An Analysis of Public Datasets for Hierarchical Classification

Gustavo Vieira Maia
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
vmgustavo@ufmg.br

Frederico Gualberto Ferreira Coelho
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
fredgfc@ufmg.br

***Abstract*—Hierarchical classification is a machine learning task that leverages inherent parent-child relationships between class labels and offers advantages in predictive performance and interpretability over traditional ”flat” classification. Despite its potential, its adoption in domains other than text, image and biology is limited, partly due to a perceived scarcity of suitable public datasets. This study performs an investigation into the availability of hierarchical datasets within the UCI Machine Learning Repository and OpenML. We employed a novel methodology using Large Language Models to automatically classify the metadata of over 1200 candidate datasets, followed by manual verification of promising candidates. Our findings reveal a shortage of public tabular datasets suitable for hierarchical classification. Out of the entire collection, only three potential datasets were identified. This work quantifies the data scarcity problem, highlighting it as a significant bottleneck that hinders research, development, and the broader application of hierarchical modeling techniques. To the best of our knowledge, this is the first large-scale quantitative study of hierarchical classification dataset availability in major public repositories.**

***Keywords*—Machine Learning; Hierarchical Classification; Open Datasets; Data Science**

I. Introduction

Hierarchically organizable labeled multi-class or multi-label datasets can be modeled using hierarchical classification. Despite its advantages in predictive performance and interpretability, its use remains limited in tabular data domains. We link the scarcity of applications in tabular domains partly to the lack of tabular datasets that represent hierarchical contexts. This study addresses that gap by providing a large-scale systematic investigation of public tabular datasets for their suitability in training hierarchical classification models.

Using language models for semantic analysis of dataset descriptions, we examine two of the most widely used repositories to quantify the availability of hierarchical data: the University of California Irvine Machine Learning Repository (UCI) and OpenML. The findings offer a clearer understanding of the extent of this limitation, providing a foundation for future dataset creation, benchmarking, and methodological development.

The value of hierarchical classification lies in its ability to leverage inherent dependencies between labels to improve model performance and produce more interpretable, logically consistent predictions. A dataset may be considered hierarchical if the class labels are organized into a hierarchy, such as a tree or a Directed Acyclic Graph (DAG), where there are clear parent-child or superclass-subclass relationships and the next label in the hierarchy is also an observation of the previous label. By learning the relationships between parent and child classes, models can avoid illogical outcomes that may occur in flat classification systems. For example, a well-structured hierarchical model could learn that if an animal is identified as a Tiger, it must also be classified as a Feline and a Mammal, inheriting the broader taxonomic characteristics associated with those categories. This hierarchical reasoning improves consistency across prediction levels, ensuring that the assignment of a highly specific label automatically implies the presence of all relevant, more general labels in its lineage.

Yet, the vast majority of machine learning applications are still framed as ”flat” classification problems, where models perform multi-class or multi-label classification without considering label dependencies. The preference for flat structures is reinforced by the dataset ecosystem itself: most open datasets, especially those in UCI and OpenML, are designed without hierarchical organization, making it difficult for researchers to explore and apply hierarchical approaches in diverse, real-world settings.

II. Background and Related Work

Traditional classification tasks can be categorized based on their output space. Binary classification deals with two mutually exclusive classes (e.g., true/false). Multi-class classification extends this to more than two mutually exclusive classes (e.g., cat, dog, or bird). A further generalization is multi-label classification, where a single instance can be associated with a set of labels simultaneously (e.g., a movie can be tagged as both action and comedy). Hierarchical classification and classifier chains can be considered forms of multi-label classification, as they predict a set of labels. Furthermore they also model the inherent or inferred dependencies and structures that exist between labels.

Within a hierarchical classification setting, an instance belonging to a specific class must also belong to all of its ancestor classes. This constraint allows models to use predictions for parent classes as highly informative features for predicting child classes. This topic has been a subject of research for many years [1] and has been applied across diverse domains, including text categorization, protein function prediction, and music genre classification [2].

Hierarchical classification algorithms are usually implemented as ensembles of models [1], [3], where each model specializes in a specific scope, and the multiple prediction values are combined using the hierarchical structure provided by the data. In [2] the author proposes a unifying framework to organize the existing approaches for hierarchical models, and in most of them require more than one classification model. The ensemble methodology [4] seeks to improve model performance by integrating multiple individual models into a collective framework.

Another technique relevant to the concept of hierarchical classification is the classifier chain [5], which is also an ensemble of models chained based on a structure defined by the user. Whenever there is label dependence [6] the classifier chain may be organized in a way that uses the dependencies to predict the labels, and if the labels are then organized in a DAG or tree structure, then it represents a hierarchical classification architecture.

