ISSN 2007-9737

Multi-label Classification of IoTData Stream: A Survey

Mashail Althabiti*, Manal Abdullah, Omaima Almatrafi

King Abdulaziz University,
Faculty of Computing and Information Technology,
Saudia Arabia

malthabiti0001@stu.kau.edu.sa, {maaabdullah, olamatrafi}@kau.edu.sa

Abstract. The overall number of Internet of Things (IoT) devices is rapidly growing, generating a massive amount of continuous data stream. The data stream is arriving at a rapid speed, potentially unbounded, which has emerged due to smart services and advanced technologies. Data stream classification is a challenging task that must fulfil stream constraints such as limited memory, a single scan of data, and real-time response. In many emerging applications, stream instances could be associated with more than one class label, as when predicting a given movie genre, different labels may be given: action, horror, adventure, or all, and this refers to Multi-label Classification (MLC). This review mainly aims to review the literature on the multi-label classification task from 2014 to 2023. It examines state-of-the-art versatile MLC methods in general data streams and methods utilized for IoT applications, which are considered one of the main sources of data streams generated by IoT devices. It also focuses on two main challenges: class imbalance and concept drift. It encapsulates the well-known MLC tools and datasets utilized for this task. Moreover, it highlights the gaps that need further attention in future research.

Keywords. Multi-label classification, concept drift, class imbalance.

1 Introduction

A collection of linked smart objects that can self-configure as a dynamic network is known as the Internet of Things (IoT) [1]. It allows the interaction between humans and objects anytime, anywhere, generating massive data streams. The data stream refers to a series of data instances characterized by rapid speed, huge volume, drifting nature, and non-stop arrival [2].

Such data involve valuable information and patterns crucial for many applications and can be discovered using Machine Learning (ML) methods. ML is an Artificial Intelligence (AI) based method that can learn from the experience without being explicitly programmed [3].

The application of ML methods on IoT data stream is found in many applications in different domains such as medicine, economy, entertainment, education, smart cities, and many others. For example, smart cities utilize IoT and ML methods to address urban issues and meet citizens' needs. It has been used in smart homes, waste management, tourism, and smart transportation to facilitate daily tasks or improve productivity.

One ML approach that fits in numerous IoT and smart city tasks is classification, especially in monitoring and control applications [4]. Classification is a supervised ML method that learns from old labelled data and predicts the label of the incoming data stream instances, assuming that only a single label is produced [5].

However, there are numerous real-life situations and applications where data must be classified into multi-label. For example, the text categorization task is considered the primary motivator for Multi-Label Classification (MLC) [6]. Documents can be assigned to several subjects or categories.

MLC may also be used in many application domains, such as emotion recognition, medical diagnosis, image annotation, music/movie categorization, etc. MLC has attracted much research attention due to its significance in several applications of different domains.

However, the MLC of the data stream has some challenges, including limited memory, a single scan of data, and concept drift. Concept drift is the

doi: 10.13053/CyS-28-3-5176

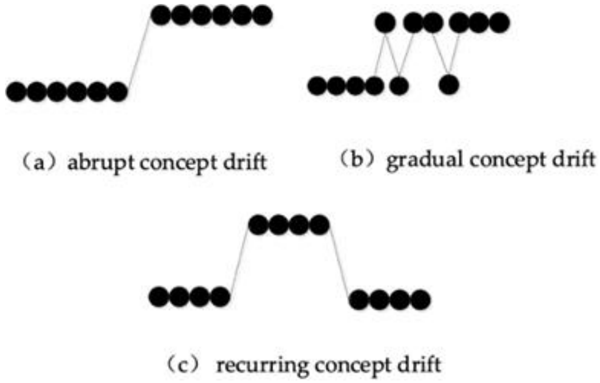

Fig. 1. Concept drift types based on speed [6]

underlying data stream distribution change leading to poor predictions [7]. Such changes can occur abruptly, gradually, or recurrent over time. MLC methods must be able to diagnose when performance degrades and repair to recover to a stable status. In addition, MLC data could arrive with skewed class proportions, another challenge that deteriorates the classifier performance called class imbalance.

