

Multi-Label Classification in Hierarchical Taxonomies (Taxonomy-Aware ML)

Mano R

Master of Computer Applications, Nehru Institute of Information Technology and Management,
Tamil Nadu, India. Reg. No: 721524622031 | Batch: 2024-2026 |

AUTHOUR: **Dr. S. Menaka MCA**, M.Phil., Ph.D., Head of Department

ABSTRACT

The exponential growth of digital data across domains including scientific publishing, e-commerce, healthcare, and legal documentation has created an urgent need for accurate and structured automated classification systems. Many such domains organize their knowledge in hierarchical taxonomies, where categories exist in a tree or directed acyclic graph structure with parent-child relationships. Traditional flat multi-label classification algorithms are inefficient and logically invalid when applied to such hierarchically structured label spaces, as they treat all labels as separate and equal, completely disregarding the crucial semantic relationships encoded in the taxonomy.

The proposed system introduces a Taxonomy-Aware Machine Learning Framework that explicitly incorporates hierarchical relationships into the classification process. The system employs the Local Classifier per Parent Node (LCPN) approach, where a dedicated Logistic Regression classifier is trained for each parent node in the hierarchy. Feature extraction is performed using Term Frequency-Inverse Document Frequency (TF-IDF) vectorization. Experimental results demonstrate significant improvements in hierarchical precision, recall, F1-score, and taxonomy compliance compared to conventional flat classification techniques.

Keywords: Multi-Label Classification, Hierarchical Taxonomy, LCPN, TF-IDF, Logistic Regression, Taxonomy-Aware ML, Scikit-Learn

I. INTRODUCTION

In the modern era of data-driven decision-making, the ability to accurately classify information into structured categories is of paramount importance. Many real-world applications, ranging from scientific literature indexing to e-commerce product management, require classifying items into multiple categories simultaneously. This challenge is known as Multi-Label

Classification. When these categories are organized in a hierarchical taxonomy — where broader parent categories contain more specific child categories — the problem becomes significantly more complex.

Traditional classification systems primarily rely on flat classification methods where all labels are treated as independent and equal. These systems fail to capture the parent-child dependencies present in hierarchical label spaces, often producing logically invalid predictions — such as assigning a highly specific child label without also assigning its more general parent label.

To overcome these limitations, this project develops a Taxonomy-Aware Machine Learning system titled "Multi-Label Classification in Hierarchical Taxonomies." The system is built using Python and a suite of powerful libraries including scikit-learn, pandas, NumPy, Matplotlib, and Seaborn. The core classification strategy is the Local Classifier per Parent Node (LCPN) approach, where a dedicated Logistic Regression classifier is trained for each parent node to predict among its direct children. The application has wide-ranging use cases including academic article classification, bioinformatics label annotation, legal document categorization, and product taxonomy management.

II. RELATED WORK

Multi-Label Classification (MLC) in hierarchical settings has attracted significant research attention due to its applicability across diverse real-world domains. Traditional methods include Binary Relevance (BR), Label Powerset (LP), and Classifier Chains (CC). Binary Relevance transforms the multi-label problem into multiple independent binary classification tasks, but ignores label correlations and hierarchy information. Label Powerset considers label combinations as unique classes, but suffers from scalability issues. Classifier Chains attempt to model label dependencies but do not explicitly enforce hierarchical consistency.

Recent deep learning approaches such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Transformer-based architectures have improved feature extraction capabilities. Nevertheless, many of these models still treat the label hierarchy as an external component and rely on post-processing techniques to correct inconsistent predictions.

The proposed Taxonomy-Aware ML framework extends existing research by directly embedding hierarchical knowledge into the learning process using the LCPN strategy, eliminating the need for post-hoc correction and producing logically consistent predictions across all levels of the taxonomy.

III. SYSTEM OVERVIEW

This project, titled "Multi-Label Classification in Hierarchical Taxonomies (Taxonomy-Aware ML)," aims to design and develop an intelligent classification system that respects the hierarchical structure of labels, ensuring that predictions are always logically consistent with the taxonomy. The system is built using Python and integrates the LCPN strategy with TF-IDF feature extraction and Logistic Regression classifiers.

