FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Online hierarchical partitioning of the output space in extreme multi-label data streams

Lara Sá Neves

Mestrado em Engenharia e Ciência de Dados

Supervisors: Goreti Marreiros, José Luís Moura Borges

July 21, 2025

Online hierarchical partitioning of the output space in extreme multi-label data streams

Lara Sá Neves

Mestrado em Engenharia e Ciência de Dados

July 21, 2025

Abstract

In many real-world applications, such as recommendation systems or healthcare monitoring, data arrives continuously and can be associated with multiple labels simultaneously. Learning from such multi-label data streams is particularly challenging due to the dynamic nature of the environment, high-dimensional label spaces, sparse label co-occurrence, and evolving data distributions. These challenges are compounded by concept drift, where not only input distributions but also label correlations and frequencies shift over time.

This thesis introduces and evaluates two complementary frameworks: *iHomer* and its enhanced variant *iHomer+*.

The *iHomer* framework addresses real-world data by incrementally clustering correlated labels using online divisive-agglomerative strategies based on *Jaccard* similarity, and by learning over these clusters with a global tree-based model guided by a multivariate *Bernoulli* process. It integrates drift detection at both global and local levels to dynamically adapt label partitions. This design was originally evaluated on real-world data, in line with standard practice in the field, where benchmark datasets offer rich and naturally occurring label dependencies. However, real data alone does not allow for controlled investigation of specific types of concept drift or systematic label space variations. To address this, *iHomer+* extends the method to controlled synthetic data streams by embedding three adaptive global learners and introducing a background replacement mechanism. This enables targeted evaluation of concept drift types (abrupt, incremental, and recurrent) by explicitly modeling label space changes.

Empirical results show that *iHomer* outperforms 17 state-of-the-art baselines on 23 real-world datasets, while iHomer+ further improves predictive performance by up to 29% on 9 synthetic benchmarks with annotated drift. Together, these contributions offer a robust and adaptive solution for online multi-label learning under concept drift.

Keywords: Multi-label classification, Incremental decision tree, Data stream, Concept drift

Resumo

Em várias aplicações do dia a dia, como sistemas de recomendação ou monitorização de informação médica, os dados são processados de forma contínua e podem estar associados a múltiplas classes ao mesmo tempo. A aprendizagem a partir desses dados apresenta desafios significativos, considerando a natureza instável do ambiente, a elevada dimensionalidade do espaço de resultados e ainda a evolução das distribuições de dados. Estes desafios são agravados pelas mudanças de conceito, onde não só as distribuições de entrada, mas também as correlações e frequências das distribuições de saída mudam ao longo do tempo.

Esta dissertação apresenta e analisa dois novos métodos: o *iHomer* e a respetiva versão melhorada, o *iHomer+*. O método *iHomer* recebe dados reais e agrupa em tempo real as variáveis de saída que estão correlacionadas, através de estratégias de agrupamento incremental e divisivo-aglomerativas, baseadas na similaridade de *Jaccard*. Um modelo baseado em árvores de decisão é treinado para cada grupo, guiado por processos multivariados de *Bernoulli*. Esta estratégia integra ainda mecanismos de deteção de mudanças tanto a nível global como local, permitindo uma adaptação dinâmica. Esta avaliação com dados reais segue as práticas do estado da arte, ao considerar cenários realistas com dependências complexas entre variáveis.

Contudo, dados reais não permitem um controlo sistemático sobre os diferentes tipos de mudança de conceito nem sobre alterações específicas no espaço de variáveis de saída. Para colmatar essa limitação, o *iHomer+* amplia a abordagem para lidar com fluxos de dados sintéticos nos quais as mudanças são conhecidas e bem definidas. Para isso, incorpora três classificadores globais adaptativos e introduz um mecanismo de substituição em segundo plano, permitindo a avaliação controlada de mudanças abruptas, incrementais e recorrentes ao longo de alterações no espaço de saída. Os resultados experimentais mostram que o *iHomer* supera 17 métodos de referência em 23 conjuntos de dados reais, enquanto o *iHomer+* eleva o desempenho preditivo em 29% em 9 conjuntos de dados sintéticos com mudanças explícitas.

Juntas, estas contribuições são uma solução robusta e adaptativa para a aprendizagem em tempo de real de dados multi-classe em cenários dinâmicos com mudanças de conceito.

Keywords: Classificação multi-classe, Árvore de decisão incremental, Fluxo de dados em tempo real, Mudança de conceito

Acknowledgements

This work was supported by the GECAD research group at ISEP, Porto, where I benefited from both research guidance and infrastructure. I would like to express my sincere gratitude to my academic supervisors, Professor Goreti Marreiros (ISEP, Porto) and Professor José Luís Moura Borges (FEUP, University of Porto), for their continuous support and supervision throughout this project. I also gratefully acknowledge the collaboration and mentorship of Professor Alberto Cano at Virginia Commonwealth University, United States of America, which significantly contributed to the development of this work. I would like to extend my thanks to my colleague Afonso Lourenço for his support and constructive input throughout this project. Finally, I would like to thank the FLAD Scholarship for its financial support, which helped make this research possible.

Lara Sá Neves

"There is little success
where there is little laughter."

Andrew Carnegie

Contents

List of Figures

List of Tables

Abbreviations

BR	Binary Relevance
CDD	Critical Difference Diagram
DSM	Data Stream Mining
HT	Hoeffding Tree
ML	Machine Learning
NP	Non-Pairwise
SMLL	Streaming Multi-Label Learning

Chapter 1

Introduction

In the modern era of data proliferation, knowledge discovery and data mining have become indispensable for extracting actionable insights from the ever-growing volume of information (Maimon and Rokach, 2009). As data is increasingly generated in real-time across domains such as business, healthcare, and scientific research, traditional static mining approaches struggle to meet the demands of continuous and timely analysis. **Data Stream Mining** (DSM) addresses these limitations by focusing on extracting knowledge from rapidly evolving data streams (Gaber et al., 2005), enabling applications where real-time decision-making is critical.

A fundamental component of DSM is **online learning**, where algorithms must continuously adapt to incoming data without access to the full dataset. Unlike batch learning, online models process each instance sequentially, often under strict constraints on memory and computation. This dissertation adopts a view of online learning as a specialized form of incremental learning, wherein each instance is processed only once, and the data is neither stored nor revisited (Minku and Yao, 2011).

In many real-world data streams, the underlying distribution is **nonstationary**, meaning it evolves over time. This leads to **concept drift**, a phenomenon in which the target concepts or data distributions alters, often unpredictably. If unaddressed, concept drift can render previously learned models inaccurate or obsolete, requiring adaptive mechanisms to maintain performance (Gomes et al., 2017b). Drift may occur abruptly, gradually, or recurrently, posing significant challenges to model stability and adaptability. In this context, it is important to distinguish between general change and concept drift. A *change* refers to any modification in the statistical properties of the data or the environment, which may or may not impact model performance. In contrast, *concept drift* denotes a specific type of change that alters the conditional distribution $p(y|x)$ or the marginals $p(x)$ or $p(y)$ in a way that negatively affects predictive accuracy. In other words, *all drifts are changes, but not all changes are drift*.

While traditional approaches to data stream mining often assume single-label classification, many real-world instances are naturally described by multiple, interdependent labels. This scenario, known as **Streaming Multi-Label Learning** (SMLL), aims to predict sets of labels for each instance while continuously adapting to evolving data (Read et al., 2012). SMLL introduces

unique challenges: (1) an exponentially large output space due to all possible label combinations, (2) severe class imbalance within and across labels, and (3) the presence of concept drift, which alters not only feature distributions but also label definitions and their interdependencies over time (Aguiar et al., 2024).

These challenges are magnified in streaming settings, where models must operate under memory and time constraints, process each instance only once, and remain robust to nonstationarity. As a result, the design of effective SMLL models demands strategies that can both capture label dependencies and adapt swiftly to changing data distributions.

Local vs. Global Models. In multi-label learning, a central design decision concerns the structure of the predictive model. Local models treat each label as an independent binary classification task, enabling modular learning but ignoring inter-label correlations (Kocev et al., 2007). In contrast, global models aim to learn a joint prediction function over the entire label set, effectively capturing label dependencies and often achieving greater efficiency and predictive power (Duarte and Gama, 2015; Osojnik et al., 2017). However, global approaches may struggle with scalability in high-dimensional label spaces.

1.1 Problem Setting

Traditional multi-label models assume stationarity and fixed label correlations, which limits their adaptability. While local models handle each label independently and offer scalability, they ignore inter-label dependencies. Global models, on the other hand, consider the label structure holistically but are often computationally intensive. Bridging the gap, hybrid models aim to balance expressiveness and efficiency by dynamically partitioning the output space. Literature has explored output specialization methods (Sousa and Gama, 2018), where subsets of labels are grouped into coherent clusters, and each group is modeled globally. This hybrid approach enables a more scalable yet dependency-aware framework by applying structured prediction over partitioned label spaces, making it especially suitable for evolving multi-label streams.

Research Question: How can we design an efficient and adaptive hybrid model for multi-label data streams that captures evolving label dependencies and adapts to various types of concept drift?

1.2 This Work

Motivated by these challenges, this dissertation proposes a novel framework that dynamically adapts label partitioning strategies in response to evolving data streams and concept drift, presenting iHOMER and its enhanced variant iHOMER+. These are two novel frameworks for adaptive output space partitioning in multi-label data streams. iHOMER incrementally builds label clusters using Jaccard-based dissimilarity and learns over them with tree-based models. iHOMER+ introduces a three-tier ensemble and drift-sensitive learner switching to better cope with various

drift scenarios. These methods are evaluated on real and synthetic benchmarks covering abrupt, gradual, and recurrent drifts.

1.3 Organization

The remainder of this document is organized as follows: Chapter 2 reviews existing work on multi-label streaming learning, Chapters 3 and 5 detail the iHOMER and iHOMER+ frameworks, Chapters 4 and 6 present empirical results, correspondingly, and Chapter 7 concludes with key findings and future research directions.

1.4 Scientific contributions

As a direct outcome of this research, a paper presenting the iHOMER framework was submitted to the 28th European Conference on Artificial Intelligence (ECAI 2025). Following its acceptance, a journal version that includes the enhanced iHOMER+ extension is being prepared and will be submitted for publication. These contributions represent important advances in the field of online multi-label learning, particularly under the challenges of concept drift.

Furthermore, within the scope of this thesis work, the author also submitted a successful application to the highly competitive CMU Portugal Dual Degree PhD Program, jointly organized by the Foundation for Science and Technology (FCT, Portugal) and Carnegie Mellon University (USA). The PhD will officially begin in September 2025 and will allow the author to continue and expand the research directions initiated in this thesis, with a focus on trustworthy and adaptive AI systems.

Chapter 2

Related work

Multi-label learning algorithms model label correlations in various ways, in a spectrum of global and local approaches, and can be categorized along four dimensions, regarding how label correlations are being exploited: (1) instance location (global vs. local), (2) dependency (conditional vs. unconditional), (3) dataset space (feature, instance, or label), and (4) correlation order (first-, second-, or high-order) (Zhang and Zhou, 2007).

Firstly, regarding instance location, label correlations can be classified as global when calculated over all instances, or local assuming correlations can only be shared when considering a single instance subset (Huang and Zhou, 2012).

Secondly, regarding dependency, label correlations can be represented based on probabilities with conditional dependence, which captures the dependencies between labels given a specific instance and predicts the likelihood of labels occurring together, or unconditional dependence, which models the probability of certain labels occurring together across the entire dataset (Dembczyński et al., 2012).

Thirdly, regarding dataset space, label correlations can be calculated based on the feature (Zhang and Wu, 2014; Weng et al., 2018), instance (Li and Yang, 2021), and label space (Ye et al., 2015; Shi et al., 2014). Intuitively, global approaches are often modeled using all labels from the label space only, without considering the instance space, optionally capturing correlations from each space separately and combining them (Xu et al., 2020; Li and Yang, 2021). On the other hand, ensembles of global approaches rely on clustering instance space, label space, or feature space, optionally clustering each space separately and combining them later. Similarities measures, graphs, optimization, probabilities, and other techniques are employed to capture these different types of correlations (Ahmadi and Kramer, 2019).

Fourthly, regarding order, methods can ignore label correlation (first-order, e.g. Binary Relevance (BR)), model correlation for label pairs (second-order), and model correlations using all labels, or subsets of them (high-order), as illustrated in Figure 2.1 (Zhang et al., 2023). First-order strategies tackle multi-label learning in a label-by-label style (Zhang and Zhou, 2007). For example, relying on local approaches that decompose the multi-label learning problem into a number of independent binary classification problems, thus ignoring the coexistence of other labels. While

conceptually simple, the resulting approach might be suboptimal due to the ignorance of label correlations. Alternatively, second-order strategies can consider pairwise relations both via the ranking between a relevant label and irrelevant labels (Fürnkranz et al., 2008), or the interaction between any pair of labels (Ghamrawi and McCallum, 2005). As label correlations are exploited to some extent the resulting approaches can achieve good generalization performance. However, in most real-world applications label correlations go beyond the second-order assumption. In this case, high-order strategies can impose all other labels' influences on each label (Godbole and Sarawagi, 2004), or at least address connections among random subsets of labels (Read et al., 2011, 2008; Tsoumakas et al., 2010). While this high-order strategy has stronger correlation-modeling capabilities than the first-order and second-order strategies, it is computationally more demanding, so one can only consider among disjoint subsets of labels with high positive correlations.

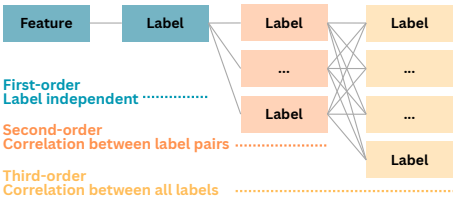


Figure 2.1: Representation of first-order, second-order, and high-order multi-label learning strategies.

Indeed, in practice, there are many similarities among the aforementioned categorizations. Global correlations, unconditional dependencies, and high-order correlations are comparable since they work with all labels. Local correlation, conditional dependency, and second-order correlations are comparable, since they work with subsets of labels.

More interestingly, these methods can be grouped into three algorithmic families: local models, global models, and hybrid partitions.

2.1 Local approaches

Problem transformation methods fit the data for traditional single-label streaming classifiers. In particular, BR is commonly used for its scalability and parallelism (Boutell et al., 2004), though it suffers from class imbalance and empty predictions. Streaming BR solutions include label-wise weighting (Ráez et al., 2004), separate windows for positives and negatives (Spyromitros-Xioufis et al., 2011), and concept drift detection via label entropy (Tomás et al., 2014). MINAS-BR handles novelty and unlabeled labels in extreme streaming settings (Júnior et al., 2023). Beyond BR, second-order methods like Label Ranking use one-vs-one classifiers to model pairwise label orderings, though scalability is an issue in high dimensions (Madjarov et al., 2012). High-order methods like Label Combination treat label sets as multi-class targets (Shi et al., 2014), while managing the growing number of label sets with probabilistic modeling (Wei et al., 2021) or early-sample buffering (Read et al., 2012).

2.2 Global approaches

Global models jointly learn over the entire label set and can exploit interdependencies. Algorithm adaptation methods directly handle multi-label outputs, reducing complexity and making them ideal for streaming. Notable examples include extensions of ML-kNN (Zhang and Zhou, 2007), using Bayesian inference over neighbors, and extensions of ML-DT (Clare and King, 2001), using multi-label entropy. Streaming ML-kNN solutions include windowing to efficiently update posteriors (Spyromitros-Xioufis et al., 2011) and dual-memory models like ML-SAM-kNN (Roseberry and Cano, 2018), which splits recent and long-term data. Extensions such as MLSAMPkNN (Roseberry et al., 2019) and MLSAkNN (Roseberry et al., 2021) introduce punitive mechanisms and adaptive clustering. ARkNN (Roseberry et al., 2023) proposes resource-efficient instance aging. Streaming ML-DT solutions extend Hoeffding Trees (HT) (Domingos and Hulten, 2000) with label-set predictors and pruning techniques. For example, iSOUPT (Osojnik et al., 2017) applies multi-target regression via perceptron-based model trees, which can be extended with drift detection (Stevanoski et al., 2024) and per-target model selection (Mastelini et al., 2019). MLHAT (Esteban et al., 2024) goes beyond, incorporating multi-label drift detection in the growing process with label imbalance adaptation via cardinality-aware metrics.

