Semantic Refinement and Hierarchy-Aware Inference for Weakly Supervised Multi-Label Text Classification

YONGJOO KIM, Korea University

This study addresses the challenge of Hierarchical Multi-Label Text Classification (H-MLTC) in a transductive setting with no labeled training data. We propose a robust self-training framework that integrates Large Language Models (LLMs) for semantic class enrichment and synthetic data generation. Our approach proceeds in distinct phases: (1) semantic initialization using MPNet and LLM-generated descriptions, (2) high-precision silver label generation via a gap-based heuristic, and (3) a refinement phase that addresses extreme class imbalance through "starved class" augmentation and focal loss. Furthermore, we demonstrate that a top-down beam search inference strategy significantly outperforms bottom-up approaches by enforcing taxonomic consistency. Our method achieves a Sample F1 score of 0.63578 on the Amazon Product Review dataset[1], showing substantial improvement over the baseline (0.39180). We also analyze the limitations of Graph Neural Networks (GNNs) in this specific semantic space, providing empirical evidence of reduced distinctiveness. Our code and data are available at: https://github.com/All4Nothing/20252R0136DATA30400.

Additional Key Words and Phrases: Hierarchical Classification, Weakly Supervised Learning, Data Augmentation, LLM, Self-Training

1 Introduction

Hierarchical Multi-Label Text Classification (H-MLTC) is a complex task where documents must be assigned to multiple categories organized in a taxonomy. This project specifically tackles a scenario where no ground-truth labels are provided for the training corpus, requiring the effective utilization of class hierarchies, keywords, and unlabeled text.

We propose a multi-stage framework inspired by TELEClass[4], enhanced with a data-centric refinement strategy. Our key contributions are:

- **Semantic Representation:** Utilizing MPNet[3] with LLM-enriched class descriptions to establish a robust initial embedding space.
- **Gap-Based Silver Labeling:** Introducing a dynamic thresholding mechanism to generate high-quality pseudo-labels for self-training.
- **Starved Class Augmentation:** Identifying and augmenting under-represented classes using LLMs with style calibration to match the domain distribution.
- **Hierarchy-Aware Inference:** Proving the effectiveness of top-down beam search over bottom-up reconstruction for final predictions.

To ensure reproducibility, we release our implementation and trained models publicly.[1]

2 Problem Definition

The task is defined as learning a mapping function $f : \mathcal{D} \rightarrow \{0, 1\}^{|C|}$, where $\mathcal{D}$ is a corpus of product reviews and C is a set of 531 classes organized in a Directed Acyclic Graph (DAG) taxonomy $\mathcal{T}$.

- **Input:** Unlabeled training corpus ($N = 29,487$), Test corpus ($N = 19,658$), Class Taxonomy, and Keywords.
- **Constraint:** Transductive learning is permitted; the test corpus can be used for representation learning.
- **Metric:** Sample F1 Score.

[1]Code available at: https://github.com/All4Nothing/20252R0136DATA30400

Author's Contact Information: Yongjoo Kim, Korea University, DATA304: Big Data Analysis.

3 Methodology

3.1 Phase 1: Semantic Initialization

Unlike traditional methods that rely solely on keywords, we leverage GPT-4o-mini to generate rich semantic descriptions for each class.

$$E_{class} = \text{MPNet}(\text{Description}(c_i)) \tag{1}$$

We utilize ‘sentence-transformers/all-mpnet-base-v2‘ to encode both documents and class descriptions. Crucially, we initialize the classification head of our model with these pre-computed class embeddings to ensure semantic consistency between the initialization and training phases.

3.2 Phase 2: Gap-Based Pseudo-Labeling

To generate reliable silver labels for training, we employ a **gap-based heuristic**. For each document, we calculate cosine similarity scores against all classes. Instead of applying a fixed threshold (e.g., > 0.5), we assign labels based on the score gap:

$$y_{silver} = \{c_1 \mid Score(d, c_1) - Score(d, c_2) > \delta\} \tag{2}$$

where c_1 and c_2 are the classes with the highest and second-highest similarity scores for document d, and δ is the dynamic threshold. This strategy ensures that we only assign labels when the model is confident about the distinction between the target class and potential negatives, minimizing error propagation in the self-training loop. Note that in the transductive setting, we generated pseudo-labels for both the unlabeled training corpus and the test corpus, allowing the model to learn representations from the target distribution directly.