The advantages of hierarchical classification in improving predictive performance over flat classification have been well-documented, leading to its adoption in various domains. In [7], it is shown that dividing the classification task into smaller, more focused subsets of documents relevant to a specific task enhances performance. In [8], a hierarchical classifier surpasses flat classification by exploiting the hierarchical structure of ImageNet. In [9], the authors demonstrate that the method used to construct the hierarchy further boosts performance when compared to a flat classifier. In [10], they highlight that modeling the hierarchy significantly affects the accuracy of gene function prediction. In [11] they show that hierarchical classifiers outperform flat approaches in predicting outcomes for highly imbalanced datasets. Lastly, [12] they introduce a python library that implements common patterns for hierarchical classification and also show that hierarchical approaches improve the performance in a consumer complaints classification task. While the existing literature on hierarchical classification is substantial and well-developed, it remains largely domain-specific, emphasizing methodological advancements and applications while neglecting efforts to expand into new domains or produce datasets for diverse application areas.

Despite the academic relevance of hierarchical classification and classifier chains, the prevailing industry standards provide limited support for such approaches. Historically, algorithms like Gradient Boosted Decision Trees (GBDTs), exemplified by frameworks such as*xgboost*, *lightgbm*, and *catboost*, have traditionally represented the state-of-the-art, consistently demonstrating strong performance in tabular data modeling. The *scikit-learn* framework is also frequently used in the tabular data setting. The field of tabular data science has been consistently using the same algorithms [13], [14]. Deep learning frameworks like Tensorflow and PyTorch are also frequently used, however GBDTs are usually the safest bet for tabular data [15]–[18].

III. Publicly Available Data

1) UCI Machine Learning Repository: For a systematic approach an API was utilized to automate metadata collection. This provided access to the dataset title, description, variables, and target column. We were able to collect metadata from 670 datasets, but by selecting only tabular datasets fit for a classification task then the total goes down to 267 datasets.

2) OpenML Datasets: Following a similar methodology, a programmatic approach using an open source library [19] was employed to retrieve metadata for 6274 datasets from the OpenML repository. Since binary classification problems are inherently non-hierarchical, a filter was then applied to this collection to select datasets fit for a classification task where the number of classes is greater than two, resulting in a final subset of 1068 datasets for analysis, and further to 934 after eliminating 134 duplicates with UCI

3) Other Repositories: Other niche repositories (HMC[1], Mulan[2], Meka[3]) focus on hierarchical or multi-label data but did not contain datasets meeting the study's requirements.

A. Classifying Dataset Metadata

In total considering both dataset repositories there are 1201 candidates to consider for a hierarchical classification setting. The methodology for the first filters of the datasets from the repositories was different because they provide different metadata. Now in order to automate the process of analysis and based on the description of each dataset a Large Language Model will be prompted to classify if the description of this dataset fits the setting of one that could be organized into an hierarchy.

[1] https://dtai.cs.kuleuven.be/software/clus/hmcdatasets/
[2] https://mulan.sourceforge.net/datasets-mlc.html
[3] https://waikato.github.io/meka/datasets/

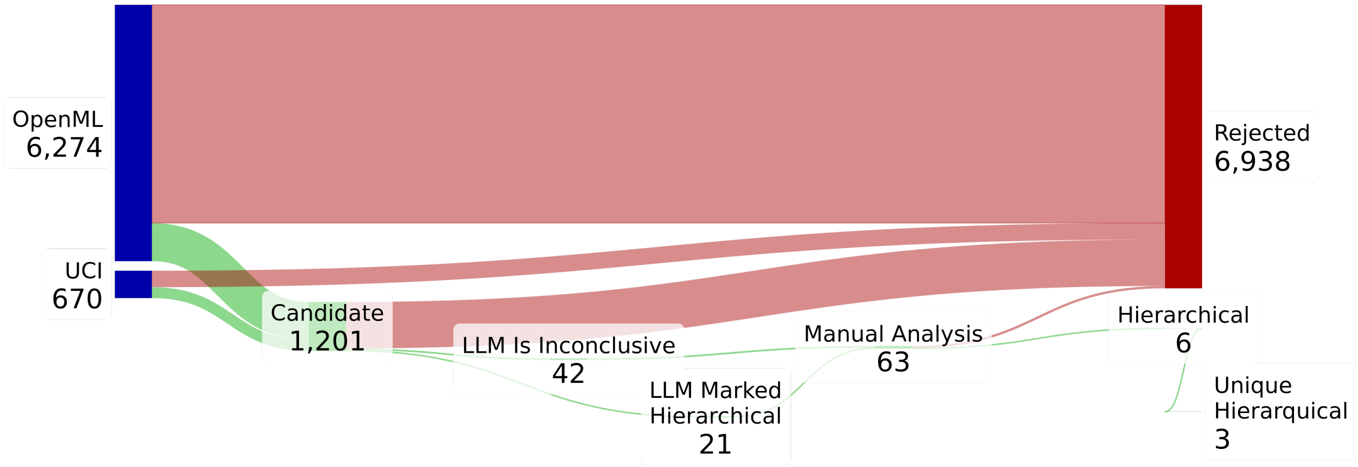

Fig. 1. Sankey diagram representing the classification process of the datasets as hierarchical or not. The diagram shows how datasets progress from initial sources (left) through filtering and analysis steps, with the majority being rejected and only a small number advancing to the final 'hierarchical' category.