2.3 Hybrid partitions

It is important to note that the aforementioned global approaches are sometimes referred to as ensemble methods due to their use of multiple binary models. Similarly, hybrid partitioning methods are occasionally labeled as ensemble methods because they involve several global multi-label models. However, since these models handle disjoint label subsets and collectively represent a unified prediction system, they effectively constitute a single multi-label model.

Hybrid partition methods blend local and global models by clustering labels and applying global predictors to each subset (Sousa and Gama, 2018), offering a middle ground between fine-grained and holistic learning. By modeling label correlations locally, i.e., within label clusters (Huang et al., 2015), or across all labels (Gatto et al., 2025), these methods aim to efficiently guide label space partitions, rather than relying on random partitioning (Szymański et al., 2016; Breskvar et al., 2018) or genetic algorithms that overlook label dependencies (Basgalupp et al., 2021; Moyano et al., 2020).

Prominent examples such as HOMER (Tsoumakas et al., 2008) and HPML (Gatto et al., 2025) focus on scalability and label correlation modeling but either predefine the label clusters or lack mechanisms for dynamically adapting them in response to concept drift. Despite their strengths, these strategies are typically designed for batch learning: they assume access to the full dataset, rely on static partitions, and are not optimized for the memory and time constraints of streaming settings.

More recent hybrid approaches have been developed to support online learning, notably those based on rule learning systems. These leverage the modular nature of rule-based models to build

localized predictors over partitions of the input space. For example, AMRules (Almeida et al., 2013) has been extended to multi-label classification (Duarte et al., 2016), incorporating mechanisms for handling concept drift using the Page-Hinkley test and detecting anomalous instances. This modularity enables surpassing both global and local correlation-based methods via rule specialization for non-disjoint outputs in multi-target regression (Duarte and Gama, 2015) and multi-label classification (Sousa and Gama, 2018). When performing rule expansion, if the variance in the target space is reduced only for certain targets, a new rule is created for those targets, while a complementary rule is retained for the rest. As a result, the system can generate rules covering all, some, or just a single target.

However, despite their flexibility and streaming capabilities, rule-based hybrid models still face some limitations. As the number of labels increases, rule expansion may become computationally expensive and unstable, and while they excel at modeling local patterns, they may underexplore global label dependencies unless explicitly integrated, potentially limiting performance in tasks with strong inter-label structure.

2.4 Summary

Despite substantial progress, no prior work fully integrates dynamic output space partitioning, hierarchical adaptation, and multi-level drift detection. This gap motivates the iHOMER and iHOMER+ approaches proposed in this thesis, which combine online clustering, adaptive tree-based learners, and ensemble strategies for robust multi-label stream learning.

Chapter 3

iHOMER

3.1 Problem definition

Online multi-label learning poses significant challenges when dealing with continuously evolving data streams. The fundamental issue is adapting efficiently to nonstationary environments, characterized by high-dimensional and sparse label spaces where label dependencies frequently shift over time. Traditional multi-label learning methods often fail in these scenarios, either because they treat each label independently, ignoring valuable inter-label correlations, or because global methods become computationally prohibitive as the label space grows. Furthermore, real-world multi-label data streams often experience concept drift, where not only the input distributions but also the correlations and frequencies among labels evolve dynamically, making static approaches ineffective.

Consequently, the central research question addressed in this chapter is how to design a scalable, adaptive, and efficient hybrid model for multi-label data streams capable of continuously capturing evolving label dependencies and swiftly adapting to various types of concept drift.

HOMER. Hierarchy Of Multi-label classifiERs (Tsoumakas et al., 2008) is an established batch multi-label classification algorithm designed to handle large label spaces efficiently. HOMER employs a balanced clustering method, usually balanced k-means, to partition labels into hierarchical clusters based on label similarity. Each cluster, represented by a meta-label (logical disjunction of labels within the cluster), simplifies classification by reducing the original large label space into smaller, manageable subsets. Classifiers are trained independently within each cluster, and predictions are generated by traversing the hierarchy from root to leaves.

However, HOMER is inherently limited in streaming contexts because it lacks incremental learning capabilities and mechanisms for adapting to concept drift, restricting its applicability to dynamic data streams.

ODAC. Online Divisive-Agglomerative Clustering (ODAC) (Rodrigues et al., 2008) continuously maintains a hierarchical clustering structure for evolving time-series data streams. ODAC computes pairwise distances using the Pearson correlation coefficient and dynamically manages clusters through splits and merges based on statistical tests supported by the Hoeffding bound. A

cluster splits when the difference between the largest and second-largest cluster diameter exceeds a defined statistical threshold, and merges occur when child clusters' diameters exceed their parent's diameter. This incremental, adaptive approach is suitable for dynamic environments but is limited by its reliance on Pearson correlation, which is not ideally suited for binary label data.

3.2 Overview

This work introduces iHOMER (Incremental Hierarchy Of Multi-label classifiERs), designed to address limitations in existing multi-label classification strategies for nonstationary environments. iHOMER integrates HOMER's hierarchical multi-label classification framework with an adaptation of ODAC's incremental and adaptive clustering strategies, introducing critical enhancements specifically tailored for streaming multi-label data. iHOMER is notably the first incremental tree-based output specialization method explicitly designed for nonstationary environments, introducing several key innovations that differentiate it from prior approaches, particularly through the creation of disjoint, non-redundant clusters without predefined hierarchical structures.

Unlike ODAC's use of Pearson correlation, iHOMER employs the Jaccard similarity measure, better suited for binary label co-occurrence, thus enhancing both theoretical relevance and practical effectiveness. To manage incremental partitions dynamically and continuously adapt to concept drift, iHOMER utilizes statistical tests guided by the Hoeffding bound. This allows it to build non-redundant, disjoint clusters without relying on predefined hierarchical structures.

Clusters in iHOMER form via a top-down hierarchical approach, where splits occur based either on a user-defined minimum number of observations (n_{min}) or statistically significant differences assessed by the Hoeffding bound. As data streams evolve and previously created partitions become outdated, iHOMER re-merges child clusters back into parent clusters, again leveraging Hoeffding-bound significance testing. When necessary, a secondary aggregation stage ensures meaningful and balanced partitions, guided by a majority labelset classifier that monitors and maintains minimum cluster sizes.

To specifically address concept drift, iHOMER incorporates a drift detection mechanism that continuously evaluates multi-label predictive performance. Upon detecting potential drift, iHOMER initiates an alternative clustering process in parallel. Once drift is confirmed, the active clustering structure is seamlessly replaced, enabling adaptation without storing or revisiting historical data. Figure 3.1 illustrates a scenario in which, following drift detection, the alternate model replaces the main one. We observe that the new structure is significantly simpler, potentially indicating that the previously learned complex hierarchy is no longer relevant.

Within each hierarchical cluster, iHOMER employs Multi-Label Hoeffding Adaptive Trees (MLHAT) (Esteban et al., 2024) as global base classifiers, as further described in Section 3.3.5. Each MLHAT operates on disjoint subsets of labels, collectively forming a unified multi-label classifier without overlapping predictions. Figure 3.2 illustrates the hierarchical evolution, depicting MLHAT models (blue circles) dynamically emerging, adapting, or dissolving in response to concept drift. A detailed view of a single MLHAT model reveals an internal structure consisting of

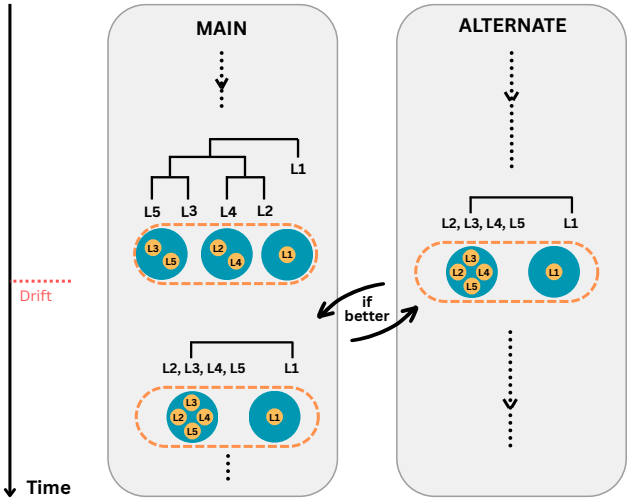


Figure 3.1: Example of iHOMER restructuring after drift detection: the alternate model replaces the main model with a simpler hierarchy.

cooperative single-label components, strategically exploiting intra-label correlations for enhanced predictive accuracy.

3.3 Detailed methodology

3.3.1 Incremental dissimilarity measure

To effectively capture pairwise similarity degrees of all labels in the label space, as exemplified in Table 3.1, the *Jaccard* index was used (Gatto et al., 2025). Formally, considering p_j and q_j as two distinct labels within the label space, the label dissimilarity is defined as:

$$d(\text{Jaccard}) = (1 - \frac{a}{a+b+c}) \tag{3.1}$$

where a represents the number of instances simultaneously classified with both p_j and q_j; b denotes the number of instances classified only with p_j; and c is the number of instances classified

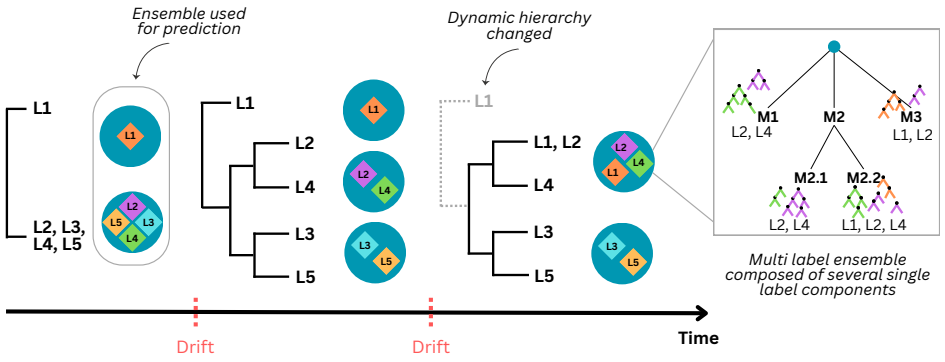


Figure 3.2: Hierarchy of label clustered classifiers evolves over time.

Table 3.1: Label space occurrences

Instance	L1	L2	L3	L4	L5
x1	1	0	1	1	1
x2	1	1	1	0	0
x3	0	1	0	1	0
x4	1	0	1	0	1
x5	0	0	1	1	0

only with q_j. It is important to note that the factors used to compute the *Jaccard* index can be updated incrementally. As each example is processed once, sufficient statistics are updated incrementally, while the dissimilarity graph, shown in Figure 3.3, is only computed when testing for splitting or aggregation. Missing labels are imputed as zero.

3.3.2 Growing output space hierarchy

The iHOMER algorithm employs online divisive agglomerative clustering to incrementally construct a tree-structured hierarchy that reflects the nested organization of label clusters. This process follows a top-down strategy, as illustrated in Figure 3.4. The leaves of the tree correspond to the resulting clusters, each associated with a distinct subset of variables (Rodrigues et al., 2008). The union of all leaves comprises the complete variable set, and the intersection between any two leaves is empty. During the growing process, based on the incremental dissimilarity measure and the corresponding diameters of the clusters, a key decision is made on whether a given leaf node, i.e., cluster, should be split. This cluster splitting is governed either by a user-defined threshold on the number of observations, denoted $n_{\min}$, or by a statistically grounded *Hoeffding bound* criterion, which provides a concentration inequality for the mean of a bounded random variable. Specifically, for a real-valued random variable r bounded in range R, after observing n independent samples, the bound ensures that, with confidence $1-\delta_{cluster}$, the true mean lies within ε of the empirical mean, where:

$$\varepsilon = \sqrt{\frac{R^2 \ln(1/\delta_{cluster})}{2n}}. \tag{3.2}$$

Each cluster C_k maintains its own Hoeffding threshold ε_k to determine whether the diameter-based heuristic condition is met. In particular, the splitting criterion compares the spread of pairwise dissimilarities within the cluster to the deviation from the mean:

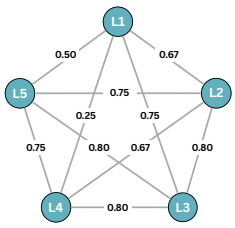


Figure 3.3: Dissimilarity graph based on label co-occurrences.

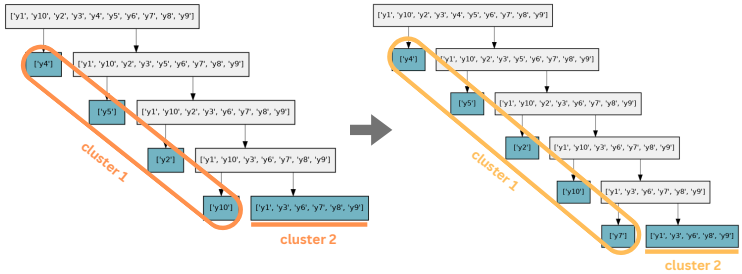


Figure 3.4: Dynamic hierarchical label clusters in Hypersphere dataset.

$$\frac{(d_1 - d_0)}{|d_1 + d_0 - 2d|} > \varepsilon_k. \tag{3.3}$$

where d_1 is the maximum distance between any two elements in the cluster, d_0 is the minimum pairwise distance, and d is the average dissimilarity across all pairs. If this inequality holds, the cluster is considered sufficiently diverse to justify splitting. In this case, the observations corresponding to the pair with maximum dissimilarity are selected to initiate new clusters as pivot variables, and the remaining variables are assigned to the nearest new cluster based on pivot proximity. This results in two subclusters centered around the most dissimilar elements in the original cluster. While this process enables the iHOMER tree to grow in a data-driven and statistically sound manner, maintaining robustness against noise and avoiding over-segmentation, near-equidistant variables can delay the decision between two equally valuable splits. To address this, a parameter $\tau_{cluster}$ is introduced. If $\tau_{cluster} > \varepsilon_k$, the system forces a test, assuming enough examples have been seen.

