Taxon: Hierarchical Tax Code Prediction with Semantically Aligned LLM Expert Guidance

Jihang Li[1,†], Qing Liu[2,†], Zulong Chen[2,*], Jing Wang[2], Wei Wang[2], Chuanfei Xu[3,*], Zeyi Wen[1,*]
[1]*The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, China*
[2]*Alibaba Group, Hangzhou, China*
[3]*Guangdong Laboratory of Artificial Intelligence and Digital Economy (Shenzhen), Shenzhen, China*
zulong.czl@alibaba-inc.com, xuchuanfei@gml.ac.cn, wenzeyi@hkust-gz.edu.cn

***Abstract*—Tax code prediction is a crucial yet underexplored task in automating invoicing and compliance management for large-scale e-commerce platforms. Each product must be accurately mapped to a node within a multi-level taxonomic hierarchy defined by national standards, where errors lead to financial inconsistencies and regulatory risks. This paper presents Taxon, a semantically aligned and expert-guided framework for hierarchical tax code prediction. Taxon integrates (i) a feature-gating mixture-of-experts architecture that adaptively routes multi-modal features across taxonomy levels, and (ii) a semantic consistency model distilled from large language models acting as domain experts to verify alignment between product titles and official tax definitions. To address noisy supervision in real business records, we design a multi-source training pipeline that combines curated tax databases, invoice validation logs, and merchant registration data to provide both structural and semantic supervision. Extensive experiments on the proprietary TaxCode dataset and public benchmarks demonstrate that Taxon achieves state-of-the-art performance, outperforming strong baselines. Further, an additional full hierarchical paths reconstruction procedure significantly improves structural consistency, yielding the highest overall F_1 scores. Taxon has been deployed in production within Alibaba's tax service system, handling an average of over 500,000 tax code queries per day and reaching peak volumes above five million requests during business event with improved accuracy, interpretability, and robustness.**

I. Introduction

Modern e-commerce platforms in China process millions of transactions daily, each requiring the correct assignment of a tax code for invoicing and regulatory compliance. This task, known as *hierarchical tax code prediction*, maps a product–represented by metadata such as title, category, and attributes–to a specific code within a tree taxonomy defined by the State Taxation Administration. With up to ten hierarchical levels and over four thousand leaf categories, manual classification is both costly and error-prone, while errors directly cause financial discrepancies or compliance violations.

A. Background

Tax code prediction is embedded within a complex commercial and regulatory ecosystem that connects merchants, consumers, and tax authorities. As shown in Fig. 1, this ecosystem spans two major operational scenarios: business-to-customer (B2C) and business-to-business (B2B), together covering the end-to-end workflow on large e-commerce platforms. In both scenarios, each product or transaction follows a standardized pipeline of product publication, tax code query, validation, and invoice issuance. These business flows motivate our design by revealing where automation can most effectively reduce manual workload and improve consistency. The following subsections detail the B2C and B2B scenarios.

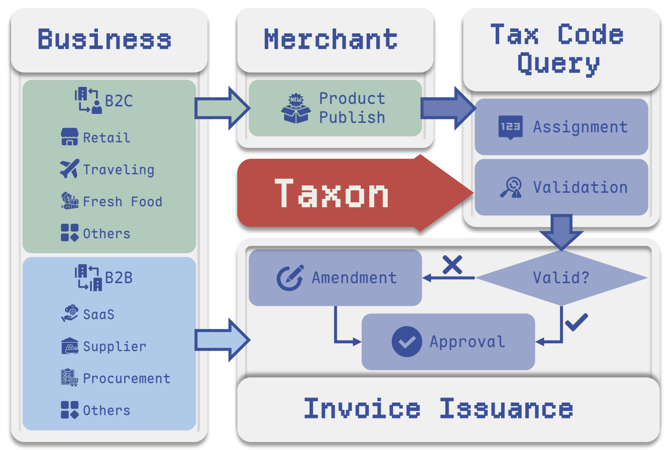


Fig. 1: An illustration of tax code prediction in e-commerce.

a) B2C Scenario: The B2C operations encompass a variety of online services that support customers' daily needs, ranging from retail (e.g., Tmall) and traveling (e.g., Fliggy) to fresh food (e.g., Freshippo), digital content (e.g., Youku), and local services (e.g., Ele.me). In these settings, tax code assignment occurs during the *product publish* phase, when third-party merchants register new items on the platform. Each item must be associated with a valid tax code before it can go online. To assist merchants, traditionally, the platform performs a three-stage progressive query against the internal tax code database organized as key-value pairs: (i) query by *product title* (e.g., "Fully Automatic Washing Machine 3 L"); (ii) if no match is found, fall back to *product category* (e.g., "Home Appliances"); and (iii) if still unmatched, a tax code must be selected manually from a list provided in the merchant console. Once a code is identified at any stage, it is returned to the merchant for confirmation and recorded in the listing.

† Equal contribution.
* Corresponding authors.

b) B2B Scenario: In contrast, the B2B branch of the ecosystem handles internal transactions. The platform itself serves as a buyer or service provider for scenarios such as supplier management, asset procurement, and enterprise SaaS. Here, tax code assignment is performed internally rather than by external merchants. When a business transaction occurs, the system proceeds to the *invoice validation* process, mirroring the same three-step query pipeline used in B2C. If the retrieved code is valid and consistent with the record, the transaction proceeds automatically to *invoice issuance* and approval. Otherwise, any absent or inconsistent code triggers an *amendment* step handled by in-house tax experts before final approval. This closed-loop workflow, as shown in Fig. 1, ensures that both B2C and B2B channels maintain traceable and auditable tax code assignments with minimal human intervention and compliant with official regulations.

B. Motivation

However, maintaining and validating tax codes across large business operations remains challenging and error-prone. The tax code database is curated manually and periodically updated whenever new products or categories emerge. This process is labor-intensive and often inconsistent, especially when product metadata is incomplete or ambiguous. Such inconsistencies propagate to erroneous code assignments, leading to financial discrepancies and compliance risks. As a result, reliance on human experts limits efficiency, accuracy, and scalability in high-volume e-commerce operations.

These issues motivate the need for an automated hierarchical tax code prediction system that can assist merchants and internal auditors in assigning codes accurately and consistently from product metadata. However, automatic prediction introduces additional challenges stemming from the hierarchical and heterogeneous nature of real-world data:

- *High-dimensional, incomplete features.* The model must integrate textual and structured modalities–titles, categories, prices, and organizational attributes–despite missing or unreliable fields that violate the clean-text assumption of traditional classifiers.
- *Deep and large label hierarchy.* The official taxonomy spans up to ten levels and over four thousand leaf nodes. Flat classifiers ignoring parent-child constraints accumulate cascading errors. Effective models must exploit hierarchical dependencies to prune irrelevant subtrees and maintain valid paths.
- *Semantic ambiguity and label relationships.* Many tax codes have overlapping meanings or "other" categories that differ only by fine-grained material or functional nuances. Capturing sibling and parent-child relations is essential to avoid inconsistent predictions.
- *Noisy and inconsistent supervision.* Historical records contain human and procedural inconsistencies, and long-tail imbalance amplifies noise. The model must remain robust through confidence calibration, hierarchy-aware losses, and semantic validation.

C. Proposal

Tax code prediction faces the same hierarchical and semantic complexities that make manual maintenance difficult. It can be formulated as a problem of *hierarchical text classification* (HTC), where each product is mapped to a node in a multi-level taxonomy defined by the *Goods and Services Tax Classification Catalogue* [1]. Like the Harmonized System (HS) used in customs and trade, it demands learning fine-grained category boundaries that respect strict parent-child dependencies and semantic constraints.

Earlier works on tax or HS code prediction relied on conventional text classifiers [2]–[4]. Although they automated parts of the process, these models remained fragile in practice: errors accumulated along deeper hierarchy levels, semantic inconsistencies with official definitions persisted under noisy supervision, and their interpretability for business auditing was limited. They also struggled to adapt to heterogeneous data sources, hindering scalability in enterprise systems.

To address these limitations, we propose **Taxon** for hierarchical tax code prediction. Taxon integrates structural reasoning, semantic alignment, and data robustness through three complementary components:

- **Hierarchical feature-gating mixture-of-experts (MoE)** architecture that captures multi-level dependencies and adaptively routes features across taxonomy levels to maintain logical path consistency;
- **Semantic consistency-assisted prediction** guided by large language models (LLMs) acting as domain experts, which distill semantic alignment between product titles and official tax definitions through expert judgments;
- **Multi-source training pipeline** integrating over eight million business records to provide diverse and reliable supervision, effectively mitigating noise and handling incomplete product metadata.

In contrast to traditional HTC approaches, Taxon explicitly bridges taxonomic structure, semantic meaning, and real-world business workflows, forming a unified architecture that is both interpretable and robust. Deployed within Alibaba's enterprise tax service system, Taxon now processes more than one million tax code queries per day across multiple business units, achieving high accuracy, scalability, and transparency under real operational conditions.

Our preliminary analyses further reveal that the primary source of classification errors lies not in misidentifying the correct leaf category but in violating the hierarchical structure that connects intermediate nodes. In other words, residual errors mainly arise from *structural inconsistency* along the predicted path rather than *semantic misunderstanding* at the leaf level. This observation highlights a common limitation of existing HTC methods, which often entangle structural and semantic errors within a single objective. Motivated by this, we explicitly decouple the two aspects: (i) maximizing semantic precision at the leaf level, and (ii) designing an independent mechanism, "**RePath**" (reconstructing path from leaf), to ensure structural consistency of the predicted hierarchy.

