improve hierarchical classification by shaping the text-label representation space with hierarchy-aware contrastive signals.

LLM-enhanced taxonomy and prompting methods. We include TELEClass (Zhang et al., 2025b), HiEPS (Xia et al., 2025), and TaxMorph (Golde et al., 2026). These baselines use

LLM-generated taxonomy information, prompt ensembling, or taxonomy-aware prompting to improve hierarchical classification.

Direct LLM prompting methods. We evaluate direct prompting with ChatGPT-4o (OpenAI, 2024) and DeepSeek-V3.2 (DeepSeek-AI, 2025). These baselines test whether strong general-purpose LLMs can solve the task without utility-aligned policy optimization.

OOS and few-shot intent detection methods. For OOS detection, we compare with representative intent detection methods, including Prompt-Mix (Sahu et al., 2023), CUC (Zhang et al., 2025a), and FCSLM (Chen and Wang, 2025). These methods are designed for robust intent classification, boundary learning, or few-shot OOS detection, but do not explicitly couple rejection with parent-child hierarchy validity.

Unless otherwise stated, TAXOAGENT uses Qwen3-14B as the base policy model (Yang et al., 2025). All hyperparameters are selected on the validation split and fixed for test evaluation.

B.4 Details of Out-of-Scope Detection

We use CLINC150 for out-of-scope intent detection. CLINC150 contains fine-grained user intents organized under coarse-grained domains, together with utterances that do not belong to any in-scope intent. To evaluate in-domain OOS robustness, we follow a domain-level held-out intent setting on the *Banking* and *Credit Cards* domains.

Each selected domain contains 15 intent classes. For each domain, we use 10 intent classes as in-scope classes and hold out the remaining 5 intent classes as in-domain OOS classes. The held-out classes are semantically close to the in-scope classes because they come from the same coarse domain. This setting is therefore more challenging than detecting unrelated open-domain OOS utterances, since the model must reject intents that are topically similar but not covered by the current label space.

Table 5: CLINC150 OOS split statistics. IS denotes in-scope intent classes. ID-OOS denotes in-domain out-of-scope intent classes.

Domain	IS Classes	ID-OOS Classes	Test IS	Test OOS
Banking	10	5	500	350
Credit Cards	10	5	500	350

For evaluation, ACC measures the classification accuracy over in-scope test utterances. OOS-F1

measures whether the model can correctly separate in-scope utterances from held-out in-domain OOS utterances. Given true positives (TP), false positives (FP), and false negatives (FN) for OOS prediction, OOS-F1 is computed as

$$\text{OOS-F1} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}}. \quad (26)$$

We report OOS-F1 separately for the *Banking* and *Credit Cards* domains to evaluate robustness across different intent groups.

B.5 Semantic error distance protocol.

To evaluate not only the number of errors but also their semantic severity, we compute semantic error distance over false-positive labels. For each test example x_i with ground-truth label set Y_i and predicted label set $\hat{Y}_i$, each false-positive label $\tilde{\ell} \in \hat{Y}_i \setminus Y_i$ receives

$$d_{\text{err}}(\tilde{\ell}, Y_i) = 1 - \max_{\ell \in Y_i} s_{\tilde{\ell}, \ell}, \quad (27)$$

where $s_{\tilde{\ell}, \ell}$ is the label-label semantic similarity defined in Eq. 4. Lower values indicate that the wrong prediction remains semantically close to at least one correct label. For each method, we collect all false-positive labels on the test set and estimate the distribution of d_{err} with the same density-estimation setting. We also report the mean semantic error distance

$$\mu_{\text{err}} = \frac{1}{|\mathcal{E}|} \sum_{(\tilde{\ell}, Y_i) \in \mathcal{E}} d_{\text{err}}(\tilde{\ell}, Y_i), \quad (28)$$

where $\mathcal{E}$ denotes the set of all false-positive error instances. The near-miss rate is computed as

$$\begin{aligned} \text{NearMiss} &= 100 \times \frac{1}{|\mathcal{E}|} \\ &\times \sum_{(\tilde{\ell}, Y_i) \in \mathcal{E}} \mathbb{I}[d_{\text{err}}(\tilde{\ell}, Y_i) < 0.25]. \end{aligned} \quad (29)$$

A higher near-miss rate means that the model more often confuses the gold label with a semantically nearby alternative rather than an unrelated cross-branch label.