3.3.3 Aggregating output space hierarchy

A critical feature of iHOMER lies in its continuous monitoring of cluster diameters, which provides a basis not only for initiating splits but also for detecting when prior splits may no longer reflect the current structure of the data. In stationary data streams, each successive split within a divisive hierarchical clustering scheme typically leads to a reduction in intra-cluster dissimilarity, producing increasingly refined subgroups. However, in non-stationary environments, the data distribution may evolve, rendering earlier partitioning decisions obsolete. In such cases, the system must possess the flexibility to reverse previous splits and re-aggregate child nodes into their parent. To detect when a previous split has become inadequate, the algorithm assesses whether the diameters of sibling clusters remain meaningfully smaller than that of their parent:

$$2 \cdot \text{diam}(C_j) - (\text{diam}(C_k) + \text{diam}(C_s)) < \varepsilon_j. \tag{3.4}$$

where C_k is a given leaf node, C_s its sibling, C_j their common parent, and ε_j the Hoeffding-based confidence interval derived from the amount of data previously accumulated at the parent node. If this condition holds, then there is insufficient evidence to justify maintaining the split, and the system chooses to re-aggregate C_k and C_s into their parent C_j. The resulting re-aggregated

node begins with fresh statistical summaries, under the assumption that only future observations will inform its internal structure. This initialization allows the system to remain responsive to evolving data patterns. Moreover, the dual use of the *Hoeffding bound*, for both splitting and re-aggregation, enhances iHOMER's ability to adapt to changing data distributions while maintaining a balance between sensitivity to novel patterns and stability against noise. However, the resulting clusters might be unbalanced. Ideally, iHOMER should extract the most meaningful and balanced clusters from the hierarchy at any given time, as illustrated in Figure 3.4. To maintain flexibility of the hierarchy building process, a second-stage aggregation is performed only when necessary, ensuring clusters represent coherent groups rather than isolated individual labels. Leveraging on the fact that the produced dendrogram can be cut at different levels, this process adjusts the tree structure by merging clusters when appropriate, refining the representation of the data over time. To achieve this, the Majority Labelset classifier is used, when requested, to analyze the distribution of labelset sizes. It retrieves a frequency table indicating how often labelsets with a given number of labels occur, using this information to determine the minimum safe cluster size. This approach is intuitive as it suggests that clusters should contain at least as many labels as the most common labelset size. For example, if the most frequent labelset size in a dataset is eight, predicting only two labels at a time would be inefficient. Conversely, if most instances have just one label, maintaining specialized models for single-label prediction is more effective. Since this strategy follows a hierarchical clustering approach, imposing a maximum number of labels per cluster is unnecessary and may hinder the natural refinement of the structure over time. Theoretically, all labels could initially form a single cluster, but as new observations are processed, the hierarchy self-adjusts, splitting or merging clusters as needed. This adaptive process ensures that the clustering structure evolves to balance granularity and coherence, making rigid constraints on cluster size unnecessary.

3.3.4 Alternate output space partitioning

As the data distribution evolves, rendering earlier partitioning decisions obsolete, iHOMER dynamically aggregates or expands its label partitioning, allowing faster recovery rates. However, the base learner performance still requires time to improve with the continuous arrival of new samples, as the historical data before the concept drift plays a restraining role in the model's update, leading to the inability of classification accuracy to recover in a short time. In other words, if the learning model has been in a state of fail-safe operation, i.e., carrying the original distributed samples or information, for a long time, it might not be able to adapt to the demand of high timeliness in streaming data. Furthermore, in addition to true concept drifts, iHOMER is sensitive to false concept drifts, such as deviation of noise samples from the normal distribution, and unstable representations due to updated small-scale datasets. Consequently, to smooth these transitions, iHOMER features a conservative replacement strategy, by selectively retiring outdated models (Bifet and Gavalda, 2009). When iHOMER detects multiple consecutive drift signals, suggesting the active learner may be misaligned with the data stream, an alternative learner is initialized. This alternate component explores a new representation of the data stream and can replace the active

model only if it demonstrates superior performance in terms of current subset accuracy. Specifically, the *Welch's T-test* is applied to assess whether the prediction errors $\alpha, \alpha_{c'}$ of the active label clustering strategy c and a candidate alternative c' differ significantly, under the assumption of unequal variances. The test statistic t, of confidence $1 - \delta_{alt-cluster}$, is computed as:

$$t = \frac{\alpha_c - \alpha_{c'}}{\sqrt{\frac{\sigma_c^2}{N_c} + \frac{\sigma_{c'}^2}{N_{c'}}}} \tag{3.5}$$

where ε denotes the mean error, σ^2 the error variance, and N the number of instances observed in an *ADWIN*-controlled sliding window for each model (Bifet and Gavalda, 2009). When the test indicates a statistically significant difference, the alternative partitioning may replace its counterpart, supporting dynamic adaptation to concept drift in the data stream. Moreover, recursive replacements are possible, with multiple competing alternatives, specializing in different temporal regions of the data stream. The more recent alternate learner typically adapts quickly to current trends, making it particularly effective when dealing with abrupt concept drifts. In contrast, the older and generally more stable active learner adapts more slowly but tends to perform better in stationary or gradually evolving segments of the stream. This mechanism enables dynamic retirement of underperforming active models while preserving stability. Moreover, the introduction of a *grace period* $n_{\min}$ before replacement decisions helps reduce false alarms and stabilizes system responsiveness in the face of noise and short-term fluctuations.

3.3.5 Global base learner

Once iHOMER has identified meaningful label clusters through hierarchical partitioning, a specialized multi-label Hoeffding adaptive tree (MLHAT) (Esteban et al., 2024) is trained on each cluster.

Incremental decision tree algorithms work by recursively partitioning input space to create increasingly homogeneous subsets relative to target labels, allowing model updates as new instances arrive without requiring complete retraining or storage of historical data (Domingos and Hulten, 2000; Lourenço et al., 2025). MLHAT extends this concept to multi-label streaming data, considering label relationships and co-occurrences during tree construction, being the first native multi-label incremental tree that incorporates embedded concept drift adaptation to quickly identify and replace underperforming tree branches (Esteban et al., 2024). The full pseudocode for the global base learner, i.e. MLHAT, is presented in Algorithm 1.

Following these principles of incremental Hoeffding trees (Domingos and Hulten, 2000), and diving deep into how it works, MLHAT leverages on each incoming instance, updating sufficient statistics from the root to a leaf. For every attribute-value-class combination n_{ijk}, the algorithm tracks how often an attribute $X_i = x_{ij}$ appears with class y_k, enabling evaluation of potential splits. The base learner then chooses the best attribute using a heuristic function $G(\cdot)$, i.e., the information gain based on the *Bernoulli* distribution, where label priors are obtained from the counters maintained by the node (Esteban et al., 2024). It compares the top two attributes and applies the

Algorithm 1 MLHAT: Multi-label Tree Growth With Alternate Trees

procedure TREEGROW($T(n)$: sub-tree growing from a node n, (X,y): instance, δ_{spl}: confidence split, δ_{alt}: confidence alternate, n_{spl}: grace period, n_{alt}: alternate grace period, α: multi-label ADWIN)
  Route to leaf l, obtain prediction $\hat{y}$
  for each node n in path **do**
    $n_n \leftarrow n_n + 1$
    if $\exists T_{\text{alt}}(n)$ **then**
      TREEGROW($T_{\text{alt}}(n)$, (X,y), δ_{spl}, δ_{alt}, n_{spl}, n_{alt})
    end if
    if n is a leaf **then**
      if (impure class dist.) and $(n_n - n_{\text{check}_n} > n_{\text{split}})$ **then**
        rank $\leftarrow$ Sorted $G(\cdot)$
        $G_{\text{best}} = \max(rank)$
        $G_{\text{second}} = \max(rank \setminus \{G_{\text{best}}\})$
        if $G_{\text{best}} - G_{\text{second}} \geq \varepsilon$ **then**
          Split n, set post-split distributions at all b
          $n_b, n_{\text{check}_b} \leftarrow$ sum class distributions for all b
        end if
      end if
    end if
    $\alpha \leftarrow$ ADWIN update based on y and $\hat{y}$
    if α detects warning and $\nexists T_{alt}(n)$ **then**
      $T_{alt}(n) \leftarrow$ new sub-tree in background
    end if
    if $\exists T_{alt}(n)$ and $W(T_{alt}(n)) > n_{\text{alt}}$ **then**
      $\varepsilon_{\text{alt}} \leftarrow e, e_{\text{alt}} \leftarrow \alpha, \alpha_{alt}$
      if $(e - e_{\text{alt}}) > \varepsilon_{\text{alt}}$ **then**
        $T(n) \leftarrow T_{\text{alt}}(n)$
        $T_{\text{alt}}(n) \leftarrow \emptyset$
      else if $(e_{\text{alt}} - e) > \varepsilon_{\text{alt}}$ **then**
        $n \leftarrow$ prune $T_{\text{alt}}(n)$
      end if
    end if
  end for
end procedure

Hoeffding Bound to decide whether the difference ΔG is statistically significant, with confidence $1 - \delta_{tree}$:

$$\varepsilon_m = \sqrt{\frac{\log_2(|L|)^2 \ln(1/\delta_{tree})}{2n}} \tag{3.6}$$

where L is the number of known label sets, δ is the desired confidence level, and n is the number of samples at the node m. A split occurs when $\Delta G > \varepsilon_h$,. In this way, the tree would expand incrementally as long as the entropy of the leaf nodes can be minimized with new splits. To manage complexity and prevent overfitting in incremental decision trees, two key hyperparameters are

used: the *grace period* $n_{\min}$, which defines how often splits are evaluated, and the *tie threshold* τ_{tree}, which serves as a pre-pruning mechanism. The tie threshold τ_{tree} limits growth when the difference $\Delta\bar{H}(X_i)$ between the best two attribute heuristics is small, i.e., $\Delta\bar{H}(X_i) \leq \tau_{tree}$. This helps avoid unstable decisions under noisy or ambiguous data. While a well-chosen τ mitigates unnecessary tree growth and improves scalability, its effectiveness is highly data-dependent. If too low, the tree may underfit; if too high, it risks overfitting and computational inefficiency. Together, τ_{tree} and $n_{\min}$ balance accuracy and complexity in data stream learning, allowing the production of increasingly larger trees that can divide the problem space into finer regions, better approximating the target concepts. However, such tree growth still remains highly uncontrollable, which can lead to two main issues: excessive memory consumption due to multiple redundant splits on all features, and a loss of plasticity due to the descendant nodes getting subsequently fixed to the space covered by their parent node. To address this, outdated subtrees are replaced with alternates that are grown when a detector detects drift through a changing error rather than on changing split attributes (Bifet and Gavalda, 2009). However, such split revision procedure can also reduce variance by simplifying the model, and introduce significant bias if important information is lost during subtree pruning (Manapragada et al., 2018). Thus, only when the alternate subtree outperforms the original over a sufficient number of instances does it replace the latter. Following this intuition, MLHAT monitors the error in the Hamming loss, triggering replacements based on the *Hoeffding bound* ε_{alt} for alternating trees (Bifet and Gavalda, 2009) and defined from monitored errors and instances seen by both main W and alternate W' sub-trees as:

$$\varepsilon_{alt} = \sqrt{\frac{2e(1-e')(W+W')\ln(2/\delta_{alt-tree})}{W \cdot W'}} \tag{3.7}$$

where three scenarios can occur: (1) if $(e-e') > \delta_{alt-tree}$, the alternate tree replaces the main one, pruning the previous sub-structure resetting the drift detector in the alternate tree for future concept drifts, (2) if $(e'-e) > \delta_{alt-tree}$, the alternate tree remains without changes and the alternate tree is pruned, the concept drift was reversed, and (3) if $\delta_{alt-tree} \geq |e-e'|$, the two sub-trees continue to receive instances and expand to eventually differentiate their performance. This process aims to reduce bias and keep the model relevant by ensuring new subtrees adapt to the updated concept before deployment, with recursive replacements among alternates of alternates being possible. In other words, having alternative subtrees switch between training and test modes, and during the latter, also pruning the alternative subtrees whose accuracy does not increase over time. Moreover, leveraging on the fact that, in the midst of a concept drift, there will be multiple alternate sub-trees that are candidates to replace the branches whose performance is declining, their predictions is combined with the main tree, weighted based on the error monitored in each of the leaves involved, and conditioned on the alternate trees having received at least $n_{\min}$ instances. Thus, providing a good balance between the main structure that has the statistical guarantees of the *Hoeffding bound*, and a fast response to possible concept drifts.

Local leaf selection. To select among different multi-label classifiers in the leaves, MLHAT monitors at each node two markers: its current multi-label entropy H_0, as previously discussed, and

its the cardinality W, i.e. the number of instances that have reached it at a given time. When there is a high number of instances but a low information gain ΔG between the entropies currently and after the eventual split, i.e. $H_0 > 0$ & $W > \eta$, a more demanding and computationally expensive multi-label classifier is required to find deeper relationships and separate the label-sets given their feature space. For this purpose, a BR transformation ensemble of ADWIN Bagging with Logistic Regression (Bifet et al., 2009) is used. Conversely, when the leaf is going through an intermediate state with entropy starting to increase but cardinality is still under a given threshold η, i.e. $H_0 > 0$ & $W \leq \eta$, the least data demanding classifier is used. For this purpose, a LP transformation of the naturally incremental kNN is adopted to its balance between low complexity and good ability at detecting relationships between labels in data-poor scenarios. Naturally, when $H_0 = 0$, entropy cannot be minimized anymore, i.e., the path to the leaf perfectly separates instances of the same label set. Thus, predicting the majority label set is accurate and computationally efficient.

Instance space selection. Moreover, following the success of bagging ensembles in data stream mining (Alberghini et al., 2022), the entire training process is conditioned by an online bootstrapping following a Poisson(λ) distribution, in order to apply an extra weight in some instances to perform resampling with replacement from the stream. This extra weight w affects the node statistics, the count of instances seen W, as well as the multi-label classifiers used in leaves in general.

The full pseudocode for iHOMER is presented in Algorithm 2.

3.4 Advantages

First, iHOMER offers high flexibility, allowing hybrid partition structures to emerge naturally during learning without requiring manual specification of parameters such as the number of clusters, classifiers, or labels per model. This stands in contrast to genetic algorithms, which often require extensive tuning and rigid configurations. *Second*, the framework enforces a strict non-repetition constraint, ensuring that labels are not duplicated across clusters, a limitation present in earlier methods that relied on random or hierarchical partitioning. *Third*, iHOMER is highly efficient in terms of computation and memory usage, leveraging lightweight clustering and learning components guided by the Hoeffding bound, specifically, an incremental dissimilarity measure for clustering and a multivariate Bernoulli process for tree construction. *Lastly*, iHOMER incorporates both global and local drift detection mechanisms. This adaptability allows it to restructure label partitions and update its learners dynamically, validated through alternate background models and robust drift confirmation techniques.

Through these mechanisms, iHOMER provides an effective and principled solution for scalable, adaptive multi-label learning in streaming environments.