D. Contribution and Organization

This work makes the following key contributions:

1) We formalize the end-to-end workflow of hierarchical tax code prediction across both B2C and B2B operations, clarifying how automated inference integrates with expert validation to ensure compliance and consistency in real business processes.
2) We present Taxon, a semantically aligned and expert-guided hierarchical prediction framework that unifies hierarchical feature-gating and LLM-assisted semantic supervision within a single, end-to-end system.
3) We design an LLM-distilled semantic labeling pipeline that bridges structured tax definitions and unstructured product text, enabling robust supervision even under noisy or incomplete annotations.
4) We demonstrate large-scale industrial deployment and comprehensive evaluation across internal and public benchmarks, showing consistent gains over strong hierarchical and semantic baselines and validating the system's reliability in daily production environments.

The remainder of this paper is organized as follows: Sec. II presents our framework and training pipeline; Sec. III reports the experimental evaluation; Sec. IV reviews related work; and Sec. V concludes our work.

II. Methodology

Hierarchical tax code prediction maps a product to a node in a multi-level taxonomy, requiring both structural validity and semantic correctness. We address this with a framework that couples a hierarchical feature-gating MoE for multi-level prediction with LLM-guided semantic consistency supervision to align product descriptions with official tax definitions.

A. Framework Overview

We formulate tax code prediction as a hierarchical multi-feature classification problem that leverages product information and structured business metadata. The framework comprises (i) a text encoder that captures the semantic features of product titles and categories, and (ii) a hierarchical feature-gating mixture-of-experts (MoE) classifier that predicts the most appropriate tax code at each level of the taxonomy.

As shown in Fig. 2, the system processes the input features in a bottom-up manner, where product titles, categories, and other relevant attributes are first encoded into latent representations. These features are then routed to specialized experts through a gating mechanism, which adaptively assigns them to the most relevant experts based on feature semantics. The final tax code is determined by aggregating predictions from the experts at various levels of the hierarchy.

During inference, the system prioritizes leaf-level predictions with the highest confidence. If no valid leaf-level prediction is found, the model selects the deepest node along the predicted path with the highest confidence score. This approach ensures that the output respects the hierarchical structure of the tax code while maintaining robustness across varying levels of granularity.

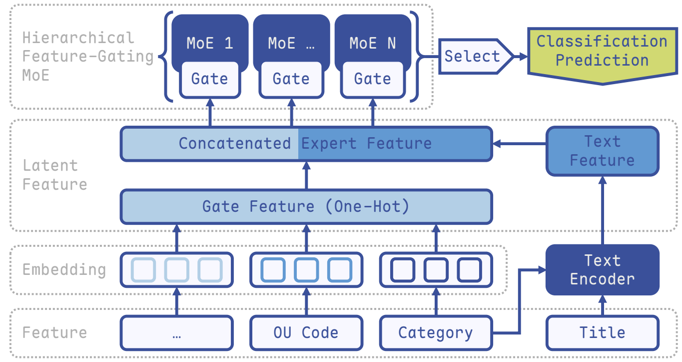


Fig. 2: Overview of the proposed framework. The system processes both textual and structured features through a hierarchical MoE to predict tax codes at multiple levels.

a) Input Features: The framework takes as input both unstructured and structured product information. The primary textual inputs are the product title and category name, which describe the product's key attributes. These are complemented by structured business metadata, including the business unit (BU) code, organization unit (OU) code, and system code (e.g., "TD", "HP6", and "TM_3C"). Although these structured identifiers carry little inherent semantics, they provide useful contextual cues that distinguish product distributions across business units. Additional features such as category property values (CPVs)–for example, "hard disk type" for computers or "material" for furniture–can also be incorporated when available to enrich the input space.

b) Feature Encoding: The textual features (*title* and *category*) are processed through a pre-trained text encoder such as BERT [5], TextCNN [6], or XLNet [7], which captures contextual semantics in a latent representation. Structured features are mapped to learnable embeddings and represented as one-hot vectors to preserve their categorical distinctions. The encoded text representation and structured embeddings are concatenated into a unified feature vector, forming a comprehensive representation of each product that combines semantic meaning with business context.

c) Hierarchical Feature-Gating MoE: The concatenated feature vector serves as the input to a hierarchy of feature-gating MoE modules, where each module corresponds to one level in the tax code taxonomy. Given that the official taxonomy contains ten levels, we deploy ten MoE modules in parallel, each contains eight experts and is responsible for predicting labels within its respective level. Each expert specializes in a subset of product types or hierarchical levels, enabling the overall model to capture level-specific dependencies while maintaining scalability across a large label space. The gating network of each MoE is trained to generate routing weights from the structured one-hot features and produces a softmax distribution over experts, guiding the input representation toward the most relevant ones. All expert and gate parameters are initialized with Xavier initialization, and optimized end-to-end via backpropagation.

d) Final Prediction: Since the hierarchical MoE produces multiple predictions across taxonomy levels, a selection strategy is applied to determine the final output. The framework first prioritizes leaf-level predictions with the highest confidence, as leaf nodes represent the most specific tax codes. If no confident leaf prediction exists, the model selects the deepest valid node along the predicted path with the highest probability score. This strategy ensures that the final output adheres to the hierarchical taxonomy, maintaining logical path consistency while preserving robustness against uncertain or incomplete predictions.

B. Hierarchical-Semantic Training

Our framework adopts an training paradigm that optimizes two complementary objectives: *hierarchical classification* and *semantic consistency*. As illustrated in Fig. 3, it couples a hierarchical feature-gating MoE with an LLM-guided semantic alignment branch in a multi-task pipeline. The hierarchical component captures dependencies across taxonomy levels to preserve structural integrity, while the semantic branch enforces alignment between product descriptions and tax definitions through expert-derived signals. In this process, a large language model acts as a *distiller of domain knowledge*, translating its reasoning into lightweight, task-specific supervision. By judging whether each product title matches its tax code definition, the LLM provides domain-grounded feedback distilled into a compact *Semantic LLM*, enabling efficient and continual learning of hierarchical-semantic consistency.

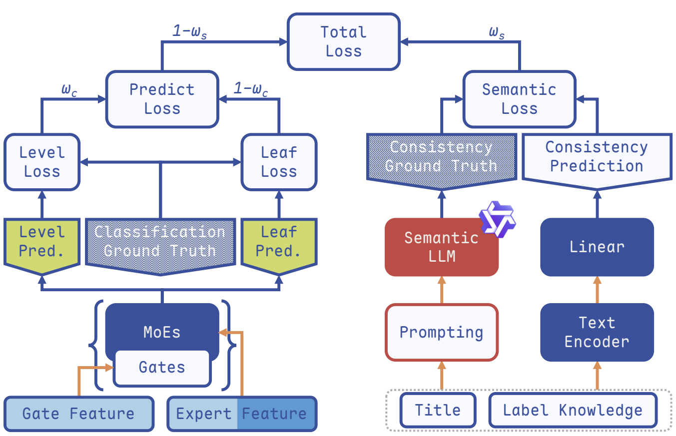


Fig. 3: Overview of the training workflow, integrating hierarchical feature-gating MoE with LLM-assisted semantic consistency. The multi-stage pipeline ensures robust learning across the tax code hierarchy.

On the left side of Fig. 3, the hierarchical feature-gating MoE models dependencies across taxonomy levels. During training, the gating network learns routing weights from one-hot business metadata (e.g., product category and system codes), directing each input to the most relevant experts. Each expert specializes in a subset of levels and produces predictions for its assigned scope. The model minimizes a hierarchical classification loss across all levels to preserve the overall structural consistency of the tax code hierarchy.

On the right side of Fig. 3, the semantic alignment branch enforces consistency between product titles and official tax code definitions. A title and its assigned tax code are fed into a semantic encoder: one branch encodes the definition into a reference embedding, while the other projects the product representation into a predicted embedding. Their divergence defines the semantic consistency loss, guiding towards taxonomically valid and semantically coherent predictions.

The total loss combines the hierarchical classification and semantic consistency terms, ensuring that the model jointly captures structural accuracy and semantic validity. Training proceeds through backpropagation with tunable weights balancing the two objectives. The following subsection details the loss formulation, dataset construction, and training pipeline, including the semantic LLM distillation process that transfers expert-level judgments into lightweight supervision for large-scale learning.

1) Hierarchical Classification Loss: Given an input x with ground-truth N-level labels $\mathbf{y} = \{y_1, y_2, \dots, y_N\}$, the framework produces predictions $\hat{y}_i$ at each level $i \in \{1, 2, \dots, N\}$ of the tax code hierarchy ($N = 10$ in our case). Each level is supervised with a cross-entropy loss [8]:

$$\mathcal{L}_i = \text{cross\_entropy}(\hat{y}_i, y_i). \tag{1}$$

Among the N levels, one corresponds to the leaf node, which provides the most fine-grained supervision. The overall hierarchical classification loss is computed as a weighted sum of all intermediate-level losses and the leaf-level loss:

$$\mathcal{L}_c = \omega_c \sum^{N-1} \mathcal{L}_i + (1 - \omega_c)\mathcal{L}_{\text{leaf}}, \tag{2}$$

where the weight $\omega_c \in [0, 1]$ controls the relative importance of intermediate versus leaf-level supervision. This formulation allows the model to propagate structural information throughout the taxonomy while maintaining precision at the leaf level, preventing error accumulation across deeper hierarchy layers.

2) Auxiliary Semantic Consistency Loss: To align model predictions with the semantic meaning of official tax code definitions, we introduce an auxiliary task that enforces semantic consistency. For each class label y_i, let $d(y_i)$ denote its official description in the tax code catalogue. A semantic encoder converts the pair $y_i, d(y_i)$ into a latent representation, producing a reference semantic label z. In parallel, the encoded product title is projected into a predicted semantic label $\hat{z}$ by a lightweight linear layer. The loss is then defined as:

$$\mathcal{L}_s = \text{cross\_entropy}(\hat{z}, z), \tag{3}$$

This auxiliary supervision encourages predictions that are semantically consistent with expert-verified tax code definitions, thereby reducing misclassifications caused by surface-level textual similarity or ambiguous descriptions. The two objectives are combined into the total training loss:

$$\mathcal{L} = \omega_s \mathcal{L}_c + (1 - \omega_s)\mathcal{L}_s, \tag{4}$$

where the weight $\omega_s \in [0, 1]$ balance the contribution of structural and semantic learning.