B.6 Utility-margin and risk-coverage protocol.

To analyze OOS rejection, we compare the utility of in-scope classification against the utility of the OOS rejection decision. For each input x ,

let $S_{\text{IS}}(x)$ denote the maximum structured utility among in-scope label trajectories and $S_{\text{OOS}}(x)$ denote the utility of the rejection trajectory. The structured utility margin is defined as

$$M(x) = S_{\text{IS}}(x) - S_{\text{OOS}}(x). \quad (30)$$

A positive margin indicates that the model prefers accepting the input as in-scope, while a negative margin indicates that the model prefers OOS rejection. We use the absolute margin as a structured confidence score:

$$C(x) = |M(x)|. \quad (31)$$

For risk-coverage analysis, examples are ranked by $C(x)$ in descending order. At coverage level q , we keep the top q fraction of examples, denoted by $\mathcal{S}_q$, and compute

$$\text{Risk}(q) = 1 - \frac{1}{|\mathcal{S}_q|} \sum_{x_i \in \mathcal{S}_q} \mathbb{I}[\hat{z}_i = z_i], \quad (32)$$

where z_i is the ground-truth accept/reject decision and $\hat{z}_i$ is determined by the sign of $M(x_i)$. For in-scope examples, a prediction is counted as correct only if the model accepts the input and predicts the correct in-scope intent. For OOS examples, a prediction is counted as correct when the model rejects the input. Lower risk at the same coverage indicates better uncertainty ranking.

B.7 Long-tail gain protocol.

To determine whether Macro-F1 gains come from rare labels, we compute per-label F1 gains over the strongest prior baseline. For each child label ℓ , let n_ℓ^{train} be its training frequency and let $F_1^m(\ell)$ be the per-label F1 score of method m . The gain of TAXOAGENT is defined as

$$\Delta F_1(\ell) = F_1^{\text{TAXOAGENT}}(\ell) - \max_{m \in \mathcal{B}} F_1^m(\ell), \quad (33)$$

where $\mathcal{B}$ denotes all non-TAXOAGENT baselines. A positive value means that TAXOAGENT outperforms the strongest prior method on label ℓ . We divide child labels into three frequency groups: *tail* labels with $n_\ell^{\text{train}} < 50$, *medium* labels with $50 \leq n_\ell^{\text{train}} < 200$, and *head* labels with $n_\ell^{\text{train}} \geq 200$. In the volcano plots, each point represents one child label; the horizontal axis is $\log_{10}(n_\ell^{\text{train}} + 1)$, the vertical axis is $\Delta F_1(\ell)$, point size indicates test support, and color indicates gain magnitude. Group-level gains are computed by averaging $\Delta F_1(\ell)$ within each frequency group.

C Additional Experimental Results

C.1 SFT versus GRPO

Table 6: Comparison between supervised fine-tuning and GRPO-based decision optimization.

Model	IOS-APP-ZH		Google-APP-EN	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1
Qwen3-14B Raw	31.38	25.02	68.00	66.00
Qwen3-14B + SFT	73.51	31.92	73.00	68.00
Qwen3-14B + GRPO	74.95	34.02	77.00	73.00

Table 6 compares supervised fine-tuning with GRPO-based policy optimization. Although SFT improves over the raw model, GRPO yields consistently higher Micro-F1 and Macro-F1. This supports our claim that hierarchical semantic classification involves non-decomposable utilities that are difficult to optimize with token-level or label-level supervised losses alone. By comparing multiple trajectories for the same input and reinforcing the relatively higher-utility ones, GRPO better aligns the policy with structured decision quality.

C.2 Robustness under Different Test Ratios

To examine robustness under different annotation regimes, we vary the held-out test ratio from 0.1 to 0.5. A larger test ratio corresponds to fewer training examples and therefore a more challenging low-supervision setting. This setting is particularly relevant to industrial taxonomy systems, where annotation cost is high and only a limited portion of instances may be labeled.