The automation was done by a system and user prompts[4] to call the model's API. The system prompt informs that a dataset is hierarchical if its labels (classes) are organized in a structured way typically as a tree or a Directed Acyclic Graph (DAG) with parent-child or superclass-subclass relationships. Also the system prompt has examples of explicit hierarchy indicators, implicit hierarchy traits and domain examples commonly represented by hierarchical datasets. To improve the classification some examples of when to reject as hierarchical and when to classify as undefined are also added. The prompt was developed with input gathered from multiple large language model chat interactions and best practices for prompt creation.

Two free large language models (`gemini-2.5-flash` by Google, optimized for speed and efficiency; and `deepseek-chat-v3` by DeepSeek, trained on code and natural language for dialogue and instructions) were used to assess dataset descriptions. Each dataset was evaluated twice by both models (four assessments total). If any model flagged a dataset as hierarchical or undefined, it was then manually reviewed; otherwise, it was marked non-hierarchical.

B. Potential Hierarchical Tabular Datasets

Upon manual inspection, the majority of the candidate datasets were found to contain non-tabular data, including image collections, time series measurements, three-dimensional spatial data, and text corpora. While some datasets did meet the criterion of being tabular, only a very small fraction could plausibly be structured for a hierarchical classification task. Additionally, a substantial number of duplicated datasets were identified across and within repositories, often obscured by variations in naming conventions, minor preprocessing differences, or alternative upload instances that made them difficult to detect through comparisons of names or IDs. This duplication inflated the dataset counts at intermediate stages of the analysis, with the final set initially comprising six candidates that, after deduplication, was reduced to three unique datasets suitable for hierarchical classification.

The final selection of datasets that could be organized into a hierarchy:

- UCI ML **Mice Protein Expression** [20]: where the type of treatment that each mice received may be grouped (the control mice group is divided into stimulated to learn and not stimulated which is further split into injected with saline or injected with memantine);
- UCI ML **MicroMass** [21]: where classes may be grouped based on genus, genera and gram type and they even provide a taxonomy tree;
- Open ML **DDXPlus** [22]: where the conditions may be grouped into different types of pathologies (e.g. Bronchitis and Pneumonia are inflammatory, Influenza and HIV are viral infections).

Both **Mice Protein Expression** and **MicroMass** are relatively small datasets, with 931 and 1,080 instances respectively, making it unreasonable to train complex models on them. In contrast, the **DDXPlus** dataset contains a much larger number of instances (1,292,579), making it a more suitable option for training complex hierarchical models.

[4] https://gist.github.com/vmgustavo/4cd82cec547f02a4f3fd3a5f30d06719

IV. Conclusion

This study conducted a systematic examination of the availability of public tabular datasets appropriate for hierarchical classification, focusing on the UCI Machine Learning Repository and OpenML. By using large language models and manual verification, over 1,200 datasets were reviewed, yet only three were found to be suitable for hierarchical classification. This result confirms that the limited availability of such datasets is a constraint for research in this area. To our knowledge, this is the first study to quantitatively confirm the rarity of such datasets.

The scarcity of hierarchical tabular datasets in the most widely used open machine learning repositories constrains the development of comprehensive benchmarks and the reproducibility of experimental results. This lack of resources risks biasing research toward domains where hierarchical data is more abundant, thereby limiting advances in practical applications. Beyond their established use in text, image, and bioinformatics, the exploration of hierarchical classification methods remains restricted due to the absence of suitable data. This shortage also slows the advancement of supporting software libraries, making the widespread adoption of hierarchical classification in real-world applications a distant possibility, despite well-documented advantages in both predictive performance and model interpretability.

Addressing this gap will require data collection initiatives and community-driven efforts to publish and curate hierarchical datasets for tabular domains. In parallel, developing synthetic data generation techniques for hierarchical structures and expanding the research into other less structured data repositories could provide immediate support for benchmarking and algorithmic testing, mitigating some of the current limitations. We believe that systematically expanding the availability and diversity of hierarchical datasets will be critical to unlocking the full potential of hierarchical classification methods and enabling their broader adoption across application areas.