Algorithm 2 iHOMER

Given: Stream (x, y), cluster hierarchies $H \in \{H_{main}, H_{alt}\}$, with each H composed of multiple models h (trained per cluster), and leaves l,
$n_{min} \leftarrow$ Grace period,
$\delta \leftarrow$ Confidence levels,
$\varepsilon \leftarrow$ Hoeffding-based confidence interval,
$\tau \leftarrow$ Tie-breaking thresholds,
$d_1 \leftarrow$ Maximum distance between any two elements in a cluster, $d_0 \leftarrow$ Minimum pairwise distance in a cluster, $d \leftarrow$ Average dissimilarity across all pairs,
$R \leftarrow$ Range of dissimilarity measure,
$n \leftarrow$ Number of instances at a node or cluster,
$\mathrm{d}(C) \leftarrow$ Clustering statistics for parent/sibling clusters, where C_l is the cluster of a given leaf node, C_{sib} its sibling, C_{parent} their common parent,
$|L| \leftarrow$ Number of known label sets,
$e, e' \leftarrow$ ADWIN error rates,
$\sigma^2 \leftarrow$ Variances of predictions,
$W, W' \leftarrow$ Counts of instances in main and alternate branches,
$G \leftarrow$ Heuristic function,
$t_{statistic} \leftarrow$ Welch's test threshold

for each (x, y) in stream **do**
  for each cluster hierarchy H of $\{H_{main}, H_{alt}\}$ **do**
    Route x to clusters in H, and predict disjoint $\hat{y}$
    // Online divisive agglomerative clustering
    for each leaf l in H not yet tested **do**
      Update $\mathrm{d}(Jaccard)$
      $\varepsilon_l = \sqrt{\frac{R^2 \ln(1/\delta_{cluster})}{2n}}$
      if $\frac{(d_1 - d_0)}{|d_1 + d_0 - 2d|} > \varepsilon_l$ and $\varepsilon_l < \tau_{cluster}$ **then**
        Split leaf l
        Initiate new clusters as pivot variables
        Assign remaining based on pivot proximity
      end if
      if $2 \cdot \mathrm{d}(C_{parent}) - (\mathrm{d}(C_l) + \mathrm{d}(C_{sib})) \geq \varepsilon_1$ **then**
        Re-aggregate C_l and C_{sib} into C_{parent}
        Reinitialize statistics at C_{parent}
        Generate balanced structure
      end if
    end for
    for each model h in cluster hierarchy H **do**
      for each node n in the path to l **do**
        Update node statistics with (x, y)
        if alternate subtree $T_{\mathrm{alt}}(n)$ exists **then**
          Recursively update $T_{\mathrm{alt}}(n)$ with (x, y)
        end if
        Update ADWIN with $(\hat{y}, y)$
        if $T_{\mathrm{alt}}(n)$ exists and $W(T_{\mathrm{alt}}(n)) > n_{\min}$ **then**
          $\varepsilon_{alt} = \sqrt{\frac{2e(1-e')(W+W')\ln(2/\delta_{alt-tree})}{W \cdot W'}}$
          if $e - e' > \delta_{alt-tree}$ **then**
            Replace main one with $T_{\mathrm{alt}}(n)$
          else if $e' - e > \delta_{alt-tree}$ **then**
            Prune $T_{\mathrm{alt}}(n)$
          end if
        end if
        if ADWIN warns and no $T_{\mathrm{alt}}(n)$ exists **then**
          $T_{\mathrm{alt}}(n) \leftarrow$ new substitute subtree at n
        end if
        if n is a leaf l, and seen instances $> n_{\min}$ **then**
          Evaluate splits using heuristic G
          $\varepsilon_l = \sqrt{\frac{\log_2(|L|)^2 \ln(1/\delta_{tree})}{2n}}$
          if $G_{\mathrm{best}} - G_{\mathrm{sec}} > \varepsilon_l$ and $\varepsilon_l < \tau_{tree}$ **then**
            Split leaf l on best attribute
            Create child nodes
          end if
        end if
      end for
    end for
  end for
  Concatenate disjoint $(\hat{y}, y)$, and compute errors α_{main}, α_{alt}
  if $\frac{\alpha_{main} - \alpha_{alt}}{\sqrt{\frac{\sigma^2_{main}}{N_{main}} + \frac{\sigma^2_{alt}}{N_{alt}}}} < t_{statistic}$ **then**
    Replace the H_{main} with H_{alt}
  else if drifting H_{main} **then**
    (Re)initialize H_{alt}
  end if
end for
Output: Final evaluation metrics

Chapter 4

iHOMER results

Experiments across 23 real-world datasets show iHOMER outperforms 5 state-of-the-art global baselines by 23%, 12 local baselines by 32%, and random search by 40%, demonstrating the effectiveness of dynamic label space clustering in SMLL.

iHOMER was compared against 19 other multi-label incremental algorithms proposed over the past 25 years, covering a broad spectrum of adaptability to both concept drift and class imbalance. Some of these algorithms address both challenges, while others focus on only one or neither. Within local models, binary relevance transformations of classical incremental decision tree models, like Hoeffding Tree (HT) (Domingos and Hulten, 2000), Hoeffding Adaptive Tree (HAT) (Bifet and Gavalda, 2009), and Extremely Fast Decision Tree (EFDT) (Manapragada et al., 2018), were included, as well as more recent approaches such as Stochastic Gradient Tree (SGT) (Gouk et al., 2019), Mondrian Tree (MT) (Mourtada et al., 2021), Aggregated Mondrian Forest (AMF) (Mourtada et al., 2021), and Adaptive Random Forest (ARF) (Gomes et al., 2017a). Moreover, the comparison spans diverse families of algorithms, including Gaussian Naïve Bayes (NB) (Bifet et al., 2009), k-Nearest Neighbors (kNN) (Montiel et al., 2021), Adaptive Model Rules (AMR) (Duarte et al., 2016), and ensemble methods like ADWIN Bagging of Logistic Regression (ABALR) (Bifet et al., 2009) and ADWIN Boosting of Logistic Regression (ABOLR) (Bifet et al., 2009). Within global models, it also includes specialized multi-label methods like Multi-Label Hoeffding Tree (MLHT) (Read et al., 2012), MLHT with Prune Set (MLHTPS) (Read et al., 2012), Incremental Structured Output Prediction Tree (iSOUPT) (Osojnik et al., 2017), Multi-Label Broad Ensemble Learning System (MLBELS) (Bakhshi and Can, 2024), and Multi-Label Hoeffding Adaptive Tree (MLHAT) (Esteban et al., 2024). To enable a fair and reproducible comparison, the same hyperparameter configurations and evaluation protocols as those reported in the associated open-source repository referenced in the literature (Esteban et al., 2024). A detailed description of each algorithm's characteristics, along with the full parameter specifications used in the experiments, is available therein.

Evaluation metrics. Given the incremental nature of data streams, the standard prequential

evaluation (also known as test-then-train (Gama et al., 2009)) was adopted. This approach evaluates the model on each instance before using it for training, allowing for continuous monitoring of performance over time. In multi-label classification, each instance may be associated with multiple labels simultaneously, requiring evaluation metrics that account for this complexity. For this purpose, four widely used metrics that capture different performance aspects were considered: Sample Accuracy, Subset Accuracy, Micro F1, Macro F1, and Hamming Loss. To visualize how performance evolves over time, the rolling Sample Accuracy was tracked using a sliding window approach. Subset Accuracy (also known as Exact Match) measures the proportion of instances where all predicted labels match exactly, whereas Sample Accuracy reflects partial correctness using the *Jaccard* index per instance. Micro and Macro F1 balance precision and recall to provide a single multi-label classification performance metric. The difference between these reveals class imbalance effects, since Macro F1 highlights performance on rare labels, while Micro F1 emphasizes overall instance-level accuracy. Let $Y_i = \{y_{i1}, \ldots, y_{iL}\}$ and $Z_i = \{z_{i1}, \ldots, z_{iL}\}$ be the true and predicted label sets for the i-th instance, respectively, with L labels and n instances, the metrics are defined as:

$$\text{Subset Accuracy} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}[Y_i = Z_i] \tag{4.1}$$

$$\text{Hamming Loss} = \frac{1}{nL}\sum_{i=1}^{n}\sum_{l=1}^{L} \mathbf{1}[y_{il} \neq z_{il}] \tag{4.2}$$

$$\text{Macro F1} = \frac{1}{L}\sum_{l=1}^{L} \frac{2 \cdot \sum_{i=1}^{n} \mathbf{1}[y_{il} = 1 \wedge z_{il} = 1]}{\sum_{i=1}^{n} \mathbf{1}[y_{il} = 1] + \sum_{i=1}^{n} \mathbf{1}[z_{il} = 1]} \tag{4.3}$$

$$\text{Micro F1} = \frac{2 \cdot \sum_{l=1}^{L}\sum_{i=1}^{n} \mathbf{1}[y_{il} = 1 \wedge z_{il} = 1]}{\sum_{l=1}^{L}\sum_{i=1}^{n} \mathbf{1}[y_{il} = 1] + \sum_{l=1}^{L}\sum_{i=1}^{n} \mathbf{1}[z_{il} = 1]} \tag{4.4}$$

Datasets. Table 4.1 provides a detailed overview of the real-world datasets used in this study, which are publicly available and widely used in multi-label classification tasks. These datasets range from small to large-scale, with up to 269,648 instances, 1836 features, and 374 labels, making them highly relevant for evaluating the scalability of iHOMER in high-dimensional streams. Additionally, the table includes key multi-label statistics such as *cardinality* (the average number of labels per instance), *density* (cardinality divided by the total number of labels), and the *mean imbalance ratio* (the average degree of label imbalance). The last column of the table indicates whether the instances in each dataset are presented in a temporal order: yes (✓), no (×), or unknown (-). While these datasets do not explicitly provide information about concept drift, the temporal ordering can offer insight into the potential for concept drift.

Table 4.1: Real-world datasets used. Temp. = Temporally ordered.

Dataset	Inst.	Feat.	Lbls	Card.	Dens.	MeanIR	Temp.
Flags	194	19	7	3.39	0.48	2.255	×
WaterQuality	1.060	16	14	5.07	0.36	1.767	-
Emotions	593	72	6	1.87	0.31	1.478	×
VirusGO	207	749	6	1.22	0.20	4.041	×
Birds	645	260	19	1.01	0.05	5.407	-
Yeast	2.417	103	14	4.24	0.30	7.197	×
Scene	2.407	294	6	1.07	0.18	1.254	×
CAL500	502	68	174	26.04	0.15	20.578	×
Human	3.106	440	14	1.19	0.08	15.289	-
Yelp	10.806	671	5	1.64	0.33	2.876	✓
Medical	978	1.449	45	1.25	0.03	89.501	✓
Eukaryote	7.766	440	22	1.15	0.05	45.012	-
Slashdot	3.782	1.079	22	1.18	0.05	19.462	✓
Hypercube	100.000	100	10	1.00	0.10	-	-
Hypersphere	100.000	100	10	2.31	0.23	-	-
Langlog	1.460	1.004	75	15.94	0.21	39.267	✓
Stackex	1.675	585	227	2.41	0.01	85.790	✓
Tmc	28.596	500	22	2.22	0.10	17.134	✓
Ohsumed	13.929	1.002	23	0.81	0.04	7.869	✓
D20ng	19.300	1.006	20	1.42	0.07	1.007	✓
Bibtex	7.395	1.836	159	2.40	0.02	12.498	×
Nuswidec	269.648	129	81	1.87	0.02	95.119	-
Imdb	120.919	1.001	28	1.00	0.04	25.124	×

Average performance. Table 4.2 presents the average performance across four key metrics. Among all evaluated models, iHOMER achieves the best overall performance on three of the four metrics, i.e., Subset Accuracy, Micro F1, and Hamming Loss, highlighting its ability to adaptively model label dependencies through dynamic label clustering. This hybrid global-local architecture effectively captures co-occurrence patterns that static approaches miss. Compared to both global and local baselines, iHOMER shows consistent and substantial gains, particularly in Subset Accuracy (0.387) and Micro F1 (0.553). The closest competitor in Macro F1 is MLHAT, a global method, which outperforms iHOMER in that metric (0.445 vs. 0.282), likely due to a more balanced treatment of rare labels. In contrast, local models, which treat labels or label subsets independently, generally perform worse, especially on metrics sensitive to inter-label dependencies, such as Subset Accuracy and Micro F1. Nevertheless, local learners such as EFDT, HAT, kNN, and AMR emerged as the most competitive within their category. While these models demonstrate some capacity to model local patterns, they fall short when label relationships become complex or shift over time.

Table 4.2: Average performance across real-world datasets. Methods are grouped by Type and ordered by Subset Accuracy within each group.

Method	Subset Acc.	Micro F1	Macro F1	H. Loss	Type
iHOMER	**0.387**	**0.553**	0.282	**0.093**	G+L
MLHAT	0.351	0.550	**0.445**	0.105	G
MLBELS	0.281	0.502	0.326	0.123	G
iSOUPT	0.269	0.436	0.303	0.114	G
MLHT	0.178	0.259	0.136	0.160	G
MLHTPS	0.175	0.298	0.208	0.132	G
ABALR	0.279	0.435	0.318	0.109	L
ABOLR	0.277	0.433	0.319	0.110	L
KNN	0.275	0.430	0.303	0.111	L
AMR	0.267	0.428	0.312	0.113	L
ARF	0.260	0.417	0.288	0.098	L
AMF	0.243	0.403	0.281	0.101	L
EFDT	0.230	0.434	0.299	0.110	L
HAT	0.240	0.423	0.287	0.110	L
HT	0.204	0.385	0.257	0.116	L
MT	0.171	0.288	0.188	0.123	L
NB	0.133	0.301	0.225	0.156	L
SGT	0.049	0.199	0.153	0.300	L

Figure 4.1 shows the distribution of Hamming Loss and Subset Accuracy for all methods across each dataset, represented as interquartile boxplots. iHOMER's performance is marked by an orange dot. The datasets are arranged along the x-axis based on iHOMER's relative rank within each distribution, from best (lowest Hamming Loss and highest Accuracy, respectively) to worst. This visualization highlights iHOMER's consistent performance and its comparative standing against other methods, independent of the number of instances, features, or labels.

For Hamming Loss, iHOMER consistently falls below the median (where lower values are better), outperforming other methods in 75% of the cases. Additionally, for Subset Accuracy, iHOMER performs above the median in 90% of the cases, demonstrating its strong performance across most datasets.

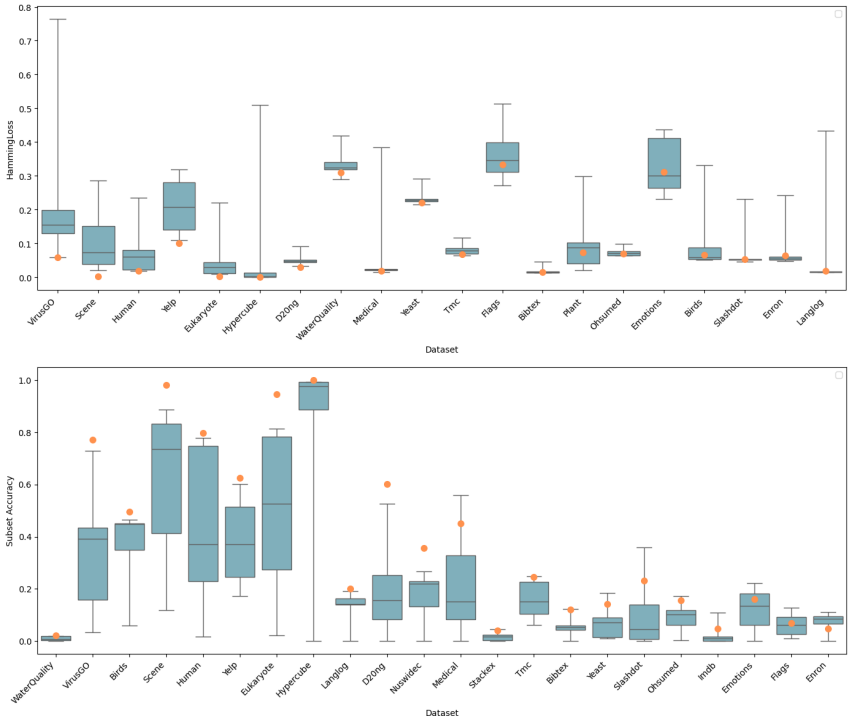


Figure 4.1: Distribution of Hamming Loss and Subset Accuracy scores across datasets. The boxplots represent the performance of all methods except iHOMER, while the orange dots indicate iHOMER's position within each dataset. Datasets are ordered from left to right by increasing percentile rank of iHOMER (lower is better for Hamming Loss and higher is better for Subset Accuracy).

Critical difference diagrams. To validate the statistical significance of performance differences, the Friedman test was conducted, with the resulting statistics ranging between approximately 122 and 148 for all metrics. In all cases, the p-value was zero, indicating a clear rejection of the null hypothesis that all methods perform equally. A Nemenyi post-hoc test was then applied, with the resulting Critical Difference Diagrams (CDDs) for 95% confidence in Figure 4.4, further corroborating its consistent performance across all datasets. Notably, iHOMER ranks among the

top three methods in all CDDs, underscoring its robustness across both label-based and instance-based performance criteria. In Subset Accuracy and Micro F1, iHOMER holds a statistically significant lead over most local learners and several global ones, affirming its effectiveness in capturing high-fidelity label combinations and optimizing overall predictive balance. While iHOMER remains highly competitive in Hamming Loss, it does not claim the top position. This outcome is partly attributable to the metric's design: Hamming Loss penalizes every individual label misclassification equally, regardless of whether the rest of the label set is correctly predicted. As a hybrid model that prioritizes structured label dependencies and exact match performance, iHOMER optimizes for holistic correctness rather than minimizing marginal errors. Consequently, it achieves higher Subset Accuracy and F1 scores at the expense of slightly increased label-wise mismatches, which are more harshly reflected in Hamming Loss.