3) Dataset Construction: To support large-scale training, we construct four internal datasets that together reflect the full tax code workflow illustrated in Fig. 1. Each dataset corresponds to a distinct stage in the product-to-invoice life cycle, covering both B2C and B2B operations. Tab. I summarizes their sources and cardinalities:

TABLE I: Internal datasets used for model training.

Source	**Dataset**	**Samples**
Tmall Supermarket Stock	Goods Registry	702,869
Business Event Records	Knowledge Base	198,010
Invoice Validation Records	Validation Record	331,195
Domestic Output Invoice Pool	Invoice Archive	7,400,210

- *Goods Registry.* Collected from in-stock records of the Tmall Supermarket. When merchants register products on the e-commerce platform, the system logs aligned pairs of *(product title, tax code)* or *(product category, tax code)*. It captures the merchant-facing process of code assignment.
- *Knowledge Base.* This manually curated repository is maintained by tax experts and periodically updated during business events. Whenever a product with a new title or category appears without an existing match, experts validate and insert the corresponding tax code entry.
- *Validation Record.* Extracted from the invoice validation process, this dataset contains manual correction cases where automatically retrieved codes failed validation. These samples provide valuable supervision signals for identifying hard or ambiguous examples.
- *Invoice Archive.* Collected from internal aggregated and de-identified invoice dataset that includes validated codes.

To ensure data integrity, all datasets undergo strict deduplication based on normalized product titles and tax code identifiers before partitioning into training, validation, and test splits. Together, these corpora provide comprehensive coverage of both B2C and B2B operations, forming a robust foundation for hierarchical and semantic supervision.

4) Training Pipeline: To maximize data quality and training efficiency, we design a four-stage pipeline that integrates dataset cleansing, semantic labeling, and joint optimization. This process transforms the raw internal data into a high-quality corpus suitable for hierarchical and semantic supervision. The overall workflow is illustrated in Fig. 4.

a) Stage 1 - Dataset Cleansing: Starting from an 8M-sample raw internal set, we perform intra-category clustering and rule-based filtering to remove mislabeled, duplicated, or inconsistent entries, yielding a 682,207-sample cleansed set.

b) Stage 2 - Development Set Construction: A preliminary hierarchical MoE model is trained on the cleansed set to assess prediction confidence. Among 670,013 correctly predicted samples, fewer than 3% fall below a confidence threshold of 0.9 (see Fig. 5). To balance easy and hard cases, we randomly sample 5% of high-confidence positives and retain all low-confidence or incorrect samples, forming a 64,030-entry development set that supports semantic labeling and model evaluation as shown in Tab. II.

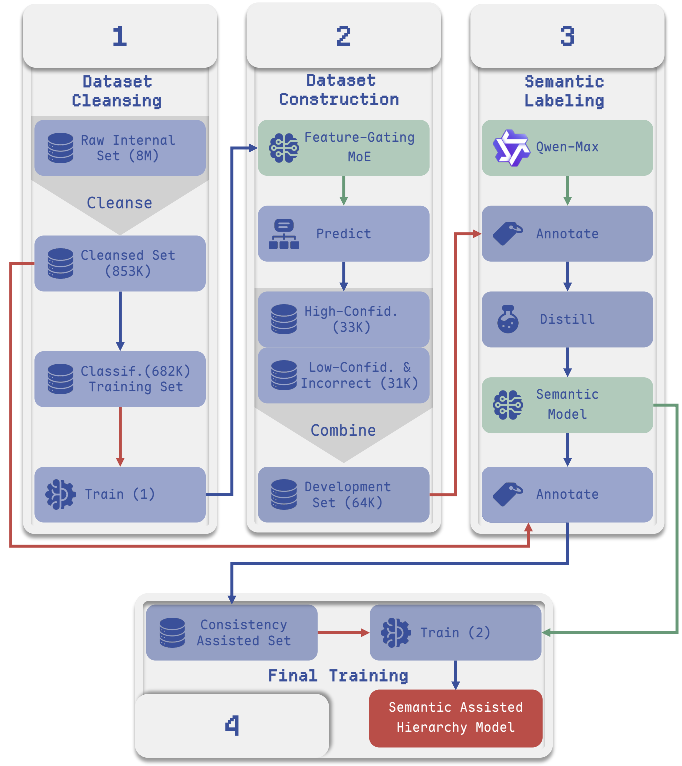


Fig. 4: Four-stage data processing and training pipeline.

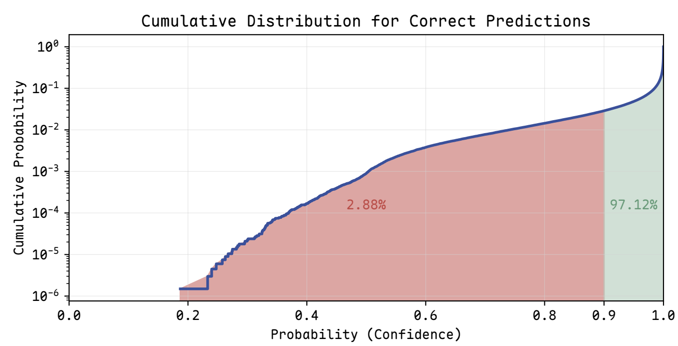


Fig. 5: Log-scale Cumulative distribution of prediction confidence for correctly predicted samples. Only 2.88% of predictions fall below 0.9 confidence, motivating balanced sampling.

TABLE II: Composition of the development dataset stratified by prediction correctness and confidence.

Type	**Samples**	**Sampling**	**Selected**	**Consistency**
$P \geq 0.9$	650,713	5%	32,536	Y: 72.3% N: 21.5% −: 6.2%
$P < 0.9$ Incorrect	19,300 12,194	100% 100%	19,300 12,194	Y: 59.6% N: 38.2% −: 2.2%
Total	682,207	64,030	–	–

c) Stage 3 - Semantic Labeling: Entries in the development set are annotated with semantic consistency judgments using Qwen-Max [9], a LLM acting as a domain expert. We prompt it with structured JSON-based instructions to judge whether each product title aligns with the definition of its assigned tax code. Each annotation outputs one of three labels: `Y` (consistent), `N` (inconsistent), or – (uncertain), together with a brief rationale. These labels provide supervision for distilling a lightweight semantic judgment model that mimics the LLM during large-scale training. Specifically:

1) First, we provide the model with a role hint that constrains it to act as a professional tax expert. The role hint specifies the task of verifying whether the product title aligns with the official tax code definition, as shown in Fig. 6. This helps the model focus on semantic consistency within the tax domain.

```
Role
You are a tax expert specializing in determining the correct
classification of goods based on official tax code taxonomy. Your
task is to judge whether a given product title has been correctly
assigned to a tax code category according to the definition.
```

Fig. 6: Role hinting the model to act as a tax expert and defines the task of semantic consistency judgement.

2) Next, we specify detailed structured requirements in JSON format that define both the input and output schemas. These schemas include field meanings and labeling rules, ensuring consistency in the labeling process, as shown in Fig. 7.
3) Finally, we append three input-output examples as few-shot demonstrations to help guide the model's reasoning process, as shown in Fig. 8. For clarity, only one example is presented here, though the model is provided with multiple instances during training.

Although the prompts are issued in Chinese to match our data, we present English versions for readability. The resulting judgments are summarized in the "Consistency" column of Tab. II and distilled into the semantic judgment model used in large-scale training.

d) Stage 4 - Final Consistency-Assisted Training: The distilled semantic model is applied to the cleansed corpus to generate consistency signals for all entries, producing a consistency-assisted training set. The final hierarchical MoE model is then trained with the combined objectives of hierarchical classification and semantic consistency (as defined in Sec. II-B). This joint optimization enforces both structural fidelity to the taxonomic hierarchy and semantic alignment with official definitions, resulting in a robust and interpretable tax code predictor.

C. Business Integration

Fig. 9 illustrates the real-world deployment of Taxon within Alibaba's ecosystem, forming a closed loop from data collection and semantic distillation to real-time tax code prediction and validation in production. Offline platforms handle data processing and model preparation, while online platforms host the deployed service, enabling millions of daily tax code queries from the invoice system.

```
Requirement
1.The input is a JSON string in the format:
  {
    “goods_name”: “xxx”,
    “tax_code”: “xxx”,
    “tax_code_definition”: {
      “tax_code”: “xxx”,
      “tax_code_name”: “xxx”,
      “tax_code_desc”: “xxx”
    }
  }
  where goods_name is the product title, tax_code is the assigned
  tax code, and tax_code_definition contains the detailed
  definition of the category.

2.You must decide whether the product title (goods_name) is
  consistent with the definition of the assigned tax code
  (tax_code_definition).

3.The output must be a single JSON string in the format:
  {
    “is_consistency”: “Y/N/-”,
    “reason”: “xxx”
  }
  No additional explanations should be provided.

4.In the output, is_consistency indicates whether the product’s tax
  code assignment is consistent with the definition:
  - Y = consistent,
  - N = inconsistent,
  - - = cannot be determined.
  Only these three values are allowed.

5.In the output, the reason field should briefly explain the
  judgment in no more than 30 words.
```

Fig. 7: Structured requirements specifying input/output schemas, including field meanings and labeling rules, guiding the model's annotation process

```
Example
Input:
{
  “goods_name”: “Maternity shorts, summer wear, suit-style wide-leg
  shorts, petite casual five-point leggings, maternity clothing
  summer collection, black, 3XL (recommended 150-1)”,
  “tax_code”: “1040201990000000000”,
  “tax_code_definition”: {
    “tax_code”: “1040201990000000000”,
    “tax_code_name”: “Other clothing”,
    “tax_code_desc”: “”
  }
}
Output:
{
  “is_consistency”: “Y”,
  “reason”: “Maternity shorts are semantically consistent with
  other clothing’s constraints, classification is correct.”
}
```

Fig. 8: A few-shot example demonstrating the desired input-output behavior for semantic consistency judgment. This example helps guide the model's reasoning.