Among the different MLC challenges, this paper focuses on class imbalance and concept drift since they are the main characteristics of the data stream. It provides a review of multi-label classification methods used in data streams, generally and in IoT, to identify the state-of-the-art methods in the literature and highlight the gaps to be explored in future work. It mainly answers the following questions,

- Q1: What versatile multi-label classification methods are used to process data streams?
- Q2: What multi-label classification methods were proposed to deal with IoT?
- Q3: Do multi-label classification methods in the literature address Concept Drift and Class Imbalance?
- Q4: What are the main multi-label classification datasets and tools?

The remainder of this article is structured as follows. Section 2 provides a brief background about related concepts.

Section 3 provides the methodology used to conduct the review. Section 4 provides the results of the literature questions. Finally, Section 5 presents the discussion and conclusion.

2 Background

The data stream is a series of instances, $\mathrm{DS} = (x_1, y_1), (x_2, y_2), (x_3, y_3) \dots (x_i, y_i)$ that arrive continuously, where x is the input vector representing a set of features, and y is the corresponding label.

In other words, the data stream is a massive, non-stop sequence of instances arriving at high speed. Nowadays, data streams are generated at a higher speed than ever before due to technological advances. Its high speed, chronological order, infinite and immense volume, and dynamical changes characterize it.

An example of the data stream is NASA satellites generating 4 Terabytes (1012 Bytes) of streaming images in only one day . Also, millions of query streams are processed daily by Google and Yahoo, two popular search engines. The demand for versatile data services is expected to increase by 2050, as it is estimated that around seven billion people will live in urban centres and smart cities.

Smart Cities are where citizens can live in an urban setup surrounded by an intelligent network of interconnected devices and information data that become part of their daily routine.

The connectivity between the main places and services is enabled, such as transportation networks, utility and public services, health care systems, and others, improving citizens' quality of life. Proliferating technologies like Internet of Things (IoT) sensors, cloud computing, machine learning, smartphones, and security systems work hand in hand to build smart cities.

The Internet of Things (IoT) is a network of physical objects or devices integrated with sensors and other technologies. This enables real-time data streams between these devices to be exchanged over the Internet.

By 2030, it is anticipated that 50 billion Internet of Things (IoT) sensors and gadgets will be connected to the Internet over high-capacity, low-latency networks.

As a result, online learning algorithms and intelligent systems require the Internet of Things.

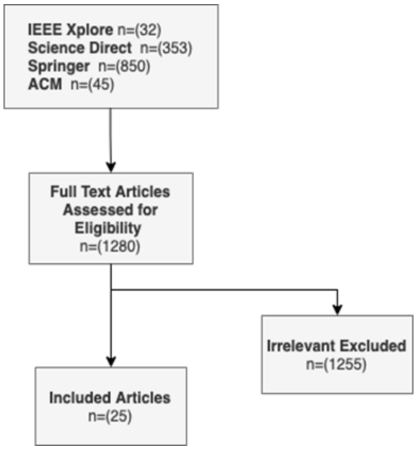

Fig. 2. Steps to select the articles in the literature

Multi-label classification (MLC) is one of the challenges of the data stream. In single-label classification, each data instance is assumed to be assigned to one class. On the contrary, MLC associates a data instance with a set of labels. The multi-label classifier can be defined as $F: x \rightarrow Y$.

The primary function of F is to take the input as any instance $x \in X$, where X is the input space and produce a set of labels $Y \in L$, where L is a set of disjoint labels [8]. An example of an MLC application is image classification, a prominent data type in smart cities collected by IoT devices [9].

Multi-classification algorithms can automatically annotate the input images with appropriate keywords. That would help categorize the streaming images and use them in appropriate applications. The following subsections will present an overview of two main challenges of MLC inthe data stream.

2.1 Concept Drift

Concept drift is one of the data stream phenomena representing the changes in the statistical properties of the features or class labels over time [6].

Formally, the concept drift occurs when the Pt $(x, y) \neq P\_(t + \Delta)\ (x, y)$, where $P(x, y)$ is the joint distribution between data attributes and labels at time t. Concept drift is characterized by the speed in three types, as illustrated in Figure 1, detailed below:

- Abrupt drift Occurs when the underlying data distribution changes suddenly at a particular time with a new concept; therefore, the model built before the drift is unreliable.
- Gradual drift: Occurs when data is alternated between two concepts, meaning new data takes time to substitute the old one.
- Recurrent drift: Occurs when an old concept reappears once or multiple times in the future.