3.3 Analysis of Pseudo-Label Distribution

Before proceeding to model training, we analyzed the distribution of the generated silver labels. Figure 1 illustrates the extreme class imbalance inherent in the dataset.

- **Long-Tail Distribution:** While the mean number of documents per class is 148.6, the median is only 83.0, indicating a heavy right-skewed distribution.
- **Starved Classes:** We identified **75 classes** (approx. 14% of total) as "starved," having fewer than 15 documents. Notably, 4 classes (e.g., *breakfast_cereal_bars*, *breads*) received zero documents, making them impossible to learn without synthetic augmentation.
- **Dominant Classes:** Conversely, the top 5% of classes (e.g., *fragrance* with 1,834 docs) occupy a disproportionate amount of the training signal, necessitating the use of Focal Loss to prevent bias.

3.4 Phase 3: Data-Centric Refinement

A major challenge in weakly supervised learning is the "Starved Class" problem. Analyzing the pseudo-labels from Phase 2, we identified 75 classes with fewer than 15 assigned documents. To mitigate this imbalance:

(1) **Synthetic Generation:** We prompted an LLM to generate 5 synthetic reviews for each starved class.
(2) **Style Calibration:** Synthetic reviews generated by LLMs often exhibit distinct linguistic patterns (e.g., perfect grammar, lack of colloquialisms). To bridge the domain gap, we applied a rule-based calibration (e.g., simulating tokenization artifacts like "ca n’t") to align the synthetic text distribution with the noisy Amazon review corpus.
(3) **Scale Adjustment:** The augmentation scale was fixed to 1.0 to prevent synthetic data from dominating the loss.

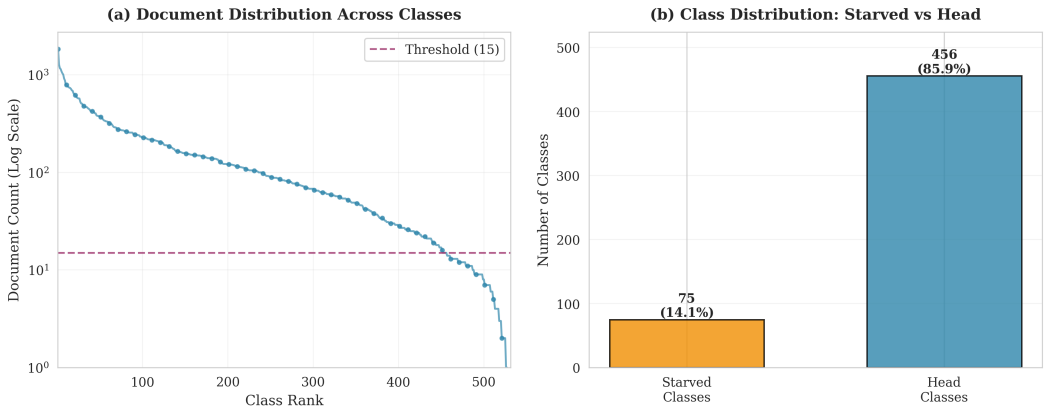

Fig. 1. Comprehensive analysis of pseudo-label distribution. The dashboard reveals severe class imbalance through multiple views: (Top-Left) The right-skewed histogram showing a median of only 83 documents; (Top-Right) Extreme outliers in document counts; (Bottom-Right) The dominance of categories like ‘fragrance‘, contrasting with 75 ‘starved classes‘.

3.5 Phase 4: Training with Custom Focal Loss

We train a classifier initialized with the MPNet backbone using the augmented dataset. To handle the extreme imbalance (1 positive vs. 500 negatives) and the noise in pseudo-labels, we implemented a Custom Focal Loss:

$$L = -\alpha(1 - p_t)^{\gamma} \log(p_t) \tag{3}$$

where $\gamma = 2.0$ focuses learning on hard examples. Additionally, we made the temperature parameter τ and logit scale trainable, allowing the model to dynamically adjust the sharpness of its predictions during training.