1) Offline Training: We train Taxon on the full 8M-sample corpus using a single NVIDIA V100 GPU, and end-to-end training takes about 3 days and 20 hours. The offline pipeline consists of four stages across internal AI and data platforms:

1) *MaxCompute Platform* deduplicates and normalizes invoice issuance records and tax audit logs to construct the 8M internal dataset described in Sec. II-B.
2) *Bailian LLM Platform* provides Qwen-Max inference APIs; a confidence-based strategy selects hard examples for consistency labeling.
3) *PAI Platform (Platform of Artificial Intelligence)* distills Qwen-Max annotations into a lightweight semantic model for corpus-wide semantic labeling.
4) *Nebula Distribution Training Platform* enables multi-node, multi-GPU distributed training for large-scale optimization.

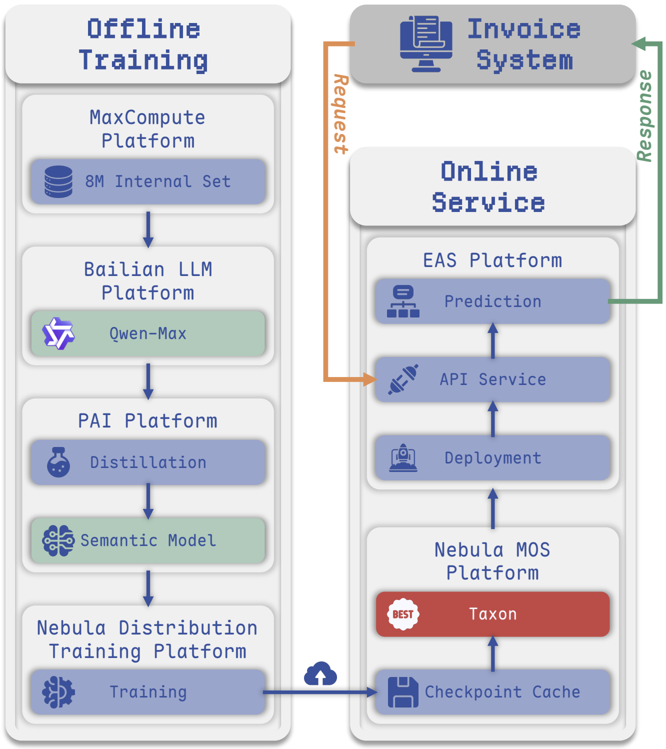


Fig. 9: Integration of the proposed framework into Alibaba's enterprise tax service ecosystem.

2) Online Serving: Trained checkpoints are uploaded to the *Nebula Model Object Storage (MOS) Platform* for unified version and metric management. The best-performing model on the validation set is released as the production model, *Taxon*. It is deployed via the *Elastic Algorithm Service (EAS) Platform*, which encapsulates Taxon as a scalable API for real-time tax code prediction in both product publishing and invoicing workflows. When a request arrives from the *Invoice System*–the group-wide invoicing center–the EAS service performs prediction and validation, returning results to the pipeline. The deployed system supports high concurrency and low latency, sustaining about 30 queries per second with an average end-to-end response time of about 90 ms handling over 500K requests daily and peaking above 5M during major business events.

This integration forms a continuous improvement loop: newly issued invoices and expert-corrected samples are periodically fed back into MaxCompute for incremental retraining, allowing Taxon to evolve with changing tax regulations and product catalogs while maintaining accuracy, interpretability, and compliance. Tax code catalogues evolve over time, but updates are typically infrequent (often on a yearly cadence). When changes occur, we refresh the internal knowledge base and retrain the model using newly issued invoices and expert-corrected samples under the updated rules, ensuring Taxon stays compliant as regulations and product catalogs evolve.

III. Experiments

We conduct experiments to evaluate the effectiveness, generality, and robustness of Taxon. Specifically, we examine: (i) whether expert-guided semantic supervision improves accuracy across taxonomic depths; (ii) the contribution of each architectural component; and (iii) how Taxon compares with state-of-the-art hierarchical and label-semantic-aware baselines under realistic business data conditions. We evaluate on both proprietary and public datasets spanning diverse domains and hierarchy complexities. Unless otherwise stated, all models use identical data splits and optimization settings. We report results at both path- and leaf-level granularities, followed by ablations and error analyses to quantify the impact of semantic alignment and hierarchical modeling.

A. Datasets

We conduct experiments on two datasets with distinct language and domain characteristics to demonstrate the versatility of our approach: *TaxCode* (our proprietary dataset) *and Web of Science (WOS)* [10], as summarized in Tab. III. We exclude other widely used hierarchical text classification datasets such as *Reuters Corpus Volume I (RCV1)* [11] and *The New York Times Annotated Corpus (NYT)* [12], because they are inherently multi-label, where each sample may correspond to multiple intermediate or leaf nodes, whereas our formulation assumes a single root-to-leaf path per instance.

TABLE III: Datasets for main and ablation experiments.

Dataset	Language	Level	Samples	Split
TaxCode (Ours)	Chinese	10	852,758	Train.: 545,765 Val.: 136,441 Test.: 170,551
WOS	English	2	46,985	Train.: 30,070 Val.: 7,518 Test.: 9,397

The *TaxCode* dataset is a proprietary large-scale Chinese corpus derived from e-commerce tax classification records, corresponding to the cleansed set introduced in Fig. 4. It reflects realistic product and service labeling governed by national tax regulations, covering diverse categories such as groceries, electronics, and logistics services. After multi-stage filtering and de-duplication, the dataset comprises 852,758 instances labeled under a ten-level hierarchical taxonomy aligned with the *Goods and Services Tax Classification Catalogue* [1]. Each sample contains a product title, its hierarchical tax path, and structured metadata (e.g., BU and OU codes), providing supervision for structural and semantic modeling.

The *Web of Science (WOS)* dataset [10] is a widely adopted English benchmark for hierarchical text classification. It consists of 46,985 research paper abstracts organized into a two-level taxonomy of scientific disciplines. We include this dataset to assess cross-lingual and cross-domain generalization. Unlike TaxCode, WOS contains well-formed textual inputs and shallow hierarchies, making it a complementary test bed for evaluating model robustness beyond noisy business data.

For both datasets, we partition them into training, validation, and test sets in a 64/16/20 ratio to ensure balance across hierarchy levels, as summarized in the "Split" column of Tab. III. In the main experiments, we randomly sample 5% of the datasets for training, validation, and testing. For the ablation experiments, we use the entire dataset to ensure comprehensive evaluation.

B. Metrics

We evaluate model performance using both *macro* and *micro* F_1 scores at two granularities: *path level* and *leaf level*. Metrics are derived by measuring the overlap between the predicted and true label sets along the path:

$$\begin{aligned} \mathrm{Precision} &= \frac{|\hat{\mathbf{y}} \cap \mathbf{y}|}{|\hat{\mathbf{y}}|}, \quad \mathrm{Recall} = \frac{|\hat{\mathbf{y}} \cap \mathbf{y}|}{|\mathbf{y}|}, \\ F_1 &= \frac{2 \times \mathrm{Precision} \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}. \end{aligned} \tag{5}$$

Macro F_1 measures the average performance across all categories. Specifically, we first compute precision, recall, and F_1 for each label and then average the scores across all 4,482 categories. In contrast, micro F_1 aggregates all predictions and computes a single score over the entire set of samples. The two perspectives jointly reflect both per-category robustness and overall predictive accuracy.

a) Path-Level Evaluation: At the path level, we treat each root-to-node path as a sequence of labels. We compare the predicted path $\hat{\mathbf{y}}$ with the ground-truth path $\mathbf{y}$ and compute F_1 from the overlap of their label sets. Following the practice in prior hierarchical text classification work [13], correctness is evaluated per node along the path rather than requiring all ancestors to match, reflecting how well the model captures taxonomy structure across depths.

b) Leaf-Level Evaluation: At the leaf level, we focus on end-task correctness which evaluates whether the final prediction matches the correct tax code, since taxation decisions operate at the most specific level. For samples with partial annotations due to incomplete metadata, we treat the deepest annotated node as the *effective leaf label*. If a prediction stops early (e.g., due to low confidence), the farthest predicted node is used as the effective leaf prediction. We compute F_1 over these effective leaf labels to quantify end-task correctness.

C. Baselines

We compare Taxon with representative hierarchical baselines including HGCLR [13], HPT [14], HILL [15], HyILR [16], and LH-Mix [17]. HGCLR and HILL are contrastive-learning approaches that incorporate hierarchy-aware objectives; HPT is a prompt-tuning method that aligns HTC with masked language modeling; HyILR models instance-specific local hierarchical relations with hyperbolic geometry; and LH-Mix improves hierarchical prompt tuning via local-hierarchy-guided Mixup. All methods are trained and evaluated on the same data splits with identical evaluation metrics, and we tune hyperparameters on the validation set.

We do not include a direct LLM in-context learning baseline, because the label space contains over 4,000 leaf tax codes and the hierarchy is up to ten levels deep, making naïve "choose-from-all-labels" prompting impractical; retrieval- augmented prompting would introduce a retriever whose recall and engineering choices materially affect end-to-end accuracy.

D. Main Results

We evaluate our proposed framework on both public and internal business domain datasets to assess its generality and practical effectiveness. The experiments are designed to answer three key questions: (i) whether our expert-guided knowledge distillation improves prediction accuracy across domains and taxonomic depths; (ii) how semantic alignment and hierarchy modeling contribute to overall performance; and (iii) how our approach compares with existing hierarchy-aware and label-semantic-aware baselines under realistic data scales. We report the main quantitative results on two representative datasets, followed by detailed ablation and analysis studies.