An example showing the importance of detecting concept drift is using a classification model with a drift detector in the Industrial Internet of Things (IIoT). In IIoT, many gadgets are connected [10].

All the machine conditions are collected and shown in a sensor administration system so that the production administrator can ideally control all machines' operations in realtime.

However, machine components may fail or age over time, so the drift detector analyzes the real-time data and alerts when such an abnormal situation occurs. The administrator can replace the damaged components quickly and avoid defective manufacturing products.

2.2 Class Imbalance

Class imbalance is another phenomenon existing in the MLC of the data stream. It refers to the dataset with skewed class proportions, so the classes of large proportion are called the majority classes [11,12]. In contrast, the classes with a smaller proportion are the minority classes.

Due to the presence of majority classes, the classifier tends to ignore minority classes; therefore, the classifier's performance degrades. Class imbalance occurs in several real-world scenarios, such as real-time network monitoring systems, where the classification model must learn from the data streams with skewed class distributions [23].

3 Research Methodology

To conduct the review, five steps are carried out: (1) Define the scope, (2) Search of the articles, (3) Select the articles, then (4) analyze the articles, and finally (5) present the results of the review

[14]. In the first step, the scope of the review is identified. It involves the definition of articles' inclusion and exclusion criteria.

Also, the databases to be searched are selected, and the search terms are formulated. Inclusion and exclusion criteria are mainly defined to set the boundaries of the review and select only articles that fulfil the criteria.

In this research, the inclusion criteria are: (1) selected articles should be written in English, (2) explicitly study the MLC, and (3) the date of publishing should be between 2014 and 2023. Thus, to select the articles included in the review, each should satisfy these criteria; otherwise, it will be excluded. Generally, machine learning could span many disciplines, such as engineering, neuroscience, and mathematics.

This study reviews the literature on multi-label classification from a computer science point of view. Also, four databases have been selected to search for the articles: IEEE Xplore, ScienceDirect, ACM, and SpringerLink. The impact factor of ScienceDirect (Elsevier) Lancet is 202.731, Association for Computing Machinery (ACM) is 16.6, SpringerLink is 2.6. The second step involves searching for the scientific articles in the identified databases using the search terms.

The search terms used to look for the article in selected databases are Multi-label Classification, Concept Drift, Class Imbalance, and Internet of Things (IoT). AND and OR operators have been used to link some terms, which are: (Multi-label classification AND Concept drift), (Multi-label classification AND Class Imbalance), and multi-label classification AND IoT OR Internet of Things). Articles should include one of these search terms to be selected.

In the third phase, the articles are filtered according to the criteria. Only relevant articles are selected, which is done by reading the abstract, title and sometimes the full text.

Also, the citations of each selected article are checked to enrich the sample. Figure 2 shows the steps to select the articles to be included in the literature.

In the fourth step, the selected articles are analyzed, and the findings and insights related to the literature questions and scope are highlighted. Finally, the fifth and last step in conducting the review is to present the analysis results, showing the findings and insights.

4 Results

Twenty-five articles have been selected to be included in the literature. Figure 3 shows the number of publications per year. The results are divided into three subsections as follows:

4.1 Q1: What Versatile Multi-label Classification Methodscan Process Data Streams?

MLC algorithms in a non-stationary environment can be classified into three approaches: Problem Transformation, Algorithm Adaptation and Ensemble [5].

Problem Transformation (PT) simplifies and transforms the MLC problem into simpler single-label classification problems. Thanh et al. [15] believe incremental learning applies to the massive data stream.

They developed a MLC method for drifting streams based on the Bayesian method. The method considers the correlation between the labels to predict the set of labels. It also uses the Hoeffding inequality to set the number of predicted labels. It employs a decay function to detect the drifts, giving recent data more weight.

The method has been assessed in a stream environment and a stationary environment. Braytee et al. [16] have proposed an integrated multi-label classification method called (ML-CIB).

ML-CIB mainly tackles the label incompleteness and class imbalance. It utilizes the label regularizer to handle the class imbalance and learns a new label matrix to handle the missing label.

The authors have also introduced a multi-label feature selection for an effective, relevant feature selection. Ding et al. [17] proposed an algorithm for processing multi-label imbalanced data based on majority and minority assessment. It eliminates the majority of classes whose number exceeds the penalty value as a preprocessing method.