Tables 7 and 8 show that TAXOAGENT remains stable as the test ratio increases. In contrast, several encoder-based and deep hierarchical models degrade more noticeably when the available training portion becomes smaller. This result suggests that LLM-based semantic priors and utility-aligned decision optimization are particularly useful in low-annotation regimes. For industrial taxonomy systems, where labeled data are expensive and new categories emerge frequently, such robustness is essential.

C.2.1 Backbone Scale Analysis

Table 9 shows that larger backbones generally improve performance, especially on IOS-APP-ZH. The 8B model is an exception and performs worse than the 4B model on both datasets, suggesting that scaling benefits are not strictly monotonic under the same optimization and decoding configuration. Nevertheless, the strongest results are achieved by

Table 7: Robustness under different test ratios on IOS-APP-ZH. All values are percentages. Best results are shown in bold, and second-best results are underlined.

Model	0.1		0.2		0.3		0.4		0.5	
	Micro	Macro	Micro	Macro	Micro	Macro	Micro	Macro	Micro	Macro
<i>Encoder-based and hierarchical neural models</i>										
BERT	46.7	<u>31.3</u>	47.4	29.2	45.8	26.8	45.6	23.9	43.4	23.9
HTCInfoMax	71.1	24.6	70.7	26.8	70.5	27.0	<u>71.6</u>	28.0	69.8	26.2
<i>Hierarchy-aware contrastive learning methods</i>										
HGCLR	47.4	29.2	44.7	29.4	46.2	27.6	45.5	27.0	42.3	25.9
HGCLR (title+abstract)	46.4	29.6	46.6	<u>29.7</u>	45.7	<u>27.7</u>	45.0	<u>29.8</u>	46.0	28.5
HILL	72.1	24.9	72.0	22.3	<u>71.5</u>	22.6	71.2	22.0	<u>71.0</u>	21.7
HILL (title+abstract)	<u>72.8</u>	24.8	<u>72.3</u>	25.5	71.3	21.2	70.9	21.3	70.8	20.7
<i>Direct LLM prompting methods</i>										
ChatGPT-4o	–	–	–	–	–	–	–	–	37.4	27.1
DeepSeek-3.2	–	–	–	–	–	–	–	–	39.6	<u>32.7</u>
TAXOAGENT	78.1	34.8	78.0	35.2	78.1	35.2	77.2	35.0	77.4	35.2
Δ over best prior	$\uparrow 5.3$	$\uparrow 3.5$	$\uparrow 5.7$	$\uparrow 5.5$	$\uparrow 6.6$	$\uparrow 7.5$	$\uparrow 5.6$	$\uparrow 5.2$	$\uparrow 6.4$	$\uparrow 2.5$

the 32B model, indicating that larger LLMs provide stronger semantic grounding and better taxonomy-aware reasoning capacity.

C.3 Sensitivity to Semantic Reward Weight

The semantic reward in Eq. 8 balances parent-level and child-level semantic penalties through α . We vary α to examine how this balance affects performance.

Table 10 shows that $\alpha = 0.25$ performs best overall. When $\alpha = 0$, the reward relies only on child-level semantic penalties and underuses coarse taxonomy information. When $\alpha = 1$, the reward ignores fine-grained child-level semantic mismatch, which hurts child-label discrimination. Intermediate values provide a better trade-off, suggesting that parent-level semantics are useful as coarse constraints but should not dominate fine-grained decision learning.

C.4 Trajectory Correction Flow

The previous ablations show that multi-step trajectory reasoning improves final classification accuracy and hierarchy consistency. We now ask a more diagnostic question: *where does the improvement come from inside the trajectory?* If the proposed agentic formulation is effective, intermediate steps should not merely rephrase the same prediction; instead, they should actively correct wrong parent decisions, remove invalid child labels, and recover missing fine-grained labels.

Protocol. For each test example x_i , we trace label-level decision units throughout the multi-step prediction trajectory. Let $\hat{Y}_i^{(t)}$ denote the intermediate prediction after stage t . We define the traced decision-unit set as

$$\mathcal{U}_i = Y_i \cup \bigcup_{t=0}^T \hat{Y}_i^{(t)}, \quad (34)$$

where Y_i is the ground-truth label set and T is the final stage. Each unit $u \in \mathcal{U}_i$ is assigned a stage-wise status $z_{i,u}^{(t)}$, such as *correct hit*, *missing gold*, *near false positive*, *cross-branch error*, or *hierarchy violation*. We then visualize the transitions of these statuses across five stages: initial decoding, parent verification, child expansion, semantic checking, and final output.