Tab. IV reports the main results on the *TaxCode* dataset at both path-level and leaf-level granularities. Among the compared methods, our proposed Taxon framework achieves the overall best performance, with the underlined values representing the highest results without the final row. Compared the baselines, Taxon consistently yields higher macro- and micro-level F_1 scores, demonstrating the effectiveness of our semantically consistent and expert-guided framework in handling large hierarchical label spaces.

TABLE IV: F_1 across different methods on *TaxCode* dataset.

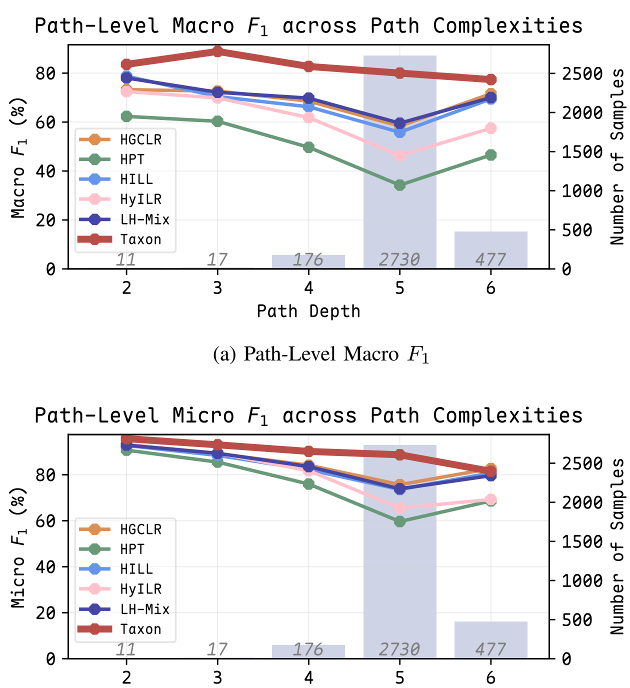
Method	Path (%)		Leaf (%)	
	Macro	**Micro**	**Macro**	**Micro**
HGCLR [13]	80.53	<u>90.18</u>	62.32	77.81
HPT [14]	73.13	85.09	40.71	62.79
HILL [15]	80.78	89.78	61.65	77.50
HyILR [16]	75.88	87.88	54.02	71.22
LH-Mix [17]	79.84	89.24	63.73	76.49
Taxon	<u>85.06</u>	89.22	<u>78.30</u>	<u>87.06</u>
Taxon + RePath	**89.37**	**93.16**	**78.30**	**87.06**

However, the path-level micro F_1 of Taxon (89.22) is slightly lower than HILL (89.78) and LH-Mix (89.24). Examination of misclassified cases shows that this gap mainly stems from path inconsistency rather than leaf errors–for instance, the model may predict the correct leaf but include an extra intermediate node. This indicates that while leaf predictions are accurate, independent layer-wise classifiers can occasionally yield redundant or invalid path nodes.

To address this issue, we conduct an additional experiment that reconstructs the hierarchical path directly from the predicted leaf node, following the corresponding ancestor chain in the taxonomy. We denote this variant as "**Taxon + RePath**". As shown in the last row of Tab. IV, this simple post-processing step substantially improves the path-level macro and micro F_1 scores to 89.37 and 93.16, respectively, while maintaining identical leaf-level results. The consistent gain confirms that most residual path errors originate from internal-node inconsistencies rather than semantic misunderstanding at the leaf level. By leveraging the more reliable leaf predictions to reconstruct a coherent root-to-leaf path, the RePath procedure effectively enhances the overall structural correctness of hierarchical predictions.

To further analyze model behavior across different levels of label hierarchy, we follow the statistics proposed in [16] and evaluate performance with respect to *path complexity*, which reflects the depth of the hierarchical path defined by a specific tax code. This analysis provides additional insight into how each model handles short versus long classification paths. All test samples fall within path depths from 2 to 6, with 11, 17, 176, 2730, and 477 instances respectively for each depth.

At the path level (Fig. 10), both macro and micro F_1 scores of Taxon remain consistently higher than those of all baselines across all depths, demonstrating superior stability and robustness when predicting at varying path complexities. Notably, all methods exhibit similar performance trends as path complexity increases, suggesting that deeper and structurally richer taxonomies remain more challenging for hierarchical prediction models.

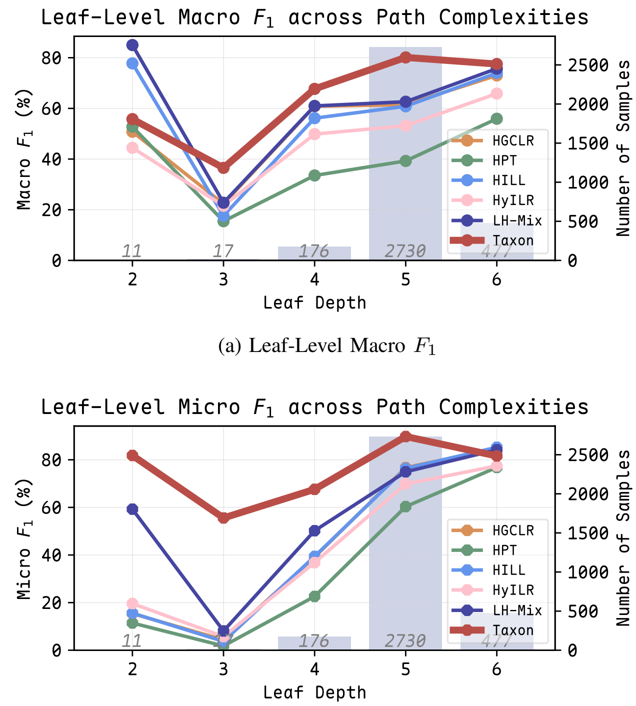


(a) Path-Level Macro F_1

(b) Path-Level Micro F_1

Fig. 10: Model performance at different levels of path complexity. Bars indicate the number of test samples at each depth.

At the leaf level (Fig. 11), Taxon shows slightly lower macro F_1 than LH-Mix and HILL at depth 2, and lower micro F_1 than LH-Mix and HILL at depth 6, but surpasses all other baselines at the remaining depths. Notably, at depth 5 where the majority of samples reside, Taxon achieves the highest macro and micro F_1 among all methods.



(a) Leaf-Level Macro F_1

(b) Leaf-Level Micro F_1

Fig. 11: Model performance at different levels of path complexity. Bars indicate the number of test samples at each depth.

Beyond the proprietary *TaxCode* dataset, we further evaluate our framework on the public *WOS* benchmark to examine Taxon's generalization capability under a shallower two-level hierarchy. The results are reported in Tab. V, where we present both **Taxon** and **Taxon + RePath** for consistency with the previous experiment. Since the WOS taxonomy contains only two levels, the RePath correction provides negligible improvement, as reflected by the identical scores at the first two decimal places. Nonetheless, Taxon achieves strong performance across all metrics, recording 86.74 macro F_1 and 93.67 micro F_1 at the path level, and 86.06 macro F_1 and 87.68 micro F_1 at the leaf level. Our proposed method surpasses or matches state-of-the-art baselines such as HGCLR, HILL, and LH-Mix. These results confirm that Taxon generalizes effectively beyond the tax domain and remains competitive even without domain-specific semantic supervision.

TABLE V: F_1 across different methods on *WOS* dataset.

Method	Path (%)		Leaf (%)	
	Macro	Micro	Macro	Micro
HGCLR [13]	**86.89**	91.23	85.84	**88.08**
HPT [14]	86.40	90.70	85.34	87.63
HILL [15]	86.19	90.68	85.40	87.77
HyILR [16]	86.22	90.74	85.07	87.56
LH-Mix [17]	86.08	90.35	85.14	87.24
Taxon	86.74	**93.67**	**86.06**	87.68
Taxon + RePath	86.74	**93.67**	**86.06**	87.68

E. Ablation Study

To systematically assess the contribution of each component in our proposed framework, we design a progressive series of ablation experiments. Starting from a minimal setup that combines a text encoder with a linear classification head, we incrementally integrate additional modules to evaluate their individual effects:

1) Text encoder backbone;
2) Category name feature fusion;
3) Feature-gating MoE substitution;
4) Hierarchical supervision;
5) Semantic consistency auxiliary task.

This stepwise design allows us to isolate where each enhancement contributes most to performance improvements.

1) Text Encoder Choice: We begin by comparing different text encoders to identify the most effective backbone for representing product descriptions. Experiments are conducted on the *Goods Registry* dataset from the Tmall supermarket stock. As shown in Tab. VI, transformer-based encoders significantly outperform the convolutional baseline. Among them, BERT achieves the highest accuracy (93.28%), slightly ahead of XLNet (93.21%), and clearly surpassing TextCNN (91.26%). This demonstrates that contextualized representations from pre-trained language models capture the nuanced semantics in short and noisy product titles more effectively than shallow convolutional encoders. Based on this result, BERT is selected as the major encoder in all subsequent ablation studies.

TABLE VI: Comparison of text encoders on *Goods Registry*.

Model	Accuracy
TextCNN	91.26%
XLNet	93.21%
BERT	**93.28%**

2) Category Name Fusion: We then inspect whether incorporating structured business features can enhance classification accuracy. Since most business attributes (e.g., business unit or organization unit codes) lack intrinsic semantics, we focus on the *category name* as the only semantically meaningful feature. This feature is integrated into the model in one-hot or textual forms. To ensure that the effectiveness of the category feature is not confined to a single backbone, the one-hot variant is applied to all three encoders (TextCNN, XLNet, and BERT), while the textual variant is evaluated only on BERT–the best-performing text encoder from the previous study.