3.6 Phase 5: Hierarchy-Aware Inference

For the final prediction, we specifically adopt a **Top-Down Beam Search** strategy. We start from the root and recursively expand children only if the parent’s score exceeds a dynamic threshold. This strictly enforces the taxonomic constraint (a document cannot belong to a child class without belonging to its parent), which our experiments proved to be superior to bottom-up approaches.

4 Experimental Setup

All experiments were conducted with a fixed random seed (42) for reproducibility. We used the ‘WeightedMultiLabelDataset‘ with a 9:1 train-validation split. The model was trained for 5 epochs with a batch size of 16 using the AdamW optimizer (learning rate 2e-5) and a cosine annealing scheduler.

5 Results and Analysis

5.1 Quantitative Analysis

We evaluated our framework against baseline methods and ablation variants using the Sample F1 score. As a primary baseline, we implemented the **TaxoClass** framework[2] (Attempt 1), which represents a standard self-training approach using BERT and keywords.

Table 1 summarizes the performance comparison. Our final model achieves a Sample F1 score of **0.63578**, significantly outperforming the baseline (0.3918) and the basic TELEClass implementation (0.4418).

Table 1. Main Results on Amazon-531 Dataset. We compare our method against the self-training baseline. "Top-down" indicates our proposed inference strategy.

Method	Strategy	Sample-F1
Baselines		
TaxoClass-Impl (Attempt 1)	Self-Training + BERT	0.39180
TELEClass-Basic (Attempt 2)	Hierarchy-Guided + Keywords	0.44180
Ablation Study (Ours)		
Ours (w/o Augmentation)	MPNet + LLM Desc	0.47198
Ours (w/o Style Calib)	+ Starved Class Aug	0.59888
Ours (w/ GNN)	+ Graph Convolution	0.62553
Ours (Final)	**Refinement + Top-Down**	**0.63578**

5.2 Limitation of GNN-based Refinement

Contrary to our initial hypothesis, the integration of Graph Convolutional Networks (GNN) did not yield significant performance gains compared to the hierarchy-aware inference strategy. We implemented a two-layer GCN with strong residual connections to refine class embeddings based on the taxonomy $\mathcal{T}$. However, our ablation study and embedding analysis reveal that this approach suffered from *semantic over-smoothing*, leading to a loss of discriminative margins between sibling classes.

Quantitative Analysis of Representation Collapse. To investigate the cause, we analyzed the intra-level cosine similarity of the class embeddings (Figure 2). As shown in our statistical analysis, the GNN aggregation mechanism (‘torch.mm(adj, support)‘) caused the representations of distinct sibling nodes to converge. Specifically, for Level 1 nodes (e.g., *feeding* vs. *diapering*), the average cosine similarity surged from **0.5429 to 0.6494** after GNN application. Similarly, for Level 2 nodes, similarity increased from **0.4037 to 0.5393**.

Analysis. Ideally, sibling nodes should remain distinct to capture fine-grained differences. However, the message-passing mechanism of the GNN smoothed out the unique features of each category, blending them with their neighbors. This *over-smoothing* reduced the semantic gap required for the model to distinguish between closely related labels (e.g., *Baby Food* vs. *Baby Cereal*). Consequently, we found that explicitly enforcing hierarchical constraints during the inference phase (Top-Down Beam Search) was more effective than implicitly attempting to encode structure via graph convolutions in this high-density semantic space.

5.3 Impact of Inference Strategy

To quantify the benefit of hierarchy-aware inference, we compared our **Top-Down Beam Search** against a **Leaf-Priority** (Bottom-Up) strategy, which selects high-confidence leaf nodes first and reconstructs the lineage.