The application allows users to: process raw textual datasets from CSV files; perform TF-IDF vectorization to extract meaningful features; apply the LCPN hierarchical classification pipeline; validate predictions against taxonomic constraints; and visualize results through performance metrics, confusion matrices, and interactive taxonomy maps. Once processed, results are presented in an interactive Streamlit-based dashboard featuring taxonomy visualizations, performance gauges, and prediction outputs.

The system has applications across domains such as:

- Academic Literature and Document Categorization
- E-Commerce Product Taxonomy Management
- Bioinformatics Label Annotation
- Legal Document Classification and Knowledge Management

The final output is a lightweight, open-source, web-based application built on Streamlit, scikit-learn, and Plotly, suitable for both researchers and domain experts.

IV. MODULE DESCRIPTION

1. Data Ingestion and Taxonomy Parsing Module

This module loads and validates raw textual datasets from CSV files using Pandas. The label hierarchy is parsed separately to extract parent-child relationships and build the taxonomy graph. The module handles missing values and prepares structured data arrays for subsequent processing stages.

2. Feature Extraction Module

The Feature Extraction Module applies TF-IDF (Term Frequency-Inverse Document Frequency) Vectorization to transform raw text into numerical feature matrices. TF-IDF calculates term weights that highlight discriminative features while suppressing common uninformative terms, producing consistent and meaningful feature representations for all documents.

3. LCPN Classification Module

The LCPN (Local Classifier per Parent Node) module forms the core intelligence of the system. A dedicated Logistic Regression classifier is trained for each parent node in the taxonomy, responsible for making classification decisions among its direct child nodes. This top-down traversal approach respects the taxonomic structure and ensures all predictions are hierarchically valid.

4. Hierarchical Consistency Enforcement Module

This module validates all output predictions against the taxonomic constraints. It ensures that no child label is assigned without its corresponding parent label, producing a final clean, hierarchically consistent label set for each input document. This eliminates the fundamental flaw of conventional flat classifiers.

5. Evaluation and Visualization Module

The Evaluation and Visualization Module computes precision, recall, F1-score, and accuracy using scikit-learn's evaluation suite. It generates confusion matrices, performance curves, co-occurrence frequency heatmaps, and interactive taxonomy hierarchy graphs. The Streamlit dashboard provides visual comparisons between the taxonomy-aware system and flat baseline approaches.

V. METHODOLOGY

The methodology of the Taxonomy-Aware ML system consists of multiple stages including data acquisition, feature extraction, hierarchical classification, consistency enforcement, and visualization.

Initially, the 20 Newsgroups dataset is loaded and processed using Pandas. The dataset contains news articles spanning 20 topical categories. Labels are organized into a five-level parent taxonomy comprising COMP, REC, SCI, TALK, and MISC categories. The raw textual data is vectorized using TF-IDF with a vocabulary of 5,000 most discriminative features.

The extracted features are analyzed using the LCPN classification strategy. A dedicated Logistic Regression classifier (with OneVsRestClassifier wrapper) is trained for each parent node, learning to distinguish among its direct child categories. The top-down traversal ensures that predictions at deeper levels are always conditioned on valid parent predictions, enforcing hierarchical consistency without requiring post-processing.

Detected classifications are validated through the Hierarchical Consistency Enforcement Module, which guarantees parent-child constraint satisfaction. Results are then visualized through the Streamlit dashboard using line graphs, taxonomy treemaps, confusion matrices, label co-occurrence heatmaps, and performance gauge charts. The evaluation results showed a Hamming Loss of 0.047, Precision of 89.5%, Recall of 45.7%, and F1-Score of 58.7%.

VI. REQUIREMENTS SPECIFICATION

Software Requirements

SOFTWARE COMPONENT	SPECIFICATION
Operating System	Windows 10/11 (64-bit)
Programming Language	Python 3.12.0
IDE	Jupyter Notebook / VS Code
ML Library	Scikit-Learn (LCPN, Logistic Regression)
Feature Extraction	TF-IDF Vectorizer (scikit-learn)
Data Manipulation	NumPy & Pandas
Visualization Tools	Matplotlib, Seaborn & Plotly
Web Framework	Streamlit (version 1.20+)

Hardware Requirements

HARDWARE COMPONENT	SPECIFICATION
Processor	Intel Core i3 or i5 or higher
RAM	8 GB or Above
Hard Disk	500 GB
Monitor	15" LED Color Monitor
Input Devices	Standard Keyboard & Mouse
Network Connection	High-speed Internet

VII. SYSTEM IMPLEMENTATION

System implementation involved converting the theoretical taxonomy-aware design into a fully functional Python-based pipeline deployed via a Streamlit web application. The implementation proceeded through five key phases:

Environment Setup: All required libraries were installed including scikit-learn, pandas, NumPy, Matplotlib, Seaborn, Streamlit, and Plotly. The development environment was configured in Jupyter Notebook and VS Code with Python 3.12.