As shown in Tab. VII, incorporating the category name in one-hot form consistently improves performance across all encoders and datasets, demonstrating that structured categorical context complements other input signals such as BU/OU codes and product titles. By comparing the configurations with one-hot category names, we can again confirm that BERT remains the most effective backbone under our experimental setup. Furthermore, substituting the one-hot encoding with a textual form yields additional gains (e.g., from 90.05% to 90.64% on the multi-source setting). These results suggest that embedding category names facilitates stronger alignment between input features and tax code labels.

TABLE VII: Comparison of category name fusion strategies.

Configuration	Goods Registry	Validation Record	Invoice Archive	Multi-Source (All Three)
TextCNN + One-Hot	96.16%	88.51%	85.80%	86.77%
XLNet + One-Hot	94.32%	89.71%	89.45%	89.89%
BERT + One-Hot	94.48%	89.91%	89.60%	90.05%
BERT + Textual	**96.61%**	**91.54%**	**90.04%**	**90.64%**

3) Classifier Choice: Building on the previous setup with BERT and fused category features, we further replace the linear classification head with our proposed feature-gating MoE layer. This module dynamically routes feature representations to specialized expert submodules based on feature semantics, enabling more adaptive decision boundaries that better capture the complex relationships between different feature types.

As shown in Tab. VIII, the feature-gating MoE classifier consistently outperforms the linear classifier across all datasets, achieving the largest gain on the *Validation Record* set, with a +1.25% absolute improvement. These results highlight the effectiveness of adaptive expert routing, which models inter-feature specialization, improving both classification accuracy and generalization across diverse and heterogeneous business domains.

TABLE VIII: Comparison of classification heads.

Configuration	Goods Registry	Validation Record	Invoice Archive	Multi-Source (All Three)
BERT + Linear	96.61%	91.54%	90.04%	90.64%
BERT + MoE	**96.61%**	**92.79%**	**90.11%**	**90.77%**

4) Hierarchical Supervision: To investigate the impact of hierarchical classification, we introduce a multi-level loss that supervises intermediate-level predictions along the taxonomic path, with a loss weight of $\omega_c = 0.2$ (see Eq. (2) and Fig. 3). As a comparison, we also explore a sequence-generation approach that produces intermediate-level labels.

As shown in Tab. IX, incorporating the multi-level loss consistently improves accuracy across all datasets. This demonstrates that hierarchical supervision enables the model to better capture the underlying structure of the tax code hierarchy. In contrast, replacing ground-truth labels with sequentially generated labels results in a slight reduction in accuracy, highlighting the importance of high-quality hierarchical signals for effective supervision.

TABLE IX: Effectiveness of hierarchical classification.

Configuration	Goods Registry	Validation Record	Invoice Archive	Multi-Source (All Three)
BERT + MoE	96.61%	92.79%	90.11%	90.77%
BERT + MoE + Sequential	96.40%	91.88%	89.43%	90.11%
BERT + MoE + Hierarchical	**96.69%**	**93.04%**	**90.31%**	**90.97%**

5) Semantic Consistency: Finally, we evaluate the effect of the proposed semantic consistency auxiliary task, which explicitly encourages predictions to align with the official definitions of tax codes. Due to the additional computational cost, this ablation is conducted on the smaller development set (as defined in Tab. II and illustrated by block 2 in Fig. 4).

As shown in Tab. X, introducing the semantic loss with a weight of $\omega_s = 0.2$ (see Eq. (3) and Fig. 3) yields consistent yet moderate improvements across datasets, except for the *Validation Record* set, where results remain comparable. We further test a variant that replaces the distilled semantic annotation model (block 3 of Fig. 4) with an untrained Qwen-2.5-32B [9] model for direct semantic labeling, which achieves smaller gains, confirming the importance of domain adaptation in the distilled model.

Although the quantitative improvement appears limited, the qualitative effects are more substantial. We summarize three main reasons: (i) our semantic model corrects noisy labels in the ground truth, so while apparent agreement with the original annotations decreases, true performance improves; (ii) it effectively fixes most knowledge-level misclassifications, where the predicted tax code better matches the product's meaning even if it differs from the noisy label; and (iii) residual errors arise partly from imperfect semantic labels–the LLM's own labeling inaccuracies occasionally misguide training, leaving a small portion of hard cases unresolved. Together, these findings indicate that explicit semantic alignment enhances interpretability and robustness, refining the model's understanding of the relationship between product text and formal tax code definitions.

TABLE X: Effectiveness of semantic consistency.

Configuration	Goods Registry	Validation Record	Invoice Archive	Multi-Source (All Three)
BERT + MoE + Hierarchical	98.30%	**97.50%**	92.13%	94.71%
BERT + MoE + Hierarchical + Qwen2.5-32B	98.51%	96.83%	92.35%	94.89%
BERT + MoE + Hierarchical + Semantic	**98.69%**	96.82%	**92.47%**	**94.92%**

Across all ablation stages, we observe a cumulative improvement trend: (i) pre-trained Transformer backbones, particularly BERT, offer stronger contextual representations than shallow CNNs; (ii) incorporating category name features adds structured semantic cues that complement other business attributes; (iii) the feature-gating MoE improves cross-domain generalization through adaptive expert specialization; (iv) hierarchical supervision enables structurally consistent multi-level predictions; and (v) the semantic consistency task further refines alignment with official definitions. Together, these modules contribute distinct yet complementary gains, culminating in a semantically and hierarchically consistent tax code predictor.

IV. Related Work

Tax code prediction can be viewed as a special case of hierarchical text classification (HTC), where the label space follows a multi-level taxonomic structure. This section reviews related work from two perspectives: (i) general methods for HTC, and (ii) studies specifically targeting tax or similar prediction or classification systems.

A. Hierarchical Text Classification

HTC aims to assign each text to one or more labels organized in a hierarchical taxonomy. Prior work can be broadly categorized along two dimensions: (i) the *task form* (sequence generation vs. classification), and (ii) the *methodological emphasis* (hierarchy-aware, label-semantic-aware, hybrid, and prompt/LLM-based). Generative methods based on CNNs or RNNs suffered from exposure bias and error propagation [18]–[21], and generally underperformed classification approaches in fully supervised settings [22]–[24]. With the rise of prompt tuning and LLMs, generative formulations have shown strong few-shot performance. However, when training data are sufficient, the most competitive results are still achieved by hierarchy-aware and label-semantic-aware methods.

a) Sequence Generation Methods: Sequence generation approaches recast hierarchical classification as a path prediction task. SGM [25] introduced a decoder-based architecture to capture label dependencies for multi-label classification. Seq2Tree [24] modeled HTC as a sequence-to-tree problem with constrained decoding for label consistency, while PAAM-HiA-T5 [26] proposed a hierarchy-aware T5 [27] with path-adaptive attention for level/path dependency modeling. UMP-MG [28] leveraged unidirectional message passing to respect tree directionality, and HiDEC [29] achieved compact hierarchy-aware decoding with fewer parameters. Recent work such as HiGen [30] added dynamic, level-guided losses and task-specific pre-training to mitigate data imbalance. Despite progress, error propagation along decoding paths remains a key challenge, especially under deep hierarchies.

b) Hierarchy-Aware and Label-Semantic-Aware Methods: A large body of research explicitly models label hierarchies and semantics. Hierarchy-aware models such as AttentionXML [31], HTrans [20], and HiAGM [22] capture structural dependencies via hierarchical attention, recurrent transfer learning, or GCN/TreeLSTM encoders. In parallel, label-semantic-aware methods leverage label names and definitions: HTCInfoMax [32] maximizes text–label mutual information, while HiMatch [23] formulates HTC as text–label semantic matching in a shared embedding space. These lines are increasingly unified in hybrid models such as HBGL [33], HGCLR [13], and K-HTC [34], integrating hierarchy-guided contrastive learning with external knowledge graphs. Recent variants (HiTIN [35], HJCL [36], HALB [37], HBM [38], HILL [15], HiAdv [39]) further extend this paradigm with structure-entropy encoders, supervised contrastive learning, asymmetric/adaptive losses, and local hierarchy modeling. Other work explores alternative geometries and negative sampling, including hyperbolic embeddings (HyILR [16]) and hierarchical ranking (HiSR [40]), yielding incremental but consistent gains.

c) Prompt-Tuning and LLM-Based Methods: The emergence of large pre-trained models and parameter-efficient tuning has brought new perspectives to HTC. HPT [14] proposed hierarchy-aware prompt tuning to align multi-label classification with masked language modeling objectives, while HierVerb [41] designed path-constrained verbalizers and hierarchical contrastive learning for few-shot HTC. NERHTC [42] reformulated HTC as a named-entity-recognition task with CRF-based path consistency. Another recent advance, LH-Mix [17], enhances prompt tuning for HTC by integrating text-specific local hierarchies into manually designed depth-level prompts. To further capture implicit correlations among sibling categories, it applies a Mixup strategy guided by local hierarchy correlation, substantially improving robustness across multiple benchmark datasets. LLM-based extensions further broadened this frontier. Retrieval-style ICL [43] applied in-context learning with hierarchical retrieval databases; DPT [44] introduced dual prompt tuning with cross-layer contrastive objectives; and TELEClass [45], TTC [46], and KG-HTC [47] combined LLMs with taxonomy enrichment, multimodal consistency, and knowledge graph–based RAG mechanisms for zero- or few-shot HTC. These models excel under low-resource conditions but remain difficult to scale efficiently to large taxonomies.

To summarize, HTC research spans structural, semantic, hybrid, and LLM-based paradigms, with the current state-of-the-art in hybrid hierarchy- and label-semantic-aware frameworks. However, existing models often encode hierarchy with explicit structure encoders (e.g., GCN, TreeLSTM) and model label semantics via contrastive alignment. We instead propose a *hierarchical feature-gating MoE* framework, where hierarchical relations are implicitly learned through expert routing and semantic relations are strengthened by a judgment task. This architecture offers a new way to balance global structural consistency with fine-grained semantic discrimination.