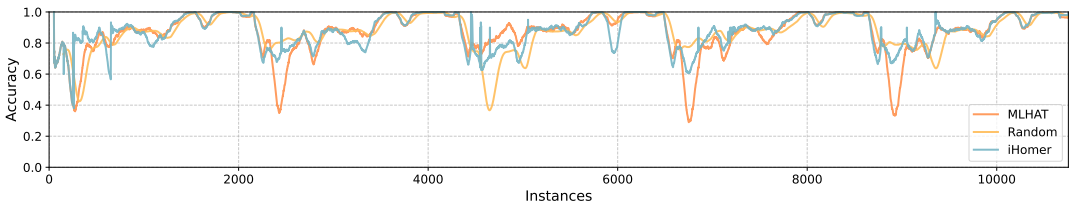


Figure 4.2: Rolling Sample Accuracy for Yelp dataset.

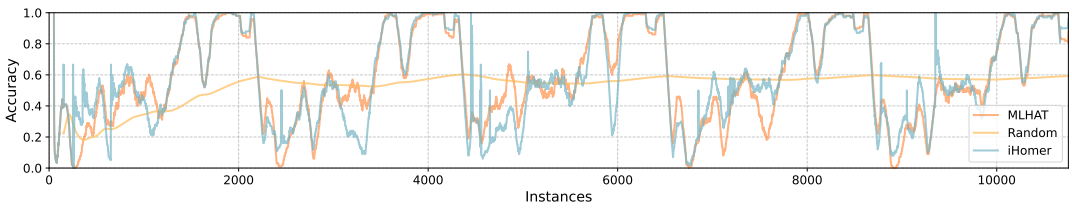
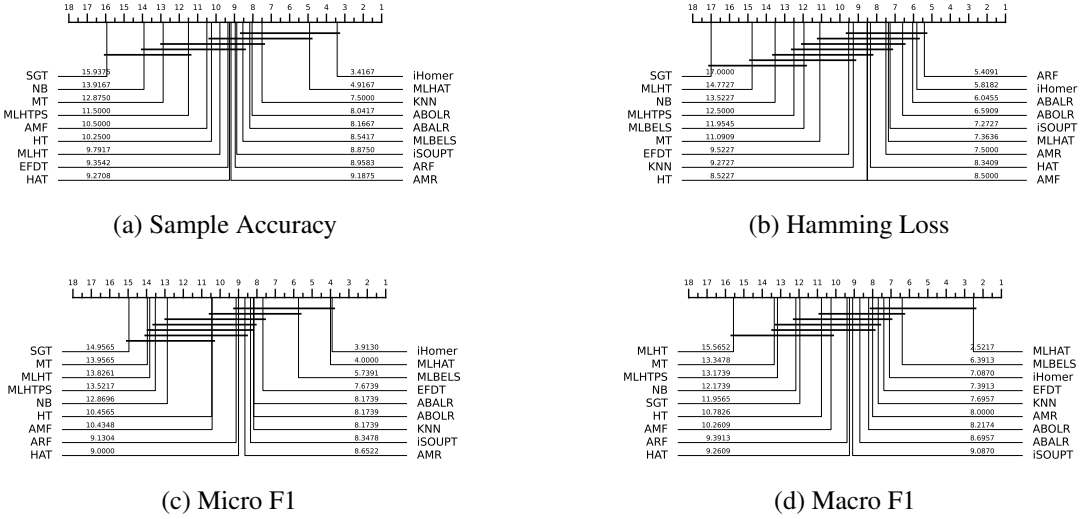


Figure 4.3: Rolling Subset Accuracy for Yelp dataset.



(a) Sample Accuracy

(b) Hamming Loss

(c) Micro F1

(d) Macro F1

Figure 4.4: Critical Difference Diagrams from the Nemenyi test with 95% confidence for Sample Accuracy, Hamming Loss, Micro F1, and Macro F1 across real-world datasets.

iHOMER vs. MLHAT. While MLHAT is recognized in the literature as a strong multi-label classifier, iHOMER demonstrates superior performance across diverse scenarios characterized by varying cardinality, density, and concept drift. These results highlight iHOMER's ability to accurately model complex label dependencies and capture complete label combinations. As shown in Table 4.2, iHOMER achieves a substantial 10.26% improvement in Subset Accuracy (widely regarded as the most rigorous evaluation metric for multi-label tasks) and an 11.43% reduction in Hamming Loss compared to MLHAT. These gains indicate both higher predictive accuracy and fewer misclassifications. iHOMER also shows a modest improvement in Micro F1, while exhibiting slightly lower performance on Macro F1, despite not being explicitly optimized for F1-based objectives. Both models benefit from using multi-label metrics for drift detection, enabling quicker adaptation than methods that handle each label independently. Wilcoxon signed-rank tests confirm that the improvements in Subset Accuracy and Hamming Loss are statistically significant. Although iHOMER also achieves higher average Micro F1 scores, this particular difference does not reach statistical significance. To illustrate iHOMER's superiority on a representative dataset, Figure 4.3 presents the rolling Sample Accuracy over time, highlighting the comparative performance of iHOMER, a single MLHAT, and MLHAT ensembles with randomly generated hybrid partitions. The rolling metric reveals that iHOMER produces smoother, more stable trends and avoids the sharp drops observed in other methods, underscoring the value of informed label space partitioning in streaming data scenarios.

Qualitative analysis. These results confirm that iHOMER consistently outperforms both local and global baselines across multiple evaluation metrics and across datasets of varying sizes and characteristics, demonstrating resilience to noise and fluctuations in the input and output spaces. Although iHOMER achieves lower Macro F1 compared to MLHAT, this can be attributed to the method's prioritization of frequent label groupings in favor of optimizing global predictive performance. This trade-off is expected in systems designed to maximize overall stream accuracy

Table 4.3: Wilcoxon signed-rank test: iHOMER vs. MLHAT

Metric	**Statistic**	**P-value**	**P-value < 0.05**
Subset Acc.	72.0000	0.0249	✓
Micro F1	114.0000	0.4820	
Macro F1	21.0000	0.0001	✓
Hamming Loss	20.0000	0.0002	✓

rather than emphasize rare-label detection. By unifying online label space partitioning with hierarchical adaptation and drift-aware learning, it addresses key limitations of existing methods while maintaining scalability, robustness, and adaptability in dynamic environments.

Chapter 5

iHOMER+

5.1 Problem definition

Although iHOMER effectively adapts a single label-space hierarchy to evolving data streams, its reliance on sequential reconfiguration can introduce latency when concept drift is abrupt or recurrent. During such rapid shifts, the time required to split and regroup clusters means the model may underperform until the new hierarchy stabilizes. Consequently, there is a need for a multi-model strategy that preserves iHOMER's stability during stationary periods yet responds immediately to sudden changes, minimizing prediction delay and error spikes. Moreover, the restructuring of label partitions and base learners can be conservative, as the system must accumulate sufficient statistical evidence before replacing outdated components. This often results in a trade-off between model stability and reactivity.

5.2 Overview

To address these limitations, iHOMER+ extends the original framework with a three-tier ensemble structure specifically designed for robust concept drift handling that maintains three global learners concurrently: an **active** learner that favors stability and performs well in stationary or slowly drifting segments; a **secondary** learner that rapidly adapts using the most recent, high-performing label clustering structure; and an **alternate** learner that is initialized only after multiple consecutive drift signals, indicating that the current models may be misaligned with the data. All three employ the Jaccard-based, divisive-agglomerative clustering and MLHAT base learners of iHOMER, but differ in their window sizes and split/prune thresholds.

This architecture enables iHOMER+ to simultaneously balance short-term adaptability and long-term stability. In the presence of sudden or sustained drift, predictions from these learners are combined and weighted based on their error, allowing the ensemble to quickly shift focus toward the most suitable model. By incorporating redundancy and temporal specialization, iHOMER+ ensures faster recovery from concept drift and improved responsiveness without sacrificing the structural insights and scalability of the original iHOMER design.

5.3 Detailed methodology

5.3.1 Ensembles

The idea behind iHOMER+ stems from a possible categorization of methods based on whether they're (1) ensemble of global, (2) global, (3) ensemble of local, and (4) local methods to explore correlations in clusters of labels.

Firstly, from an ensemble of global perspective, correlations between labels are modeled at a dataset-level with the Jaccard similarity measure, clustered in order to obtain hybrid partitions of the label space, and validated with drift detection mechanism that monitors multi-label accuracy to trigger manage an ensemble of three global learners concurrently, denoted as: (1) active, which adapts more slowly performing better in stationary or gradually evolving segments of the stream, (2) secondary, trained using the best-performing recent label clustering structure, adapting quickly to current trends and abrupt drifts (3) alternate, that is initialized when multiple consecutive drift signals are detected, suggesting both current models may be misaligned with the data stream. Moreover, leveraging on the fact that in the midst of a concept drift, there will be three global models, their predictions is combined, weighted based on the error monitored, and conditioned on the alternate tree having received at least $n_{\min}$ instances.

Secondly, from a global perspective, the tree-based learner uses a multivariate Bernoulli process Dai et al. (2013) to calculate the entropy, which considers groups of labels that appear together with higher probability, leading to more accurate approximations of the information gain, while partitioning the instance with the tree structure itself. Moreover, a drift detector is incorporated at each node to monitor multi-label accuracy and it accelerates the reaction to concept drift by triggering an early warning to build a parallel sub-tree in the background of the current node, which replaces the main one if the concept drift is confirmed.

Thirdly, from an ensemble of local perspective, to deal with imbalanced label sets, four prediction scenarios are applied at the leaves, based on values of multivariate binary entropy and cardinality. In face of a scenario with high difficulty, the tree leaf will dynamically adopt a multi-label bagging classifier that allows to partition the instance across the various local classifiers.

Fourthly, from a local perspective, in low difficulty scenarios, the tree leaf will dynamically shift to a multi-label kNN classifier, or majority voting.

Ensemble of global. In multi-label learning, the quality of model performance often depends on the effective utilization of label correlations. In streaming environments, labels typically exhibit complex relationships, including rich semantic information, hierarchical structures, various intricate associations, and ambiguities, that evolve over time. Efficiently leveraging label correlations can assist the model in better learning feature representations, boosting model generalization ability and training efficiency across drifting environments. In this regard, most current methods in the literature, such as MLHAT Esteban et al. (2024), in the process of exploring label correlations, abstract the complex relationships between labels further and reduces them to a co-occurrence relationship. When two label objects frequently appear together, they are considered to have a strong positive association. In binary label spaces, where inputs consist of 0s and 1s denoting

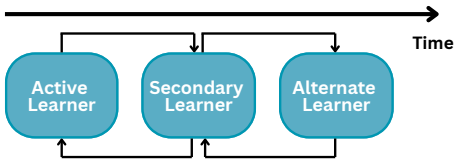


Figure 5.1: iHOMER+ architecture with three learners: Active, Secondary, and Alternate. Arrows show potential switching based on performance for adaptive learning.

label absence or presence, as illustrated in Table 3.1 one must devise a way to model the number of occurrences and co-occurrences associated with each pair of labels in the label space.

Additionally, iHOMER+ adopts the same procedure as iHOMER for modeling label correlations, including the incremental construction and aggregation of the output space hierarchy based on Jaccard similarity and statistical criteria.

5.3.2 Three-tier output space partitioning ensemble.

The architecture of iHOMER+ features a sophisticated three-tier ensemble design to enhance responsiveness to evolving data distributions and maintains up to three learners concurrently during the learning process, denoted as (1) active, (2) secondary, (3) alternate, as illustrated in Figure 5.1.

The active learner, functioning as the main ensemble, serves as the current decision-making model responsible for all predictions. Supporting this, the secondary ensemble is trained using the best-performing recent label clustering structure. The active and secondary learners specialize in different temporal regions of the data stream. The more recent learner typically adapts quickly to current trends, making it particularly effective when dealing with abrupt concept drifts. In contrast, the older and generally more stable learner adapts more slowly but tends to perform better in stationary or gradually evolving segments of the stream. This secondary learner stands ready to replace the active learner when concept drift is detected. When multiple consecutive drift signals are detected, suggesting both current models may be misaligned with the data stream, an alternative learner is initialized. This third component explores a new representation of the data stream and can replace the secondary model if it demonstrates superior performance in terms of current subset accuracy. To detect performance degradation and guide model replacement within the iHOMER+ architecture, a statistical comparison between ensembles is conducted, similarly to iHOMER. When the test indicates a statistically significant difference, the model with lower error may replace its counterpart, supporting dynamic adaptation to concept drift in the data stream. While this approach may be more sensitive to noise and short-term fluctuations, the introduction of a grace period n_{min} before replacement decisions helps reduce false alarms and stabilizes system responsiveness. Moreover, leveraging on the fact that in the midst of a concept drift, there will be two global models that are candidates to replace the main predictor whose performance is declining, their predictions is combined, weighted based on the error monitored, and conditioned on the alternate tree having received at least n_{min} instances.

Figure 5.2 illustrates an example of the adaptive mechanism employed by iHOMER+, where different outcomes may unfold in response to concept drift, as previously discussed. One such

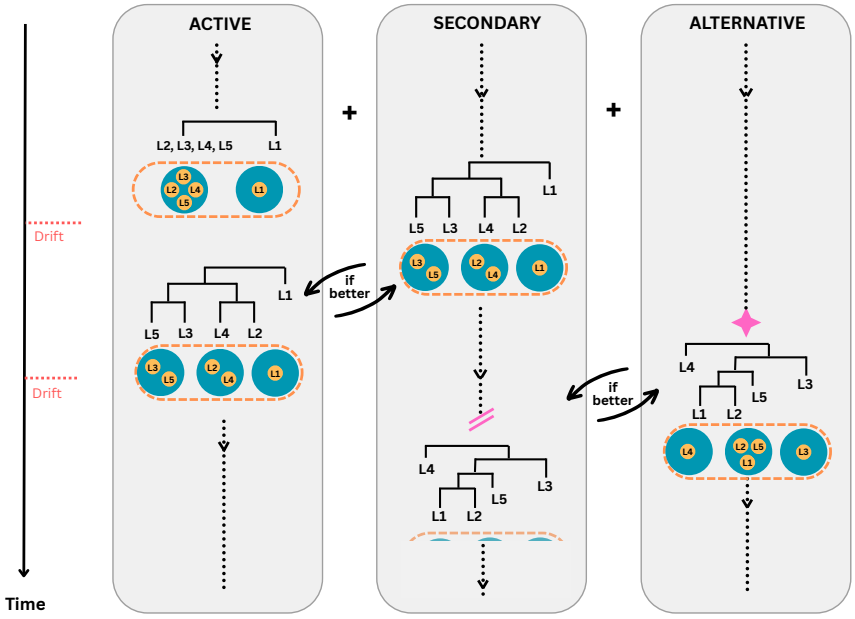


Figure 5.2: Architecture of iHOMER+: An ensemble of three learners (Active, Alternate, and Secondary) dynamically adapts the label hierarchy in response to concept drift. Learner switching is guided by statistical comparison of predictive performance, enabling efficient replacement and recovery.

outcome is the Active–Secondary switch, in which the previously demoted Secondary model, being older and having learned a more complex and more effective structure, outperforms the current Active model and is reinstated for prediction. In this case, more simple model is moved into the Alternate pool. The figure also depicts the initialization of a new Alternate model (indicated by the pink star), which begins to explore a fresh label space partitioning. If it surpasses the current Active model in predictive performance, it replaces it. Altogether, this mechanism maintains a dynamic pool of candidate structures that can be reactivated as the data distribution evolves or reverts. Such a strategy enables rapid and stable adaptation across diverse drift types, while preserving temporal diversity and avoiding overfitting to transient patterns.