References

[1] A. D. Gordon, "A review of hierarchical classification," 1987. [Online]. Available: https://api.semanticscholar.org/CorpusID:115390896

[2] C. N. Silla and A. A. Freitas, "A survey of hierarchical classification across different application domains," *Data Mining and Knowledge Discovery*, vol. 22, pp. 31–72, 2010. [Online]. Available: https://api.semanticscholar.org/CorpusID:207113055

[3] C. Vens, J. Struyf, L. Schietgat, S. Deroski, and H. Blockeel, "Decision trees for hierarchical multi-label classification," *Machine Learning*, vol. 73, pp. 185–214, 2008. [Online]. Available: https://api.semanticscholar.org/CorpusID:1847933

[4] L. Rokach, "Ensemble-based classifiers," *Artificial Intelligence Review*, vol. 33, pp. 1–39, 2010. [Online]. Available: https://api.semanticscholar.org/CorpusID:11149239

[5] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains: A review and perspectives," *J. Artif. Intell. Res.*, vol. 70, pp. 683–718, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:209516429

[6] K. Dembczynski and W. Cheng, "On label dependence in multi-label classification," 2010. [Online]. Available: https://api.semanticscholar.org/CorpusID:7981244

[7] A. K. Tegegnie, A. N. Tarekegn, and T. A. Alemu, "A comparative study of flat and hierarchical classification for amharic news text using svm," *International Journal of Information Engineering and Electronic Business*, vol. 9, no. 3, p. 36, 2017.

[8] L. E. M. Guerrero, Y. F. Ceballos, and L. D. T. Rojas, "Leveraging imagenet's hierarchical structure for enhanced image classification and retrieval," *Journal of Image and Graphics*, vol. 13, no. 4, 2025.

[9] S. Gauch, A. Chandramouli, and S. Ranganathan, "Training a hierarchical classifier using inter document relationships," *Journal of the American Society for Information Science and Technology*, vol. 60, no. 1, pp. 47–58, 2009.

[10] N. Cesa-Bianchi and G. Valentini, "Hierarchical cost-sensitive algorithms for genome-wide gene function prediction," in *Machine learning in systems biology*. PMLR, 2009, pp. 14–29.

[11] D. Ghazi, D. Inkpen, and S. Szpakowicz, "Hierarchical versus flat classification of emotions in text," in *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text*, 2010, pp. 140–146.

[12] F. M. Miranda, N. Köhnecke, and B. Y. Renard, "Hiclass: a python library for local hierarchical classification compatible with scikit-learn," *Journal of Machine Learning Research*, vol. 24, no. 29, pp. 1–17, 2023.

[13] F. Psallidas, Y. Zhu, B. Karlas, M. Interlandi, A. Floratou, K. Karanasos, W. Wu, C. Zhang, S. Krishnan, C. Curino, and M. Weimer, "Data science through the looking glass and what we found there," *ArXiv*, vol. abs/1912.09536, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:209439707

[14] A. Tschalzev, S. Marton, S. Ludtke, C. Bartelt, and H. Stuckenschmidt, "A data-centric perspective on evaluating machine learning models for tabular data," *ArXiv*, vol. abs/2407.02112, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:270878049

[15] R. Shwartz-Ziv and A. Armon, "Tabular data: Deep learning is not all you need," *ArXiv*, vol. abs/2106.03253, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:260435623

[16] A. Shmuel, O. Glickman, and T. Lazebnik, "A comprehensive benchmark of machine and deep learning across diverse tabular datasets," *ArXiv*, vol. abs/2408.14817, 2024. [Online]. Available: https://api.semanticscholar.org/CorpusID:271962985

[17] D. C. McElfresh, S. Khandagale, J. Valverde, C. VishakPrasad, B. Feuer, C. Hegde, G. Ramakrishnan, M. Goldblum, and C. White, "When do neural nets outperform boosted trees on tabular data?" *ArXiv*, vol. abs/2305.02997, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:258479721

[18] Y. V. Gorishniy, I. Rubachev, V. Khrulkov, and A. Babenko, "Revisiting deep learning models for tabular data," in *Neural Information Processing Systems*, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:235593213

[19] J. van Rijn, A. Kadra, P. Gijsbers, N. Mallik, S. Ravi, A. Müller, J. Vanschoren, and F. Hutter, "openml-python: a python api for openml," https://github.com/openml/openml-python, 2014.

[20] G. K. Higuera, Clara and K. Cios, "Mice Protein Expression," UCI Machine Learning Repository, 2015, DOI: https://doi.org/10.24432/C50S3Z.

[21] P. Mah and J.-B. Veyrieras, "MicroMass," UCI Machine Learning Repository, 2014, DOI: https://doi.org/10.24432/C5T61S.

[22] A. Fansi Tchango, R. Goel, Z. Wen, J. Martel, and J. Ghosn, "Ddxplus: A new dataset for automatic medical diagnosis," *Advances in neural information processing systems*, vol. 35, pp. 31 306–31 318, 2022.