To quantify correction behavior, we define the trajectory correction rate as

$$\text{TCR} = 100 \times \frac{\sum_i \sum_{u \in \mathcal{U}_i} \mathbb{I}[z_{i,u}^{(0)} \in \mathcal{E}_0] \cdot \mathbb{I}[z_{i,u}^{(T)} \in \mathcal{R}_T]}{\sum_i \sum_{u \in \mathcal{U}_i} \mathbb{I}[z_{i,u}^{(0)} \in \mathcal{E}_0]} + \epsilon, \quad (35)$$

where $\mathcal{E}_0$ contains initially problematic states, including missing gold labels, near false positives, cross-branch errors, and hierarchy violations. The resolved set $\mathcal{R}_T$ contains final states that are either

Table 8: Robustness under different test ratios on Google-APP-EN. All values are percentages. Best results are shown in bold, and second-best results are underlined.

Model	0.1		0.2		0.3		0.4		0.5	
	Micro	Macro	Micro	Macro	Micro	Macro	Micro	Macro	Micro	Macro
<i>Encoder-based and hierarchical neural models</i>										
BERT	<u>82.8</u>	<u>80.9</u>	80.8	<u>79.5</u>	81.2	<u>80.0</u>	81.2	<u>80.0</u>	80.4	<u>79.1</u>
HTCInfoMax	82.5	58.1	82.1	57.3	<u>82.0</u>	57.1	71.6	28.0	81.3	56.4
<i>Hierarchy-aware contrastive learning methods</i>										
HGCLR	54.6	59.1	54.9	59.4	56.3	61.8	53.7	59.5	53.6	59.0
HGCLR (title+abstract)	56.1	61.8	56.0	61.3	55.9	60.7	55.2	60.2	56.0	60.7
HILL	82.0	55.2	81.6	54.8	81.3	55.0	81.1	53.6	80.9	53.5
HILL (title+abstract)	82.6	55.4	<u>82.2</u>	56.3	<u>82.0</u>	55.2	<u>81.7</u>	55.0	<u>81.6</u>	54.8
<i>Direct LLM prompting methods</i>										
ChatGPT-4o	–	–	–	–	–	–	–	–	73.4	70.2
DeepSeek-3.2	–	–	–	–	–	–	–	–	75.8	73.1
TAXOAGENT	84.3	82.8	83.8	83.3	83.1	81.5	82.8	82.2	82.5	80.7
Δ over best prior	$\uparrow 1.5$	$\uparrow 1.9$	$\uparrow 1.6$	$\uparrow 3.8$	$\uparrow 1.1$	$\uparrow 1.5$	$\uparrow 1.1$	$\uparrow 2.2$	$\uparrow 0.9$	$\uparrow 1.6$

Table 9: Effect of Qwen3 backbone size.

Backbone	IOS-APP-ZH		Google-APP-EN	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1
Qwen3-0.6B	66.95	29.83	65.00	61.00
Qwen3-1.7B	68.93	32.07	78.00	76.00
Qwen3-4B	77.28	35.11	82.00	83.00
Qwen3-8B	73.99	33.06	81.00	78.00
Qwen3-14B	77.35	35.21	83.00	81.00
Qwen3-32B	78.34	37.02	83.00	83.00

Table 10: Sensitivity to the semantic reward mixing coefficient α .

α	IOS-APP-ZH		Google-APP-EN	
	Micro-F1	Macro-F1	Micro-F1	Macro-F1
0.00	75.38	33.02	74.00	69.00
0.05	77.09	34.23	80.00	78.00
0.25	77.35	35.21	83.00	81.00
0.50	77.01	35.18	82.00	80.00
0.75	76.02	34.07	79.00	77.00
0.95	76.01	33.78	78.00	75.00
1.00	75.28	33.29	76.00	77.00

correctly recovered or removed as false positives. We also report a soft correction rate that additionally treats benign near-miss errors as non-severe outcomes.