5.3.3 Pruning the feature space hierarchy.

Hoeffding Adaptive Tree (HAT) extends the VDFT Domingos and Hulten (2000), using the same splitting strategy, but replacing outdated subtrees with alternates that are grown when a detector detects drift through a changing error rather than on changing split attributes Bifet and Gavalda (2009). Naturally, HAT could also use moving windows of instances, as does the CVFDT Hulten et al. (2001), however, the true flavor of HAT comes from using ADWIN as a drift detector and error estimator, being a parameter-free and robust way to carefully tune an appropriate window on top of changes in the distribution of error values Bifet and Gavalda (2007). When the alternate subtree outperforms the original subtree over a sufficient number of instances, it replaces the latter, with recursive replacements among alternates of alternates being possible. In other words, having

alternative subtrees switch between training and test modes, and during the latter also pruning the alternative subtrees whose accuracy does not increase over time Hulten et al. (2001).

Following this intuition, MLHAT monitors the error in the Hamming loss, triggering replacements based on the Hoeffding bound ε_{alt} for alternating trees Bifet and Gavalda (2009) and defined from monitored errors and instances seen by both main W and alternate W' sub-trees as:

$$\varepsilon_{alt} = \sqrt{\frac{2e(1-e')(W+W')\ln(2/\delta_{alt})}{W \cdot W'}} \tag{5.1}$$

where, (1) if $(e-e') > \delta_{alt}$, the alternate tree replaces the main one, pruning the previous substructure resetting the drift detector in the alternate tree for future concept drifts, (2) if $(e'-e) > \delta_{alt}$, the alternate tree remains without changes and the alternate tree is pruned, the concept drift was reversed, and (3) if $\delta_{alt} \geq |e-e'|$, the two sub-trees continue to receive instances and expand to eventually differentiate their performance. However, while this process aims at reducing bias and help keep the model relevant, by ensuring that new subtrees adapt to the new concept before being deployed, such split revision procedure can also significantly reduce variance by simplifying the model, introducing considerable bias if important information is lost when the subtree is pruned Manapragada et al. (2018).

Moreover, leveraging on the fact that in the midst of a concept drift, there will be multiple alternate sub-trees that are candidates to replace the branches whose performance is declining, their predictions is combined with the main tree, weighted based on the error monitored in each of the leaves involved, and conditioned on the alternate trees having received at least $n_{\min}$ instances. Thus, providing a good balance between the main structure that has the statistical guarantees of the Hoeffding bound, and to provide a fast response to possible concept drifts.

5.3.4 Ensemble of local.

While local models of structured outputs use a collection of predictive models, each predicting a component of the overall structure that needs to be predicted, global predictive models make a single prediction for the entire structured output, i.e., simultaneously predict all of its components. It is important to note, however, that global approaches indirectly follow the same principles, being a big algorithm made of small algorithms, with local classifiers per level, node, or parent node, instead of flat classification. Following this intuition, the learning and prediction strategy doesn't need to be uniform for all data partitions, allowing for better accommodation of the complexities of the data flows, such as drifting imbalances that affect the number of instances received in each leaf node.

Chapter 6

iHOMER+ results

Experimental results across 9 synthetic datasets, each with annotated concept drift, show that iHOMER+ improves predictive performance by 5% over the current state-of-the-art global method and by 29% over six benchmark local methods.

iHOMER+ was compared against a variety of multi-label incremental algorithms. This includes the global model MLHAT (Esteban et al., 2024), as well as binary relevance adaptations of classical local incremental decision tree methods such as Hoeffding Adaptive Tree (HAT) (Bifet and Gavalda, 2009) and Extremely Fast Decision Tree (EFDT) (Manapragada et al., 2018). Additionally, more recent approaches were considered, including Stochastic Gradient Tree (SGT) (Gouk et al., 2019) and Adaptive Random Forest (ARF) (Gomes et al., 2017a). The evaluation also included methods from other algorithmic families, such as k-Nearest Neighbors (kNN) (Montiel et al., 2021) and ADWIN Boosting of Logistic Regression (ABOLR) (Bifet et al., 2009). The evaluation metrics are the same as in iHOMER.

Datasets. Since evaluating adaptive multi-label classifiers in non-stationary environments requires knowledge of when concept drift occurs, an aspect often unavailable in real-world datasets, nine synthetic datasets are explicitly designed to simulate controlled drift scenarios. Six of these are based on the Radial Basis Function (RBF) and Hyperplane (HP) generators, originally developed using the MOA framework (Bifet et al., 2010) and widely used in prior work (Esteban et al., 2024). To complement these, we generate three additional datasets using the `scikit-learn` framework (Pedregosa et al., 2011). Each Random Tree dataset contains 50.000 instances. In the *incremental* variant, the stream is divided into three equal parts, with a smooth probabilistic transition between two concepts occurring in the middle third (from instance 16.660 to 33.330). The *recurrent* variant alternates between two configurations across four equal segments, repeating the first and second concepts. In the *sudden* variant, four abrupt concept drifts are introduced at fixed points in the stream: instances 12.500, 25.000, and 37.500. These drift patterns are illustrated in Table 6.1, and the full set of nine datasets is summarized in Table 6.2.

Table 6.1: Drift configurations for the Random Tree datasets.

Dataset	Drift	Cardinality	Dependency	Description
RTreeInc	Incremental	4, 8	30, 15	Smooth drift introduced by probabilistic mixing of instances from two concepts in the middle segment.
RTreeRec	Recurrent	4, 8, 4, 8	30, 15, 30, 15	Concepts alternate cyclically across four equally sized stream segments.
RTreeSud	Abrupt	1, 6, 4, 8	30, 22, 30, 15	Sharp and immediate concept shifts at fixed points in the stream.

Table 6.2: Synthetic datasets used in the experimental study.

Dataset	Drift Type	Drift Width	Instances	Features	Labels	Generator
HPSud	Sudden	1	50,000	30	8	(Bifet et al., 2010)
HPInc	Incremental	275	50,000	30	8	(Bifet et al., 2010)
HPRec	Recurrent	1	50,000	30	8	(Bifet et al., 2010)
RBFSud	Sudden	1	50,000	80	25	(Bifet et al., 2010)
RBFInc	Incremental	275	50,000	80	25	(Bifet et al., 2010)
RBFRec	Recurrent	1	50,000	80	25	(Bifet et al., 2010)
RTreeSud	Sudden	1	50,000	30	8	(Pedregosa et al., 2011)
RTreeInc	Incremental	1/3 total	50,000	30	8	(Pedregosa et al., 2011)
RTreeRec	Recurrent	1	50,000	30	8	(Pedregosa et al., 2011)

Table 6.3: Performance of *iHOMER+* across synthetic multi-label datasets.

Dataset	Subset Acc.	Sample Acc.	Micro F1	Macro F1	Jaccard	Hamming Loss
HPRec	0.0540	0.7181	0.4443	0.3366	0.3159	0.3218
HPInc	0.0390	0.7242	0.2477	0.2381	0.1652	0.3057
HPSud	0.0398	0.7382	0.3954	0.3597	0.2631	0.3118
RBFRec	0.0239	0.9168	0.2114	0.2059	0.1296	0.1232
RBFInc	0.0517	0.9092	0.2316	0.2130	0.1468	0.1108
RBFSud	0.0711	0.9151	0.2940	0.2847	0.1869	0.1049
RTreeRec	0.1099	0.7427	0.8355	0.7908	0.7159	0.2573
RTreeInc	0.0900	0.7206	0.8225	0.7927	0.6958	0.2794
RTreeSud	0.1026	0.7445	0.8372	0.7924	0.7181	0.2555

Different concept drift scenarios. To evaluate the adaptability of *iHOMER+* under different concept drift scenarios, we analyzed both its predictive performance and structural dynamics across nine synthetic datasets with controlled drift settings (detailed results in Table 6.3). These include sudden, incremental, and recurrent drift types, generated via Random Tree, Random Radial Basis Function and Hyperplane generators.

MLHAT VS. *iHOMER+* . To assess the strengths of *iHOMER+*, we compare it against two MLHAT-based baselines: (1) the standard MLHAT model with a fixed label structure, and (2) an ensemble of MLHATs trained on randomly selected label subsets of varying sizes. As shown in Figure 6.1a, *iHOMER+* consistently outperforms both approaches, exhibiting faster adaptation post-drift and stronger alignment with evolving label dependencies. This advantage arises from its dynamic restructuring of label partitions in response to performance feedback, a capability absent in the fixed-structure baselines. Additionally, across all plots in Figures 6.1a, 6.2, and 6.3a, we observe early patterns of model replacement, as the optimal label hierarchy and clustering are still being learned. Naturally, these structures improve over time, leading to more stable and accurate predictions.

To further investigate the trade-off between computational cost and predictive accuracy, we evaluate five strategies: the baseline MLHAT, three MLHAT variants using disjoint label partitions (splitting the label set into two to four subsets), and *iHOMER+*. In the partitioned variants, each subset is assigned to a separate MLHAT model, and final predictions are aggregated across all sub-models. The choice to explore up to four partitions reflects a practical upper bound on specialization, corresponding to ensembles of up to four specialized MLHAT learners. This setting allows a meaningful comparison to *iHOMER+*, which dynamically maintains a maximum of three active learners at any time based on performance-driven restructuring. Unlike fixed partitions, *iHOMER+* adapts its label grouping in real time in response to concept drift.

Across all datasets, results show a consistent pattern: ensembles built from random label partitions offer limited benefit, as they fail to exploit underlying label dependencies. *iHOMER+* achieves substantially higher predictive performance, particularly in datasets with structured label relationships. For instance, on `SynHPRec` and `SynHPSud`, *iHOMER+* reaches sample accuracies of 0.72 and 0.74, compared to 0.68 and 0.69, respectively, achieved by the baseline MLHAT (see Figure 6.4). On data streams with less labels, such as those generated by random trees, the performance differences among strategies are less pronounced. While *iHOMER+* incurs a higher computational cost than the standard MLHAT (Table 6.4), the overhead remains moderate. On average, *iHOMER+* requires 1433 seconds per dataset, compared to 686 seconds for the basic MLHAT. Notably, it is often more efficient than the most expensive MLHAT variant (Split 4), which averages 1663 seconds but yields lower accuracy. This underscores that disjoint-label MLHAT strategies introduce runtime costs without commensurate performance improvements, whereas *iHOMER+* achieves significant accuracy gains with reasonable computational demands.

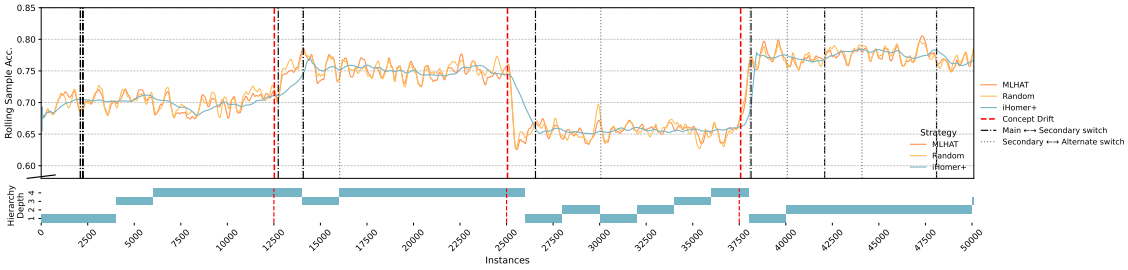


Figure 6.1: Comparison of rolling sample accuracy and hierarchy depth over time for the RTreeRec dataset.

(a) The top plot shows the performance of *iHOMER+*, MLHAT, and a random baseline, with vertical lines indicating concept drift events (red), switches between main and secondary learners (black), and between secondary and alternate learners (gray). The bottom heatmap illustrates the evolving depth of the learned label hierarchy that typically resets occur after each drift, followed by gradual reconstruction.

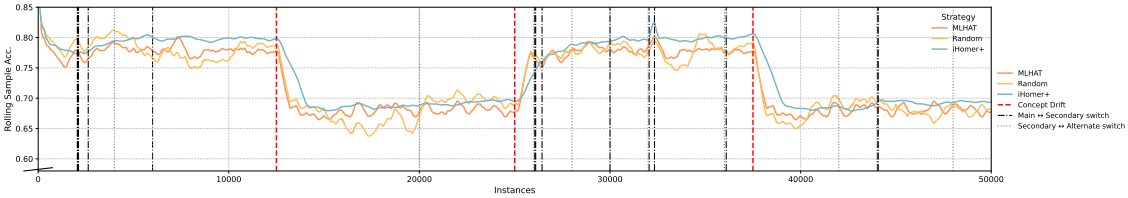


Figure 6.2: Rolling Sample Accuracy for HPRec dataset.

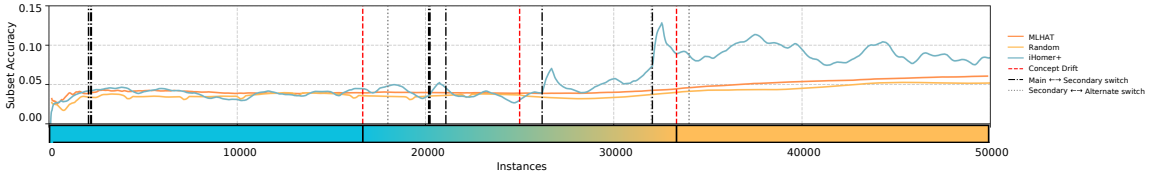


Figure 6.3: Rolling Subset Accuracy for the RTreeInc dataset.

(a) Unlike Figures 6.1a and 6.2, the red lines here simply delimit the region affected by incremental drift, with the central line marking the midpoint of the concept drift.

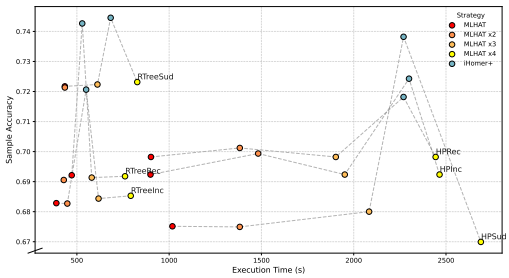


Figure 6.4: Accuracy vs. Execution time for *iHOMER+* and MLHAT variants. Lines connect strategies evaluated on the same dataset.

Dataset	MLHAT	Split2	Split3	Split4	*iHOMER+*
HPInc	135.3	138.9	142.1	145.5	260.9
HPRec	144.0	146.2	148.7	150.1	270.4
HPSud	140.2	150.1	155.4	160.0	290.3
RBFInc	905.1	912.4	920.9	930.0	1233.4
RBFRec	890.6	902.2	918.5	926.3	1328.1
RBFSud	910.0	925.3	940.7	950.1	1290.8
RTreeRec	155.1	159.0	162.2	166.1	275.2
RTreeInc	160.2	165.3	168.4	170.5	270.6
RTreeSud	150.0	153.3	157.2	160.4	280.8
GLOBAL	**454.0**	**472.5**	**479.3**	**484.6**	**666.9**

Table 6.4: Execution time (in seconds) of *iHOMER+* and MLHAT variants across synthetic datasets.