Module Development: Each functional module was implemented in Python. The data ingestion module handles CSV loading and taxonomy parsing. The TF-IDF vectorizer was configured and fitted on the 20 Newsgroups training corpus. Individual LCPN Logistic Regression classifiers were trained for each of the five parent taxonomy nodes (comp, rec, sci, talk, misc).

Hierarchical Prediction Engine: The top-down prediction algorithm traverses the taxonomy from root to leaf, making local classification decisions at each parent node and enforcing parent-child consistency automatically.

Streamlit Interface Integration: An interactive real-time dashboard was developed with three tabs — Predict, Taxonomy Visualizer, and Performance Metrics — featuring example inputs, confidence sliders, interactive treemaps, and performance gauge charts.

Testing and Debugging: Unit and integration testing were conducted on each module. Edge cases including single-node taxonomies, missing labels, and imbalanced classes were validated. The system was

deployed locally via streamlit run app.py or hosted on Streamlit Cloud.

VIII. SAMPLE OUTPUT



Predict



Interactive Taxonomy Map



Performance Metrics



Prediction Result

IX. CONCLUSIONS

The project "Multi-Label Classification in Hierarchical Taxonomies (Taxonomy-Aware ML)" has been successfully designed and developed as a robust and semantically coherent solution for the automated

classification of data into hierarchically structured label spaces.

By implementing the Local Classifier per Parent Node (LCPN) strategy with Logistic Regression and TF-IDF feature extraction, the system effectively models parent-child relationships within the taxonomy and guarantees that all predictions are hierarchically consistent. This eliminates the fundamental flaw of conventional flat classifiers that produce logically invalid label assignments.

The empirical results confirm that the taxonomy-aware approach not only ensures logical validity but also enhances overall predictive performance — particularly for rare and fine-grained labels that benefit from sharing statistical strength with their parent categories. The high precision score of 89.5% demonstrates the reliability of the LCPN predictions. The interactive Streamlit dashboard ensures complex technical outputs are accessible to both expert researchers and non-technical users.

Overall, this project demonstrates that incorporating domain knowledge in the form of hierarchical taxonomy structure into the machine learning pipeline leads to more reliable, interpretable, and semantically meaningful classification systems, readily applicable to academic literature classification, product taxonomy management, bioinformatics annotation, and legal document categorization.

ACKNOWLEDGEMENT

The author would like to express sincere gratitude to the faculty of the Department of Master of Computer Applications at Nehru Institute of Information Technology and Management, Coimbatore, Tamil Nadu, for their valuable guidance and support during the course of this project. Special thanks are extended to Dr. S. Menaka, Head of the Department, for her constant encouragement, and to Ms. A. Pavithra, Senior Python Developer at VCODEZ, for her invaluable external guidance. The author also gratefully acknowledges Techvolt Software Pvt. Ltd., Coimbatore, for providing the project opportunity and infrastructure support.

REFERENCES

1. Silla, C.N. and Freitas, A.A. (2011). A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, 22(1-2), pp.31-72.
2. Vens, C., Struyf, J., Schietgat, L., Dzeroski, S. and Blockeel, H. (2008). Decision trees for hierarchical multi-label classification. *Machine Learning*, 73(2), pp.185-214.
3. Zhang, M.L. and Zhou, Z.H. (2014). A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8), pp.1819-1837.
4. Read, J., Pfahringer, B., Holmes, G. and Frank, E. (2011). Classifier chains for multi-label classification. *Machine Learning*, 85(3), pp.333-359.
5. Pedregosa, F. et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, pp.2825-2830.
6. Shimura, K., Li, J. and Fukumoto, F. (2018). HFT-CNN: Learning Hierarchical Category Structure for Multi-label Short Text Categorization. *Proceedings of EMNLP 2018*, pp.811-816.