Figure 6 shows clear evidence of step-wise correction. On IOS-APP-ZH, 70.2% of initially problematic decision units are strictly resolved by the final stage, either by recovering the correct label

or by removing an invalid false positive. If benign near-miss outcomes are also counted as non-severe corrections, the soft correction rate rises to 86.6%. Similarly, on Google-APP-EN, TAXOAGENT strictly resolves 69.2% of initially problematic units and reaches a soft correction rate of 86.9%. The child expansion stage recovers a substantial portion of missing gold labels, while the semantic checking stage removes many near false positives and hierarchy violations.

This analysis directly validates the role of multi-step trajectory reasoning. The parent verification stage mainly suppresses structurally invalid paths, child expansion recovers fine-grained labels that are missed by the initial decode, and semantic checking converts many wrong predictions into either removed false positives or benign near-miss errors. Thus, the gains of TAXOAGENT are not only caused by stronger final decoding; they arise from an interpretable sequence of correction operations that progressively improves both semantic accuracy and hierarchy validity.

C.5 Utility Margin for OOS Detection

Structured uncertainty for OOS rejection. The OOS gains should reflect better structured uncertainty rather than merely a better rejection threshold. We compare the utility margin between the best in-scope trajectory and the OOS rejection trajectory, and further evaluate selective prediction with risk-coverage curves; details are given in Ap-

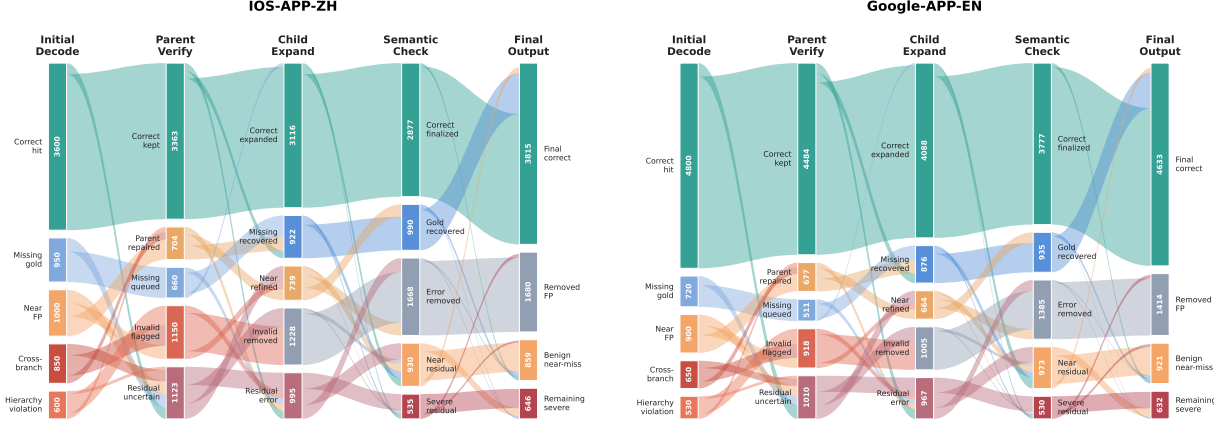


Figure 6: Alluvial visualization of the multi-step correction process in TAXOAGENT. Each ribbon represents a population of label-level decision units flowing across five trajectory stages: initial decoding, parent verification, child expansion, semantic checking, and final output. Green flows correspond to correct or recovered labels, blue flows correspond to missing labels that are recovered, orange flows correspond to near-miss refinements, gray flows correspond to false positives removed by verification, and red flows correspond to residual severe errors. Across both datasets, a large fraction of initially problematic units are converted into either correct labels or removed false positives, showing that the multi-step agent actively corrects predictions rather than merely producing a longer explanation.

pendix B.6. Figure 7 shows that direct prompting produces heavily overlapping in-scope and OOS margins, while FCSLM improves separation but still leaves substantial boundary ambiguity. In contrast, TAXOAGENT shifts in-scope examples toward positive margins and OOS examples toward negative margins, while reducing density near the decision boundary. Its consistently lower risk across coverage levels indicates that the semantic verifier, hierarchy reward, and trajectory coherence reward jointly provide a more faithful uncertainty signal for OOS rejection.