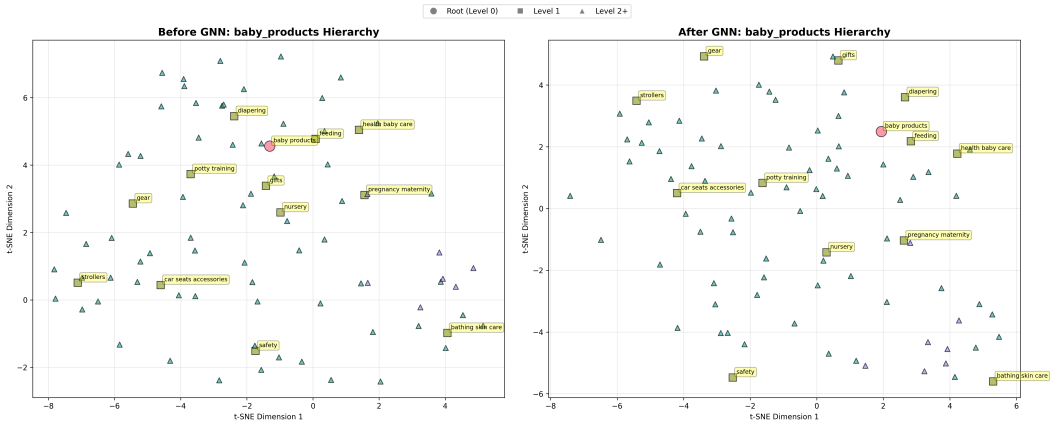

Fig. 2. **Visualization of Representation Collapse.** t-SNE plots of the 'Baby Products' sub-hierarchy (87 nodes) before and after GNN application. The *After GNN* plot demonstrates a denser clustering of nodes, indicating that distinct class representations have collapsed into a generic embedding space, blurring the boundaries between sibling categories.

- **Leaf-Priority Strategy:** Achieved an F1 score of **0.58217**. This approach often generated inconsistent paths where a leaf node was predicted with high confidence, but its parent nodes had low scores, leading to logical contradictions in the taxonomy.
- **Top-Down Beam Search (Ours):** Achieved an F1 score of **0.63578**. By strictly enforcing the parent-child dependency (a child node is only considered if its parent is active), we achieved a **9.1% relative improvement**. This confirms that explicit taxonomic constraints during inference are more effective than relying solely on the model's raw probabilities.

5.4 Case Study and Qualitative Analysis

To understand the model's behavior and limitations, we analyze specific success and failure cases from the test corpus.

Success Case: Contextual Understanding.

- **Input Document:** "barbie ballet shoes icon doll ... found this cheap doll! ... make a cardboard dancefloor ... her shoes are very pretty! pink with laces! her hair is very good quality..."
- **Predicted Path:** `toys_games` → `dolls_accessories` → `doll_accessories`
- **Analysis:** The model successfully navigated the hierarchy down to the specific leaf node `doll_accessories`. Despite the conversational noise ("wow", "get her!"), the model correctly focused on specific attributes like "shoes", "painted body", and "hair", distinguishing it from generic dolls.

Failure Case: Semantic Ambiguity (The "Flavor" Trap).

- **Input Document:** "my lip stuff tube over 600 lipbalm flavors available very good product! this smells good and feels very soft on your lips. i would buy one in every flavor!"
- **Predicted Path:** `candy_chocolate` → `chewing_gum`
- **Analysis:** This failure illustrates the challenge of **semantic overlap**. The review is laden with confectionery-related keywords like "flavors" and "smells good". The Top-down search

was misled into the `candy_chocolate` branch early on and could not recover, highlighting a limitation of greedy hierarchical decoding when "bridge keywords" (like flavor) are ambiguous.

6 Conclusion

We successfully developed a robust H-MLTC framework without labeled data, achieving an F1 score of 0.63578. Our findings highlight that data-centric refinements—specifically addressing "starved classes" and calibrating synthetic text style—yielded larger gains than architectural changes. Furthermore, we demonstrated that explicit hierarchy is better enforced through inference constraints (Top-down search) rather than embedding-level graph convolutions (GNNs) in this semantic space.

References

[1] Ruining He and Julian McAuley. 2016. Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering. In *Proceedings of the 25th International Conference on World Wide Web*. 507–517.

[2] Jiaming Shen, Wenda Qiu, Yu Shang, Michelle Zeng, C Huang, Jiawei andSUB, and Jiawei Han. 2021. TaxoClass: Hierarchical Multi-Label Text Classification Using Only Class Names. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 4239–4249.

[3] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. MPNet: Masked and Permuted Pre-training for Language Understanding. In *Advances in Neural Information Processing Systems*, Vol. 33. 16857–16867.

[4] Yunyi Zhang, Ruirui Yuan, Yd He, Dong He, Xinyu Wu, W Zhang, and M Wang. 2024. TELEClass: Taxonomy Enrichment and LLM-Enhanced Hierarchical Text Classification with Minimal Supervision. *arXiv preprint arXiv:2403.00165* (2024).