Hierarchy adaptation over time. Following that, we examined how *iHOMER+* adapts its internal label hierarchy during streaming. As concept drift occurs, *iHOMER+* triggers resets in the label clustering structure, leading to a temporary collapse in the hierarchy depth. This behavior, shown in Figure 6.1a for the sudden drift scenario, confirms the model's sensitivity to structural changes. After each drift, the hierarchy is gradually rebuilt, enabling the model to realign with the new concept distribution. This dynamic restructuring is closely tied to *iHOMER+*'s three-tier ensemble. Initially, the main and secondary models may switch roles as the secondary learner, built on a newer clustering, outperforms the main model. When multiple consecutive drifts are detected, the alternate model is initialized and may eventually replace the secondary model if it yields superior performance. These transitions are clearly marked in Figure 6.1a: black lines indicate when the secondary model replaces the main learner; gray lines mark when the alternate model replaces the secondary. This coordinated replacement process enables a smooth recovery from drifts and ensures consistent adaptation. Moreover, its adaptability is reinforced by the dynamic clustering of labels per model, which is particularly effective in handling evolving label dependencies and co-occurrence patterns, as previously adopted in iHOMER.

Performance across methods. Table 6.5 presents the average performance of each method across all synthetic datasets and evaluation metrics. *iHOMER+* achieved the best results on the majority of metrics, including Sample Accuracy (0.8150), Micro F1 (0.4800), and Macro F1 (0.4460), indicating a robust ability to model multi-label concept drift effectively.

While MLHAT, a global model trained on the full label space, maintains competitive results, *iHOMER+* surpasses it as already discussed by integrating global and local perspectives through a dynamic ensemble structure. Interestingly, HAT, a fully local learner operating at the instance level, yields the highest Subset Accuracy (0.0851) and one of the lowest Hamming Loss values (0.1980), suggesting high precision on exact label matches in simpler scenarios. However, its performance lags behind in overall label-wise metrics, which are more appropriate for multi-label contexts. Simpler models such as EFDT and ABOLR underperform across most metrics, with SGT consistently showing the weakest results.

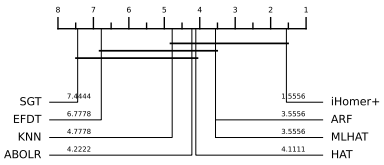
To assess the statistical significance of performance differences across datasets and classifiers, a Friedman test was performed. In all cases, the p-value was zero, leading to a clear rejection of the

Method	Subset Acc.	Sample Acc.	Micro F1	Macro F1	H. Loss
iHOMER+	0.0647	**0.8150**	**0.4800**	**0.4460**	0.2301
MLHAT	0.0619	0.7997	0.4564	0.4363	0.2440
EFDT	0.0385	0.6725	0.3765	0.3547	0.2485
HAT	**0.0851**	0.7580	0.4482	0.4195	0.1980
SGT	0.0108	0.6657	0.1758	0.1365	0.2683
ABOLR	0.0561	0.6845	0.3438	0.3177	**0.1969**
ARF	0.0618	0.8067	0.4210	0.3924	0.2213
KNN	0.0485	0.7367	0.4273	0.4064	0.2189
Friedman χ^2	23.3333	36.9259	30.6296	27.7407	31.4815
P-value	0.0015	0.0000	0.0001	0.0002	0.0001

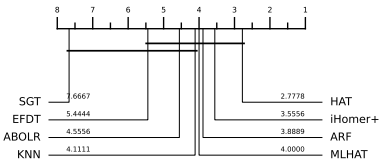
Table 6.5: Average performance across all datasets for each method and evaluation metric.

null hypothesis that all methods perform equally. A subsequent Nemenyi post-hoc test confirmed these differences, and the Critical Difference Diagrams (CDDs) at 95% confidence (Figure 6.5) show that *iHOMER+* consistently ranks among the top-performing methods.

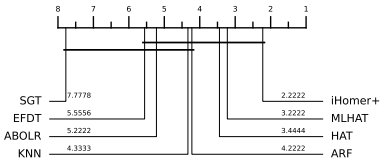
To complement the overall comparison, Table 6.6 reports Wilcoxon signed-rank test results for pairwise comparison between *iHOMER+* and MLHAT. The test reveals that *iHOMER+* significantly outperforms MLHAT in terms of Sample Accuracy, while differences in other metrics were not statistically significant at the 0.05 level. This reinforces the superior reliability of *iHOMER+* in capturing label dependencies and adapting to concept drift in multi-label settings.



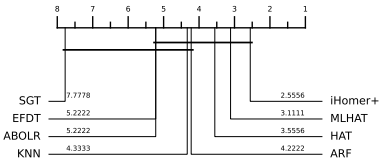
(a) Sample Accuracy



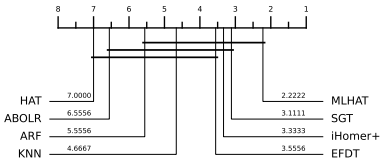
(b) Subset Accuracy



(c) Micro F1



(d) Macro F1



(e) Hamming Loss

Figure 6.5: Critical Difference Diagrams from the Nemenyi test with 95% confidence for Sample Accuracy, Subset Accuracy, Hamming Loss, Micro F1, and Macro F1 across synthetic datasets.

Qualitative analysis. The experimental results demonstrate that, although iHOMER+ requires

Metric	**Statistic**	***P-value***	**P-value < 0.05**
Subset Acc.	19.0000	0.7344	
Sample Acc.	0.0000	0.0039	✓
Micro F1	13.0000	0.3008	
Macro F1	17.0000	0.5703	
Hamming Loss	7.0000	0.0742	

Table 6.6: Wilcoxon signed-rank test results between *iHOMER+* and MLHAT for different metrics in synthetic datasets.

more computational time than MLHAT variants, its hierarchical adaptation mechanism, combined with a structured pool of candidate learners, enables more responsive and stable recovery under concept drift. It consistently achieves higher predictive accuracy across most datasets and while the statistical significance of its improvements is modest, iHOMER+ ranks as the top-performing method overall.

Chapter 7

Conclusions and Future Work

Conclusions

This thesis introduced two complementary frameworks, **iHOMER** and **iHOMER+**, for adaptive multi-label learning from nonstationary data streams. These methods address key challenges in SMLL, including high-dimensional label spaces, evolving label dependencies, class imbalance, and concept drift.

The first contribution, **iHOMER** (Incremental Hierarchy Of Multi-label ClassifiERs), incrementally partitions the output space into disjoint, semantically coherent label clusters. It employs an online divisive-agglomerative clustering approach guided by Jaccard similarity, and uses incremental multi-label Hoeffding trees as global learners within each cluster. To remain robust in nonstationary environments, iHOMER integrates both global and local drift detection mechanisms, enabling dynamic restructuring of the label hierarchy and tree-based substructures. Empirical evaluation across 23 real-world datasets demonstrates that iHOMER consistently outperforms state-of-the-art global and local baselines, improving predictive accuracy by more than 20% on average and exceeding random search approaches by over 40%. These results underscore the benefits of dynamically clustering the label space to better capture underlying dependencies.

Building on iHOMER, this work further proposed **iHOMER+**, the first incremental tree-based ensemble method designed to explicitly model concept drift in multi-label streams. iHOMER+ maintains an ensemble of three concurrent global learners: an *active* model that adapts conservatively and excels in stable segments of the stream; a *secondary* model that responds more aggressively to recent drift; and an *alternate* model that is initialized when drift uncertainty persists. This architecture enables timely and adaptive learner switching based on error trends and statistical significance, using background replacement to preserve responsiveness. Evaluated on nine synthetic datasets with annotated concept drift, iHOMER+ improves subset accuracy by up to 29% over competitive local baselines and achieves consistent gains over strong global models like MLHAT. These findings validate the effectiveness of ensemble-based output space partitioning under drift conditions, demonstrating improved adaptability and predictive robustness.

Together, iHOMER and iHOMER+ provide a principled and scalable approach to multi-label

stream learning, grounded in statistical learning theory and enriched by adaptive, hierarchical modeling of the label space.

Limitations

Despite the encouraging results obtained across real and synthetic datasets, the proposed iHOMER and iHOMER+ frameworks present certain limitations that merit discussion.

First, the performance of both models depends on several hyperparameters, such as the minimum number of instances to trigger splits, confidence levels for drift detection, and thresholds for switching models. Although default values were empirically validated and used consistently across benchmarks, these parameters may need to be retuned for specific application domains or under different drift intensities.

Second, while iHOMER integrates both global and local modeling components through its ensemble of MLHAT classifiers, the underlying label space partitioning is restricted to disjoint clusters (each label is assigned to a single group at a time). Although this non-overlapping structure improves interpretability and efficiency, it may limit the method's ability to capture shared dependencies across overlapping label groups.

Third, iHOMER's adaptive clustering mechanism relies on statistical tests driven by the Hoeffding bound, which assumes a sufficient volume of observations per cluster to detect significant changes. In highly sparse or rapidly evolving streams, these statistical estimates may become noisy or unstable, affecting the quality of both splits and merges over time.

Finally, the synthetic benchmarks used in iHOMER+ enable targeted evaluation under controlled drift settings, but real-world datasets used to evaluate iHOMER lack explicit annotations of drift. As a result, the robustness of the method under naturally occurring, complex drifts is only assessed indirectly through temporal ordering and average performance trends.

Future Work

Several avenues remain for improving both iHOMER and iHOMER+. First, iHOMER's growth efficiency could be further optimized. Although it leverages statistically guided expansion mechanisms from ODAC and MLHAT, the overall computational cost of clustering and tree maintenance remains non-negligible. Adaptive strategies for tuning internal parameters, such as grace periods, tie thresholds, and split confidence levels, could improve both reactivity and scalability. Integrating memory-aware mechanisms, such as selective leaf deactivation or adaptive subtree pruning, would further help focus computation on more active regions of the model, reducing resource consumption without compromising predictive accuracy.

Second, drift detection mechanisms could be enriched to differentiate between noise, temporary shifts, and true concept drifts more effectively. Incorporating robust statistical tests, window-based methods, or even uncertainty quantification techniques could help minimize false positives and stabilize model switching.

Third, while iHOMER and iHOMER+ have been applied to multi-label classification tasks, extending their design to support **multi-target regression** in data streams is a natural and impactful direction. This would require adapting the base learners and partitioning logic to model real-valued label outputs and continuous dependencies between targets.

In summary, the proposed methods not only close important gaps in existing literature but also open promising paths for future research on scalable, interpretable, and adaptive models for complex data streams.

Sustainable Development Goals

Sustainable, inclusive, and ethical innovation has become a core focus in the field of engineering. This thesis contributes primarily to United Nations Sustainable Development Goal (SDG) 9, *Industry, Innovation and Infrastructure*, by addressing key challenges in designing intelligent, resource-efficient systems. The proposed frameworks, iHOMER and iHOMER+, are built to operate under memory and computational constraints, supporting the development of scalable and resilient AI systems. Their impact is demonstrated through performance metrics such as execution time and parameter efficiency, benchmarked against state-of-the-art baselines.

Additionally, this work aligns with SDG 17, *Partnerships for the Goals*, as it was developed through international collaboration between institutions in Portugal and the United States. This includes joint supervision and support from international grants such as FLAD, culminating in conference and journal publications.

Beyond technical innovation, the thesis addresses broader societal and ethical considerations. The integration of iHOMER into real-time systems has potential for high-impact applications in areas such as healthcare, environmental monitoring, and personalized recommendation-domains where adaptive decision-making under uncertainty is critical. By promoting fair, collaborative, and sustainable AI development, this work reflects the core values of engineering for sustainable development.

Appendix

iHOMER+ additional results

Dataset	iHomer+	ABOLR	ARF	EFDT	HAT	KNN	MLHAT	SGT
HPRec	0.7181	0.6846	0.7012	0.6505	0.6859	0.6605	0.6983	0.6657
HPInc	0.7242	0.6846	0.7502	0.6771	0.6866	0.7054	0.6924	0.6657
HPSud	0.7382	0.6846	0.7180	0.6352	0.6600	0.6613	0.6759	0.6657
RBFRec	0.9168	0.6846	0.8944	0.6885	0.8888	0.6758	0.9019	0.6657
RBFInc	0.9092	0.6846	0.9062	0.6989	0.8282	0.8840	0.9065	0.6657
RBFSud	0.9151	0.6846	0.9004	0.6874	0.8293	0.8691	0.9080	0.6657
RTreeInc	0.7206	0.7415	0.6704	0.6109	0.7293	0.7085	0.6828	0.5895
RTreeRec	0.7427	0.7351	0.6479	0.5886	0.7220	0.7001	0.6898	0.5613
RTreeSud	0.7445	0.7585	0.6611	0.6119	0.7467	0.7290	0.7216	0.4814
GLOBAL	**0.7922**	**0.7037**	**0.7569**	**0.6509**	**0.7526**	**0.7328**	**0.7645**	**0.6153**

Table 7.1: Sample Accuracy across datasets using several online multi-label learners.

Dataset	iHomer+	ABOLR	ARF	EFDT	HAT	KNN	MLHAT	SGT
HPSud	0.3597	0.2153	0.2232	0.2244	0.2013	0.3114	0.3459	0.0098
HPInc	0.2381	0.0253	0.0304	0.1068	0.0486	0.1755	0.2497	0.0076
HPRec	0.3366	0.1900	0.2144	0.2415	0.2083	0.2878	0.3639	0.1023
RBFSud	0.2847	0.0384	0.3566	0.2515	0.3408	0.2428	0.2850	0.0075
RBFInc	0.2130	0.0152	0.2866	0.1982	0.2697	0.1683	0.2148	0.0175
RBFRec	0.2059	0.0159	0.3036	0.2293	0.3710	0.1890	0.2412	0.0274
RTreeRec	0.7908	0.7938	0.7147	0.6515	0.7860	0.7681	0.7528	0.5137
RTreeInc	0.7927	0.7964	0.7273	0.6674	0.7890	0.7700	0.7404	0.4676
RTreeSud	0.7924	0.7692	0.6752	0.6221	0.7609	0.7451	0.7334	0.0746
GLOBAL	**0.4685**	**0.3175**	**0.3369**	**0.2439**	**0.3641**	**0.3282**	**0.4266**	**0.1379**

Table 7.2: Macro F1 across datasets using several online multi-label learners.

Dataset	iHomer+	ABOLR	ARF	EFDT	HAT	KNN	MLHAT	SGT
HPRec	0.4443	0.3144	0.3409	0.3525	0.3351	0.3827	0.4678	0.2050
HPInc	0.2477	0.0279	0.0329	0.1059	0.0462	0.1888	0.2582	0.0075
HPSud	0.3954	0.2931	0.2906	0.2750	0.2897	0.3394	0.3640	0.0094
RBFRec	0.2114	0.0160	0.3080	0.2319	0.3743	0.1929	0.2418	0.0286
RBFInc	0.2316	0.0196	0.3099	0.2086	0.2770	0.1825	0.2324	0.0220
RBFSud	0.2940	0.0396	0.3678	0.2566	0.3510	0.2513	0.2921	0.0078
RTreeRec	0.8355	0.7995	0.7195	0.6554	0.7925	0.7736	0.7581	0.5994
RTreeInc	0.8225	0.8067	0.7383	0.6766	0.8000	0.7816	0.7522	0.6112
RTreeSud	0.8372	0.7772	0.6811	0.6261	0.7679	0.7531	0.7408	0.0915
GLOBAL	**0.4902**	**0.3437**	**0.3434**	**0.3324**	**0.4535**	**0.4269**	**0.4562**	**0.1764**

Table 7.3: Micro-F1 across datasets using several online multi-label learners.