C.6 Same-Backbone Controlled Baselines and Verifier-Only Alternatives

The main results compare TAXOAGENT with both encoder-based hierarchical classifiers and LLM-based prompting baselines. To isolate whether the gains come from the proposed utility-aligned optimization rather than from the backbone model itself, we conduct a same-backbone controlled comparison. All methods in this subsection use the same Qwen3-14B base model, the same taxonomy description, the same label vocabulary, and the same decoding budget unless otherwise specified.

We compare seven variants. **Raw** directly prompts Qwen3-14B to output parent and child labels. **SFT** fine-tunes the model using the serialized gold hierarchical labels. **SFT+Projection** applies deterministic structural projection at inference. **SFT+Trajectory Prompt** uses the

same parent–child–verification output format as TAXOAGENT but without utility-based optimization. **SFT+Verifier Rerank** samples four trajectories and selects the one with the highest verifier score, but does not update the policy with the verifier signal. **DPO** constructs pairwise preferences from high- and low-utility trajectories and optimizes the policy with a pairwise preference objective. **TAXOAGENT** is the full GRPO-based method.

The controlled comparison confirms that the proposed gains cannot be explained by the Qwen3-14B backbone alone. SFT substantially improves over raw prompting, but it still optimizes serialized labels rather than structured hierarchical utility. Structural projection slightly improves final validity but does not teach the raw policy to avoid invalid trajectories, as reflected by the unchanged Raw-HIR. Trajectory prompting and verifier reranking provide additional gains, indicating that the proposed decision format and verifier are both useful. However, reranking remains an inference-time selection strategy and does not internalize the utility signal into the policy. DPO further improves performance by learning from pairwise utility preferences, but TAXOAGENT performs best because GRPO compares multiple trajectories for the same input and directly reinforces relative utility under the full composite reward. These results support the necessity of utility-aligned policy optimization rather than using the verifier only as a post-hoc

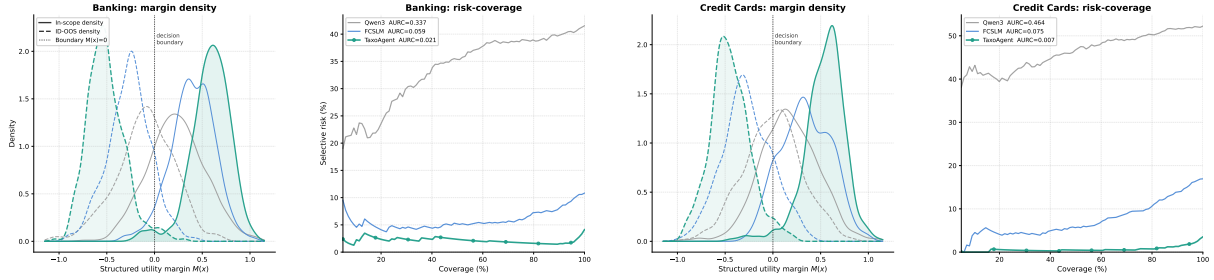


Figure 7: Utility-margin and risk-coverage analysis on CLINC150. Positive margins favor in-scope classification, while negative margins favor OOS rejection. TAXOAGENT achieves clearer IS/OOS separation and lower selective risk, indicating better structured uncertainty. Protocol details are provided in Appendix B.6.

Table 11: **Same-backbone controlled comparison on app-domain taxonomies.** All variants use Qwen3-14B as the base model. Raw-HIR is measured before structural projection, following the main paper. The results show that verifier-only reranking and pairwise preference optimization improve over SFT, but the full GRPO-based TAXOAGENT achieves the best accuracy and hierarchy consistency.