B. Tax Code and Harmonized System Classification

Tax code and Harmonized System (HS) classification constitute real-world instances of HTC problems, where labels form deep taxonomies used for customs, taxation, and trade management. The HS code system, established by the World Customs Organization (WCO), defines a six-digit international standard extended by local administrations (e.g., 10-digit CN codes in China and the U.S.). Such multi-level taxonomies pose similar challenges of fine-grained disambiguation, label imbalance, and semantic overlap observed in HTC.

Luppes et al. [2] pioneered CNN-based classification for HS codes, demonstrating strong performance (F_1 score up to 0.92 at the HS-2 level) using domain-specific embeddings. Zhao et al. [48] addressed value-added tax (VAT) classification in Chinese invoices via a Heterogeneous Directed Graph Attention Network (HDGAT), improving accuracy by leveraging relational dependencies among invoice items. Subsequent works expanded model complexity and modality. Liao et al. [4] integrated ERNIE, BiLSTM, and multi-scale attention mechanisms to handle semi-structured Chinese product descriptions with domain-specific terminology. Amel et al. [49] introduced a multimodal deep learning framework combining text and image representations from e-commerce declarations, achieving top-5 accuracy exceeding 98%. Shubham et al. [3] developed a hierarchical conditional BERT classifier and a knowledge-graph–based auditing mechanism for HS code assignment, while Anggoro et al. [50] applied contrastive learning with Sentence-BERT and Multiple Negative Ranking loss to improve embedding quality and downstream classification across international customs datasets.

Collectively, these studies highlight the persistent challenges of high-dimensional, hierarchical label spaces and semantic ambiguity in product descriptions. However, existing solutions are largely classification-based and rely on handcrafted hierarchy modeling. Our work extends this line by introducing a unified MoE-based framework that jointly learns hierarchical structure and semantic alignment, offering a scalable, end-to-end approach to national tax code prediction.

V. Conclusion

In this work, we addressed the challenge of hierarchical tax code prediction, a mission-critical yet previously under-explored task in large-scale invoicing systems. We introduced Taxon, a semantically aligned and expert-guided prediction framework that unifies hierarchical modeling and semantic reasoning. By combining a feature-gating MoE classifier for structure-aware routing with an LLM-distilled semantic consistency task, Taxon effectively bridges the gap between textual product descriptions and formal tax definitions. Our multi-source data pipeline further ensures robustness to noisy supervision and incomplete metadata, reflecting realistic e-commerce scenarios.

Comprehensive experiments on both internal and public datasets demonstrate that Taxon substantially outperforms representative hierarchical and semantic baselines across all evaluation levels. Notably, our analysis revealed that residual errors primarily arise from structural inconsistencies in intermediate nodes rather than from semantic misunderstanding at the leaf level. Motivated by this insight, we introduced the *RePath* procedure, which reconstructs the hierarchical path from leaf predictions to enforce logical path validity. This simple yet effective post-processing step consistently enhances path-level F_1, confirming that accurate leaf predictions can reliably recover the full taxonomic structure. Looking forward, we plan to integrate *RePath*-style structural correction into the training objective, e.g., via differentiable path-validity constraints or joint structured decoding, so that the model learns end-to-end hierarchical consistency rather than relying on post-processing.

VI. Acknowledgments

This work is supported by National Key R&D Program of China under Grant No. 2024YFA1012700. It is also funded by the NSFC Project (No. 62306256) and the Natural Science Foundation of Guangdong Province (No. 2025A1515010261).

REFERENCES

[1] S. T. A. of the People's Republic of China, "Goods and Services Tax Classification Catalogue," 2017. [Online]. Available: https://fgk.chinatax.gov.cn/zcfgk/c100012/c5194763/content.html

[2] J. Luppes, A. P. de Vries, and F. Hasibi, "Classifying Short Text for the Harmonized System with Convolutional Neural Networks," *Radboud University*, 2019.

[3] Shubham, A. Arya, S. Roy, and S. Jonnala, "An Ensemble-Based Approach for Assigning Text to Correct Harmonized System Code," in *2023 International Conference on Artificial Intelligence and Smart Communication (AISC)*, Jan 2023, pp. 35–41.

[4] M. Liao, L. Huang, J. Zhang, L. Song, and B. Li, "Enhanced HS Code Classification for Import and Export Goods via Multiscale Attention and ERNIE-BiLSTM," *Applied Sciences*, vol. 14, no. 22, 2024. [Online]. Available: https://www.mdpi.com/2076-3417/14/22/10267

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[6] Y. Kim, "Convolutional Neural Networks for Sentence Classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Oct 2014.

[7] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "XLNet: Generalized Autoregressive Pretraining for Language Understanding," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019. [Online]. Available: https://papers.nips.cc/paper_files/paper/2019/hash/dc6a7e655d7e5840e66733e9ee67cc69-Abstract.html

[8] R. Y. Rubinstein and D. P. Kroese, *The Cross Entropy Method: A Unified Approach To Combinatorial Optimization, Monte-carlo Simulation (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2004.

[9] Q. Team, "Qwen2.5 Technical Report," *arXiv preprint arXiv:2412.15115*, 2024.

[10] K. Kowsari, D. E. Brown, M. Heidarysafa, K. Jafari Meimandi, M. S. Gerber, and L. E. Barnes, "HDLTex: Hierarchical Deep Learning for Text Classification," in *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2017, pp. 364–371.

[11] D. D. Lewis, Y. Yang, T. G. Rose, and F. Li, "RCV1: A New Benchmark Collection for Text Categorization Research," *J. Mach. Learn. Res.*, vol. 5, pp. 361–397, Dec 2004.

[12] E. Sandhaus, "The New York Times Annotated Corpus," *Linguistic Data Consortium, Philadelphia*, vol. 6, no. 12, p. e26752, 2008.

[13] Z. Wang, P. Wang, L. Huang, X. Sun, and H. Wang, "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)*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 7109–7119. [Online]. Available: https://aclanthology.org/2022.acl-long.491/

[14] Z. Wang, P. Wang, T. Liu, B. Lin, Y. Cao, Z. Sui, and H. Wang, "HPT: Hierarchy-aware Prompt Tuning for Hierarchical Text Classification," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, Dec 2022, pp. 3740–3751. [Online]. Available: https://aclanthology.org/2022.emnlp-main.246/

[15] H. Zhu, J. Wu, R. Liu, Y. Hou, Z. Yuan, S. Li, Y. Pan, and K. Xu, "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)*, K. Duh, H. Gomez, and S. Bethard, Eds. Mexico City, Mexico: Association for Computational Linguistics, Jun 2024, pp. 4731–4745. [Online]. Available: https://aclanthology.org/2024.naacl-long.265/

[16] A. Kumar and D. Toshniwal, "HyILR: Hyperbolic Instance-Specific Local Relationships for Hierarchical Text Classification," in *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, J. Zhao, M. Wang, and Z. Liu, Eds. Vienna, Austria: Association for Computational Linguistics, Jul 2025, pp. 872–883. [Online]. Available: https://aclanthology.org/2025.acl-srw.63/

[17] F. Kong, R. Zhang, and Z. Wang, "LH-Mix: Local Hierarchy Correlation Guided Mixup over Hierarchical Prompt Tuning," in *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.1*, ser. KDD '25. New York, NY, USA: Association for Computing Machinery, 2025, pp. 636–646. [Online]. Available: https://doi.org/10.1145/3690624.3709326

[18] H. Peng, J. Li, Y. He, Y. Liu, M. Bao, L. Wang, Y. Song, and Q. Yang, "Large-Scale Hierarchical Text Classification with Recursively Regularized Deep Graph-CNN," in *Proceedings of the 2018 World Wide Web Conference*, ser. WWW '18. Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee, 2018, pp. 1063–1072. [Online]. Available: https://doi.org/10.1145/3178876.3186005

[19] K. Shimura, J. Li, and F. Fukumoto, "HFT-CNN: Learning Hierarchical Category Structure for Multi-label Short Text Categorization," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, Eds. Brussels, Belgium: Association for Computational Linguistics, Oct 2018, pp. 811–816. [Online]. Available: https://aclanthology.org/D18-1093/

[20] S. Banerjee, C. Akkaya, F. Perez-Sorrosal, and K. Tsioutsiouliklis, "Hierarchical Transfer Learning for Multi-label Text Classification," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, A. Korhonen, D. Traum, and L. Màrquez, Eds. Florence, Italy: Association for Computational Linguistics, Jul 2019, pp. 6295–6300. [Online]. Available: https://aclanthology.org/P19-1633/

[21] J. Wehrmann, R. Cerri, and R. Barros, "Hierarchical Multi-Label Classification Networks," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 5075–5084. [Online]. Available: https://proceedings.mlr.press/v80/wehrmann18a.html

[22] J. Zhou, C. Ma, D. Long, G. Xu, N. Ding, H. Zhang, P. Xie, and G. Liu, "Hierarchy-Aware Global Model for Hierarchical Text Classification," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds. Association for Computational Linguistics, Jul 2020, pp. 1106–1117. [Online]. Available: https://aclanthology.org/2020.acl-main.104/

[23] H. Chen, Q. Ma, Z. Lin, and J. Yan, "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)*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds. Online: Association for Computational Linguistics, Aug 2021, pp. 4370–4379. [Online]. Available: https://aclanthology.org/2021.acl-long.337/

[24] C. Yu, Y. Shen, and Y. Mao, "Constrained Sequence-to-Tree Generation for Hierarchical Text Classification," in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '22. New York, NY, USA: Association for Computing Machinery, 2022, pp. 1865–1869. [Online]. Available: https://doi.org/10.1145/3477495.3531765

[25] P. Yang, X. Sun, W. Li, S. Ma, W. Wu, and H. Wang, "SGM: Sequence Generation Model for Multi-label Classification," in *Proceedings of the 27th International Conference on Computational Linguistics*, E. M. Bender, L. Derczynski, and P. Isabelle, Eds. Santa Fe, New Mexico, USA: Association for Computational Linguistics, Aug 2018, pp. 3915–3926. [Online]. Available: https://aclanthology.org/C18-1330/

[26] W. Huang, C. Liu, B. Xiao, Y. Zhao, Z. Pan, Z. Zhang, X. Yang, and G. Liu, "Exploring Label Hierarchy in a Generative Way for Hierarchical Text Classification," in *Proceedings of the 29th International Conference on Computational Linguistics*, N. Calzolari, C.-R. Huang, H. Kim, J. Pustejovsky, L. Wanner, K.-S. Choi, P.-M. Ryu, H.-H. Chen, L. Donatelli, H. Ji, S. Kurohashi, P. Paggio, N. Xue, S. Kim, Y. Hahm, Z. He, T. K. Lee, E. Santus, F. Bond, and S.-H. Na, Eds. Gyeongju, Republic of Korea: International Committee on Computational Linguistics, Oct 2022, pp. 1116–1127.