Dataset	iHomer+	ABOLR	ARF	EFDT	HAT	KNN	MLHAT	SGT
HPRec	0.3218	0.2236	0.2266	0.2341	0.2231	0.2523	0.2999	0.2324
HPInc	0.3057	0.2411	0.2435	0.2553	0.2418	0.2833	0.3147	0.2426
HPSud	0.3118	0.2659	0.2760	0.2837	0.2716	0.3064	0.3288	0.2817
RBFRec	0.1232	0.0960	0.0793	0.0927	0.0803	0.0916	0.1228	0.0996
RBFInc	0.1108	0.0881	0.0727	0.0894	0.0818	0.0871	0.1115	0.0926
RBFSud	0.1049	0.0927	0.0732	0.0925	0.0817	0.0866	0.1122	0.0978
RTreeRec	0.2573	0.2649	0.3521	0.4114	0.2780	0.2999	0.3102	0.4387
RTreeInc	0.2794	0.2585	0.3296	0.3891	0.2707	0.2915	0.3172	0.4105
RTreeSud	0.2555	0.2415	0.3389	0.3881	0.2533	0.2710	0.2784	0.5186
GLOBAL	**0.2308**	**0.1977**	**0.2435**	**0.2597**	**0.2092**	**0.2190**	**0.2444**	**0.2685**

Table 7.4: Hamming Loss (lower is better) across datasets using several online multi-label learners.

Dataset	iHomer+	MLHAT	ABOLR	ARF	EFDT	HAT	KNN	SGT
RBFSud	0.0711	0.0070	0.1213	0.0525	0.1016	0.0869	0.0659	0.0000
RBFInc	0.0517	0.0033	0.1042	0.0497	0.0841	0.0616	0.0508	0.0013
RBFRec	0.0239	0.0013	0.0763	0.0441	0.0988	0.0501	0.0353	0.0087
HPSud	0.0398	0.0256	0.0224	0.0215	0.0222	0.0375	0.0369	0.0000
HPInc	0.0390	0.0045	0.0045	0.0155	0.0099	0.0303	0.0393	0.0004
HPRec	0.0540	0.0762	0.0848	0.0896	0.0948	0.1011	0.0762	0.0224
RTreeRec	0.1099	0.1029	0.0380	0.0179	0.0959	0.0288	0.0659	0.0127
RTreeInc	0.0900	0.1099	0.0508	0.0227	0.1093	0.0203	0.0613	0.0218
RTreeSud	0.1026	0.1742	0.0541	0.0328	0.1490	0.0202	0.1255	0.0297
GLOBAL	**0.0642**	**0.0561**	**0.0592**	**0.0389**	**0.0842**	**0.0496**	**0.0656**	**0.0106**

Table 7.5: Subset accuracy across datasets using several online multi-label learners.

References

Gabriel Aguiar, Bartosz Krawczyk, and Alberto Cano. A survey on learning from imbalanced data streams: taxonomy, challenges, empirical study, and reproducible experimental framework. *Machine learning*, 113(7):4165–4243, 2024.

Zahra Ahmadi and Stefan Kramer. Modeling multi-label recurrence in data streams. In *2019 IEEE International Conference on Big Knowledge (ICBK)*, pages 9–16. IEEE, 2019.

Gavin Alberghini, Sylvio Barbon Junior, and Alberto Cano. Adaptive ensemble of self-adjusting nearest neighbor subspaces for multi-label drifting data streams. *Neurocomputing*, 481:228–248, 2022.

Ezilda Almeida, Carlos Ferreira, and Joao Gama. Adaptive model rules from data streams. In *ECML PKDD 2013*, pages 480–492. Springer, 2013.

Sepehr Bakhshi and Fazli Can. Balancing efficiency vs. effectiveness and providing missing label robustness in multi-label stream classification. *Knowledge-Based Systems*, 289:111489, 2024.

Márcio Basgalupp, Ricardo Cerri, Leander Schietgat, Isaac Triguero, and Celine Vens. Beyond global and local multi-target learning. *Information Sciences*, 579:508–524, 2021.

Albert Bifet and Ricard Gavalda. Learning from time-changing data with adaptive windowing. In *Proceedings of the 2007 SIAM international conference on data mining*, pages 443–448. SIAM, 2007.

Albert Bifet and Ricard Gavalda. Adaptive learning from evolving data streams. In *IDA*, pages 249–260. Springer, 2009.

Albert Bifet, Geoff Holmes, Bernhard Pfahringer, Richard Kirkby, and Ricard Gavalda. New ensemble methods for evolving data streams. In *15th ACM SIGKDD*, pages 139–148, 2009.

Albert Bifet, Geoff Holmes, Bernhard Pfahringer, Philipp Kranen, Hardy Kremer, Timm Jansen, and Thomas Seidl. Moa: Massive online analysis, a framework for stream classification and clustering. In *Proceedings of the first workshop on applications of pattern analysis*, pages 44–50. PMLR, 2010.

Matthew R Boutell, Jiebo Luo, Xipeng Shen, and Christopher M Brown. Learning multi-label scene classification. *Pattern recognition*, 37(9):1757–1771, 2004.

Martin Breskvar, Dragi Kocev, and Sašo Džeroski. Ensembles for multi-target regression with random output selections. *Machine Learning*, 107:1673–1709, 2018.

Amanda Clare and Ross D King. Knowledge discovery in multi-label phenotype data. In *ECMLP-KDD*, pages 42–53. Springer, 2001.

Bin Dai, Shilin Ding, and Grace Wahba. Multivariate bernoulli distribution. 2013.

Krzysztof Dembczyński, Willem Waegeman, Weiwei Cheng, and Eyke Hüllermeier. On label dependence and loss minimization in multi-label classification. *Machine Learning*, 88:5–45, 2012.

Pedro Domingos and Geoff Hulten. Mining high-speed data streams. In *Sixth ACM SIGKDD*, pages 71–80, 2000.

Joao Duarte and Joao Gama. Multi-target regression from high-speed data streams with adaptive model rules. In *2015 IEEE DSAA*, pages 1–10. IEEE, 2015.

Joao Duarte, João Gama, and Albert Bifet. Adaptive model rules from high-speed data streams. *ACM TKDD*, 10(3):1–22, 2016.

Aurora Esteban, Alberto Cano, Amelia Zafra, and Sebastián Ventura. Hoeffding adaptive trees for multi-label classification on data streams. *Knowledge-Based Systems*, 304:112561, 2024.

Johannes Fürnkranz, Eyke Hüllermeier, Eneldo Loza Mencía, and Klaus Brinker. Multilabel classification via calibrated label ranking. *Machine learning*, 73:133–153, 2008.

Mohamed Medhat Gaber, Arkady Zaslavsky, and Shonali Krishnaswamy. Mining data streams: A review. In *SIGMOD Record*, volume 34, 2005. doi: 10.1145/1083784.1083789.

Joao Gama, Pedro Pereira Rodrigues, and Raquel Sebastiao. Evaluating algorithms that learn from data streams. In *2009 ACM symposium on Applied Computing*, pages 1496–1500, 2009.

Elaine Cecília Gatto, Mauri Ferrandin, and Ricardo Cerri. Multi-label classification with label clusters. *Knowledge and Information Systems*, 67(2):1741–1785, 2025.

Nadia Ghamrawi and Andrew McCallum. Collective multi-label classification. In *14th ACM*, pages 195–200, 2005.

Shantanu Godbole and Sunita Sarawagi. Discriminative methods for multi-labeled classification. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 22–30. Springer, 2004.

Heitor M Gomes, Albert Bifet, Jesse Read, Jean Paul Barddal, Fabrício Enembreck, Bernhard Pfharinger, Geoff Holmes, and Talel Abdessalem. Adaptive random forests for evolving data stream classification. *Machine Learning*, 106:1469–1495, 2017a.

Heitor Murilo Gomes, Jean Paul Barddal, , Fabricio Enembreck, and Albert Bifet. A survey on ensemble learning for data stream classification, 2017b. ISSN 15577341.

Henry Gouk, Bernhard Pfahringer, and Eibe Frank. Stochastic gradient trees. In *Asian conference on machine learning*, pages 1094–1109. PMLR, 2019.

Jun Huang, Guorong Li, Shuhui Wang, Weigang Zhang, and Qingming Huang. Group sensitive classifier chains for multi-label classification. In *2015 ICME*, pages 1–6. IEEE, 2015.

Sheng-Jun Huang and Zhi-Hua Zhou. Multi-label learning by exploiting label correlations locally. In *AAAI*, volume 26, pages 949–955, 2012.

Geoff Hulten, Laurie Spencer, and Pedro Domingos. Mining time-changing data streams. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 97–106, 2001.

Joel D Costa Júnior, Elaine R Faria, Jonathan A Silva, João Gama, and Ricardo Cerri. Novelty detection for multi-label stream classification under extreme verification latency. *Applied Soft Computing*, 141:110265, 2023.

Dragi Kocev, Celine Vens, Jan Struyf, and Sašo Džeroski. Ensembles of multi-objective decision trees. In *ECML 2007*, pages 624–631. Springer, 2007.

Yaning Li and Liu Yang. More correlations better performance: Fully associative networks for multi-label image classification. In *2020 ICPR*, pages 9437–9444. IEEE, 2021.

Afonso Lourenço, João Rodrigo, João Gama, and Goreti Marreiros. Dfdt: Dynamic fast decision tree for iot data stream mining on edge devices. *arXiv preprint arXiv:2502.14011*, 2025.

Gjorgji Madjarov, Dejan Gjorgjevikj, and Sašo Džeroski. Two stage architecture for multi-label learning. *Pattern Recognition*, 45(3):1019–1034, 2012.

Oded Maimon and Lior Rokach. *Introduction to Knowledge Discovery and Data Mining*. 2009. doi: 10.1007/978-0-387-09823-4_1.

Chaitanya Manapragada, Geoffrey I Webb, and Mahsa Salehi. Extremely fast decision tree. In *24th ACM SIGKDD*, pages 1953–1962, 2018.

Saulo Martiello Mastelini, Sylvio Barbon Jr, and André Carlos Ponce de Leon Ferreira de Carvalho. Online multi-target regression trees with stacked leaf models. *arXiv preprint arXiv:1903.12483*, 2019.

Leandro L Minku and Xin Yao. Ddd: A new ensemble approach for dealing with concept drift. *IEEE transactions on knowledge and data engineering*, 24(4):619–633, 2011.

Jacob Montiel, Max Halford, Saulo Martiello Mastelini, Geoffrey Bolmier, Raphael Sourty, Robin Vaysse, Adil Zouitine, Heitor Murilo Gomes, Jesse Read, Talel Abdessalem, et al. River: machine learning for streaming data in python. *Journal of Machine Learning Research*, 22(110): 1–8, 2021.

Jaouad Mourtada, Stéphane Gaïffas, and Erwan Scornet. Amf: Aggregated mondrian forests for online learning. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 83 (3):505–533, 2021.

Jose M Moyano, Eva L Gibaja, Krzysztof J Cios, and Sebastián Ventura. Combining multi-label classifiers based on projections of the output space using evolutionary algorithms. *Knowledge-Based Systems*, 196:105770, 2020.

Aljaž Osojnik, Panče Panov, and Sašo Džeroski. Multi-label classification via multi-target regression on data streams. *Machine Learning*, 106:745–770, 2017.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Arturo Montejo Ráez, Luis Alfonso Urena López, and Ralf Steinberger. Adaptive selection of base classifiers in one-against-all learning for large multi-labeled collections. In *EsTAL 2004*, pages 1–12. Springer, 2004.

Jesse Read, Bernhard Pfahringer, and Geoff Holmes. Multi-label classification using ensembles of pruned sets. In *ICDM 2008*, pages 995–1000. IEEE, 2008.

Jesse Read, Bernhard Pfahringer, Geoff Holmes, and Eibe Frank. Classifier chains for multi-label classification. *Machine learning*, 85:333–359, 2011.

Jesse Read, Albert Bifet, Geoff Holmes, and Bernhard Pfahringer. Scalable and efficient multi-label classification for evolving data streams. *Machine learning*, 88:243–272, 2012.

Pedro Rodrigues, João Gama, and Joao Pedro Pedroso. Hierarchical clustering of time-series data streams. *Knowledge and Data Engineering, IEEE Transactions*, 20:615 – 627, 06 2008.

Martha Roseberry and Alberto Cano. Multi-label knn classifier with self adjusting memory for drifting data streams. In *Second International Workshop on Learning with Imbalanced Domains: Theory and Applications*, pages 23–37. PMLR, 2018.

Martha Roseberry, Bartosz Krawczyk, and Alberto Cano. Multi-label punitive knn with self-adjusting memory for drifting data streams. *TKDD*, 13(6):1–31, 2019.

Martha Roseberry, Bartosz Krawczyk, Youcef Djenouri, and Alberto Cano. Self-adjusting k nearest neighbors for continual learning from multi-label drifting data streams. *Neurocomputing*, 442:10–25, 2021.

Martha Roseberry, Saso Dzeroski, Albert Bifet, and Alberto Cano. Aging and rejuvenating strategies for fading windows in multi-label classification on data streams. In *38th ACM/SIGAPP*, pages 390–397, 2023.

Zhongwei Shi, Yimin Wen, Chao Feng, and Hai Zhao. Drift detection for multi-label data streams based on label grouping and entropy. In *ICDM 2014*, pages 724–731. IEEE, 2014.

Ricardo Sousa and João Gama. Multi-label classification from high-speed data streams with adaptive model rules and random rules. *Progress in artificial intelligence*, 7(3):177–187, 2018.

Eleftherios Spyromitros-Xioufis, Myra Spiliopoulou, Grigorios Tsoumakas, and Ioannis Vlahavas. Dealing with concept drift and class imbalance in multi-label stream classification. *Department of Computer Science, Aristotle University of Thessaloniki*, 2011.

Bozhidar Stevanoski, Ana Kostovska, Panče Panov, and Sašo Džeroski. Change detection and adaptation in multi-target regression on data streams. *Machine Learning*, 113(11):8585–8622, 2024.

Piotr Szymański, Tomasz Kajdanowicz, and Kristian Kersting. How is a data-driven approach better than random choice in label space division for multi-label classification? *Entropy*, 18(8): 282, 2016.

Jimena Torres Tomás, Newton Spolaôr, Everton Alvares Cherman, and Maria Carolina Monard. A framework to generate synthetic multi-label datasets. *Electronic Notes in Theoretical Computer Science*, 302:155–176, 2014.

Grigorios Tsoumakas, Ioannis Katakis, and Ioannis Vlahavas. Effective and efficient multilabel classification in domains with large number of labels. In *ECML/PKDD 2008*, volume 21, pages 53–59, 2008.

Grigorios Tsoumakas, Ioannis Katakis, and Ioannis Vlahavas. Random k-labelsets for multilabel classification. *IEEE transactions on knowledge and data engineering*, 23(7):1079–1089, 2010.

Tong Wei, Jiang-Xin Shi, and Yu-Feng Li. Probabilistic label tree for streaming multi-label learning. In *27th ACM SIGKDD*, pages 1801–1811, 2021.

Wei Weng, Yaojin Lin, Shunxiang Wu, Yuwen Li, and Yun Kang. Multi-label learning based on label-specific features and local pairwise label correlation. *Neurocomputing*, 273:385–394, 2018.

Jiahao Xu, Hongda Tian, Zhiyong Wang, Yang Wang, Wenxiong Kang, and Fang Chen. Joint input and output space learning for multi-label image classification. *IEEE Transactions on Multimedia*, 23:1696–1707, 2020.

Chen Ye, Jian Wu, Victor S Sheng, Pengpeng Zhao, and Zhiming Cui. Multi-label active learning with label correlation for image classification. In *2015 IEEE international conference on image processing (ICIP)*, pages 3437–3441. IEEE, 2015.

Min-Ling Zhang and Lei Wu. Lift: Multi-label learning with label-specific features. *IEEE transactions on pattern analysis and machine intelligence*, 37(1):107–120, 2014.

Min-Ling Zhang and Zhi-Hua Zhou. Ml-knn: A lazy learning approach to multi-label learning. *Pattern recognition*, 40(7):2038–2048, 2007.

Zan Zhang, Zhe Zhang, Jialu Yao, Lin Liu, Jiuyong Li, Gongqing Wu, and Xindong Wu. Multi-label feature selection via adaptive label correlation estimation. *ACM Transactions on Knowledge Discovery from Data*, 17(9):1–28, 2023.