Method	IOS-APP-ZH			Google-APP-EN		
	Micro-F1 $\uparrow$	Macro-F1 $\uparrow$	Raw-HIR $\downarrow$	Micro-F1 $\uparrow$	Macro-F1 $\uparrow$	Raw-HIR $\downarrow$
Qwen3-14B Raw	31.38	25.02	10.2	68.00	66.00	12.1
Qwen3-14B + SFT	73.51	31.92	8.3	73.00	68.00	10.4
Qwen3-14B + SFT + Projection	73.89	32.18	8.3	73.55	68.42	10.4
Qwen3-14B + SFT + Trajectory Prompt	74.18	32.65	7.2	75.40	70.25	8.8
Qwen3-14B + SFT + Verifier Rerank	75.62	33.40	6.4	78.10	74.20	7.5
Qwen3-14B + DPO	76.41	34.36	5.9	80.10	78.40	6.8
TAXOAGENT	77.35	35.21	5.3	82.50	80.70	6.1

Table 12: Inference efficiency on IOS-APP-ZH. Latency is measured in seconds per instance. Tokens denote the average number of generated tokens per instance.

Model	Latency	Tokens
Qwen3-14B Raw	0.24	80
Qwen3-14B + RL	1.24	390
Qwen3-14B + PAP	1.14	370
Qwen3-14B + CAP	1.26	410
TAXOAGENT	1.27	630

filter.

C.7 Efficiency Analysis

We analyze training cost and inference overhead on the IOS-APP-ZH dataset. Training is conducted on 8 A100-80GB GPUs. The total training time ranges from approximately 5 hours on IOS-APP-ZH to more than 20 hours on Google-APP-EN. At inference time, we deploy the model on a single A100-80GB GPU.

As shown in Table 12, TAXOAGENT introduces additional inference overhead compared with direct raw decoding. This overhead comes mainly from multi-step trajectory generation and semantic

verification. However, the increased token budget is expected: the model explicitly reasons over parent candidates, child candidates, hierarchy validity, and semantic alignment. In industrial app classification, new app submissions are relatively bounded in volume compared with high-throughput online retrieval or ranking services. Therefore, the improved classification quality and hierarchy reliability can justify the additional offline inference cost.

C.8 Label-Coverage and Empty-Prediction Sanity Check

A potential concern is that hierarchy and semantic rewards may over-favor overly conservative predictions, because an empty or very small predicted set can avoid hierarchy violations and semantic mismatches. To rule out this degenerate behavior, we report precision-recall decomposition, average predicted child-label cardinality, and the empty-prediction rate.

We compare three reward variants. **No coverage** uses only the original predicted-to-gold semantic penalty. **Hierarchy/Semantic-heavy** increases the weights of hierarchy and semantic terms without the bidirectional coverage penalty, represent-

Table 13: **Label-coverage and empty-prediction sanity check.** Avg. #Child denotes the average number of predicted child labels per instance; GT #Child denotes the average number of gold child labels. Empty rate is the percentage of instances for which no child label is predicted. The full reward maintains balanced precision and recall, matches the gold label cardinality, and does not collapse to empty predictions.

Dataset	Variant	Micro-F1 $\uparrow$	Macro-F1 $\uparrow$	Child-P $\uparrow$	Child-R $\uparrow$	Avg. #Child	GT #Child	Empty Rate $\downarrow$	Raw-HIR $\downarrow$
IOS-APP-ZH	No coverage	77.05	34.80	76.2	75.1	1.09	1.17	0.42	5.1
	Hierarchy/Semantic-heavy	76.42	33.95	79.1	70.3	0.96	1.17	1.31	4.9
	Full	77.35	35.21	76.8	76.6	1.16	1.17	0.08	5.3
Google-APP-EN	No coverage	82.20	80.31	81.3	79.8	1.06	1.09	0.29	5.9
	Hierarchy/Semantic-heavy	81.21	79.06	83.1	75.9	0.95	1.09	0.92	5.5
	Full	82.50	80.70	81.6	81.2	1.09	1.09	0.05	6.1

ing a stress-test setting where conservative prediction could be over-rewarded. **Full** is the default TAXOAGENT reward with bidirectional semantic coverage and outcome reward.

The hierarchy/semantic-heavy stress-test variant obtains slightly lower Raw-HIR, but it does so by reducing recall and predicting fewer child labels. This confirms that hierarchy and semantic rewards should not be used without an explicit coverage signal. The full reward avoids this degeneracy: its average predicted child-label cardinality closely matches the gold cardinality, the empty-prediction rate remains below 0.1%, and child-level precision and recall are balanced. Therefore, the final gains of TAXOAGENT are not caused by under-prediction or empty-label collapse.