[27] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," *J. Mach. Learn. Res.*, vol. 21, no. 1, Jan 2020.

[28] B. Ning, D. Zhao, X. Zhang, C. Wang, and S. Song, "UMP-MG: A Uni-directed Message-Passing Multi-label Generation Model for Hierarchical Text Classification," *Data Science and Engineering*, vol. 8, no. 2, pp. 112–123, Jun 2023.

[29] S. Im, G. Kim, H.-S. Oh, S. Jo, and D. H. Kim, "Hierarchical text classification as sub-hierarchy sequence generation," in *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023. [Online]. Available: https://doi.org/10.1609/aaai.v37i11.26520

[30] V. Jain, M. Rungta, Y. Zhuang, Y. Yu, Z. Wang, M. Gao, J. Skolnick, and C. Zhang, "HiGen: Hierarchy-Aware Sequence Generation for Hierarchical Text Classification," in *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, Y. Graham and M. Purver, Eds. St. Julian's, Malta: Association for Computational Linguistics, Mar 2024, pp. 1354–1368. [Online]. Available: https://aclanthology.org/2024.eacl-long.82/

[31] R. You, Z. Zhang, Z. Wang, S. Dai, H. Mamitsuka, and S. Zhu, "AttentionXML: Label Tree-based Attention-Aware Deep Model for High-Performance Extreme Multi-Label Text Classification," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2019/file/9e6a921fbc428b5638b3986e365d4f21-Paper.pdf

[32] Z. Deng, H. Peng, D. He, J. Li, and P. Yu, "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*, K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, Eds. Online: Association for Computational Linguistics, Jun 2021, pp. 3259–3265. [Online]. Available: https://aclanthology.org/2021.naacl-main.260/

[33] T. Jiang, D. Wang, L. Sun, Z. Chen, F. Zhuang, and Q. Yang, "Exploiting Global and Local Hierarchies for Hierarchical Text Classification," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, Dec 2022, pp. 4030–4039. [Online]. Available: https://aclanthology.org/2022.emnlp-main.268/

[34] Y. Liu, K. Zhang, Z. Huang, K. Wang, Y. Zhang, Q. Liu, and E. Chen, "Enhancing Hierarchical Text Classification through Knowledge Graph Integration," in *Findings of the Association for Computational Linguistics: ACL 2023*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul 2023, pp. 5797–5810. [Online]. Available: https://aclanthology.org/2023.findings-acl.358/

[35] H. Zhu, C. Zhang, J. Huang, J. Wu, and K. Xu, "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)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul 2023, pp. 7809–7821. [Online]. Available: https://aclanthology.org/2023.acl-long.432/

[36] S. C. L. Yu, J. He, V. G. Basulto, and J. Z. Pan, "Instances and Labels: Hierarchy-aware Joint Supervised Contrastive Learning for Hierarchical Multi-Label Text Classification," in *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. [Online]. Available: https://openreview.net/forum?id=S0eqbM16k2

[37] J. Zhang, Y. Li, F. Shen, C. Xia, H. Tan, and Y. He, "Hierarchy-Aware and Label Balanced Model for Hierarchical Text Classification," *Knowledge-Based Systems*, vol. 300, p. 112153, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705124007871

[38] G. Kim, S. Im, and H.-S. Oh, "Hierarchy-aware Biased Bound Margin Loss Function for Hierarchical Text Classification," in *Findings of the Association for Computational Linguistics: ACL 2024*, L.-W. Ku, A. Martins, and V. Srikumar, Eds. Bangkok, Thailand: Association for Computational Linguistics, Aug 2024, pp. 7672–7682. [Online]. Available: https://aclanthology.org/2024.findings-acl.457/

[39] Z. Wang, P. Wang, and H. Wang, "Utilizing Local Hierarchy with Adversarial Training for Hierarchical Text Classification," in *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, and N. Xue, Eds. Torino, Italia: ELRA and ICCL, May 2024, pp. 17 326–17 336.

[40] J. Zhou, L. Zhang, Y. He, R. Fan, L. Zhang, and J. Wan, "A Novel Negative Sample Generation Method for Contrastive Learning in Hierarchical Text Classification," in *Proceedings of the 31st International Conference on Computational Linguistics*, O. Rambow, L. Wanner, M. Apidianaki, H. Al-Khalifa, B. D. Eugenio, and S. Schockaert, Eds. Abu Dhabi, UAE: Association for Computational Linguistics, Jan 2025, pp. 5645–5655. [Online]. Available: https://aclanthology.org/2025.coling-main.378/

[41] K. Ji, Y. Lian, J. Gao, and B. Wang, "Hierarchical Verbalizer for Few-Shot Hierarchical Text Classification," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul 2023, pp. 2918–2933. [Online]. Available: https://aclanthology.org/2023.acl-long.164/

[42] F. Cai, D. Liu, Z. Zhang, G. Liu, X. Yang, and X. Fang, "NER-guided Comprehensive Hierarchy-aware Prompt Tuning for Hierarchical Text Classification," in *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, N. Calzolari, M.-Y. Kan, V. Hoste, A. Lenci, S. Sakti, and N. Xue, Eds. Torino, Italia: ELRA and ICCL, May 2024, pp. 12 117–12 126. [Online]. Available: https://aclanthology.org/2024.lrec-main.1060/

[43] H. Chen, Y. Zhao, Z. Chen, M. Wang, L. Li, M. Zhang, and M. Zhang, "Retrieval-style In-context Learning for Few-shot Hierarchical Text Classification," *Transactions of the Association for Computational Linguistics*, vol. 12, pp. 1214–1231, 2024. [Online]. Available: https://aclanthology.org/2024.tacl-1.67/

[44] S. Xiong, Y. Zhao, J. Zhang, L. Mengxiang, Z. He, X. Li, and S. Song, "Dual prompt tuning based contrastive learning for hierarchical text classification," in *Findings of the Association for Computational Linguistics: ACL 2024*, L.-W. Ku, A. Martins, and V. Srikumar, Eds. Bangkok, Thailand: Association for Computational Linguistics, Aug 2024, pp. 12 146–12 158. [Online]. Available: https://aclanthology.org/2024.findings-acl.723/

[45] Y. Zhang, R. Yang, X. Xu, R. Li, J. Xiao, J. Shen, and J. Han, "TELEClass: Taxonomy Enrichment and LLM-Enhanced Hierarchical Text Classification with Minimal Supervision," in *Proceedings of the ACM on Web Conference 2025*, ser. WWW '25. New York, NY, USA: Association for Computing Machinery, 2025, pp. 2032–2042. [Online]. Available: https://doi.org/10.1145/3696410.3714940

[46] S. Chen, M. R. Bouadjenek, U. Naseem, B. Suleiman, S. Jameel, F. Salim, H. Hacid, and I. Razzak, "Leveraging Taxonomy and LLMs for Improved Multimodal Hierarchical Classification," in *Proceedings of the 31st International Conference on Computational Linguistics*, O. Rambow, L. Wanner, M. Apidianaki, H. Al-Khalifa, B. D. Eugenio, and S. Schockaert, Eds. Abu Dhabi, UAE: Association for Computational Linguistics, Jan 2025, pp. 6244–6254. [Online]. Available: https://aclanthology.org/2025.coling-main.417/

[47] Q. Zang, C. Zgrzendek, I. Tchappi, A. Khadangi, and J. Sedlmeir, "KG-HTC: Integrating Knowledge Graphs into LLMs for Effective Zero-shot Hierarchical Text Classification," *arXiv preprint arXiv:2505.05583*, 2025.

[48] P. Zhao, Q. Zheng, B. Dong, J. Ruan, and M. Luo, "Tax Classification of Invoice Details Based on Directed Heterogeneous Graph," in *Proceedings of the 19th Chinese National Conference on Computational Linguistics*, M. Sun, S. Li, Y. Zhang, and Y. Liu, Eds. Haikou, China: Chinese Information Processing Society of China, Oct 2020, pp. 771–782. [Online]. Available: https://aclanthology.org/2020.ccl-1.72/

[49] O. Amel, S. Stassin, S. A. Mahmoudi, and X. Siebert, "Multimodal Approach for Harmonized System Code Prediction," in *ESANN 2023 Proceesdings*. Louvain-la-Neuve (Belgium): Ciaco - i6doc.com, 2023.

[50] A. W. Anggoro, P. Corcoran, D. De Widt, and Y. Li, "Harmonized system code classification using supervised contrastive learning with sentence bert and multiple negative ranking loss," *Data Technologies and Applications*, vol. 59, no. 2, pp. 276–301, 12 2024. [Online]. Available: https://doi.org/10.1108/DTA-01-2024-0052