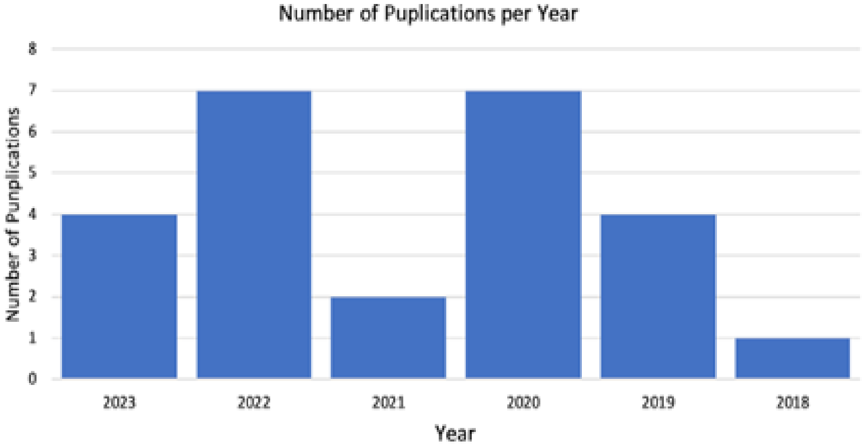

Fig.3. The number of publications per year

It also employs a metric controlling the cost of the majority class and the value of the minority class during the classification process.

The proposed algorithm uses a decision function generated by the binary relevance and chain classifier to classify multi-label imbalanced data. Ji et al. [18] proposed a framework called (ML-INC) addressing different issues in multi-label learning, including class imbalance, incomplete labels, irrelevant features, and noisy data.

To this end, ML-INC leverages low-rank and sparse decomposition and high-order label correlation, consistency, and regularization. Algorithm Adaptation (AA) is another MLC approach extending traditional classification algorithms to cope with the multi-label issue.

Roseberry et al. [19] presented a punitive k nearest neighbours algorithm with a self-adjusting memory (MLSAMPkNN) designed to deal with multi-label data streams with drift. MLSAMPkNN employs majority vote kNN on a self-adjusting sliding window containing the current concept.

It adapts the changes within the data stream efficiently using a punitive system that removes the errant data instances early from the window. The proposed algorithm has been evaluated, proving its versatility for diverse learning scenarios.

Like MLSAMPkNN, Roseberry et al. [20] have attracted attention to the issue of drifting data streams in MLC. They highlighted the need to predict the complete set of labels for multi-label instances. The authors offered a self-adjusting k nearest neighbours algorithm (MLSAkNN) to cope with these tasks and solve the mentioned issue. In contrast to previous algorithms used for such streams and instances, their algorithm uses a collectively working variety of mechanisms to deal with the issue of multi-label data.

Their algorithm works with the value k, which is self-adapting for each label and can successfully deal with multi-label instances. To address the problem of imbalanced labels, Rastogi and Mortaza [21] have suggested a multi-label classifier called Linear Imbalance Multi-label data learning with Label Specific Features (IMLSF).

IMLSF uses label correlation to increase accuracy and the structural information of the data through data locality to guarantee that similar data instances have similar class labels. To overcome class imbalance, IMLSF additionally uses a weighting mechanism that assigns varying weights to positive and negative examples based on the distribution of classes.

Sadhukhan and Palit proposed the Lattice and Imbalance Informed Multi-Label Learning (LIIML) technique [22]. The process consists of two primary stages: choosing features and managing class imbalance. First, by obtaining each label's intrinsic positive and negative class lattices, LIIML retrieves features for each label.

Second, it adopts a misclassification cost method to tackle class imbalance. LIIML is considered a hybrid method, integrating both approaches, problem transformation in the first step and algorithm adaptation in the second. The last approach is the Ensemble of Multi-Label Classification (EMLC), involving several classifiers for better performance.

Sun et al. [23] have developed an efficient ensemble paradigm named Multi-Label Ensemble with Adaptive Windowing (MLAW). Their algorithm employs Jensen–Shannon divergence to detect different kinds of concept drift. Another advantage of this method is that it prunes away infrequent label combinations to improve classification performance.

Moreover, the ensemble can also be used with the algorithm adaptation approach. MLSAkNN is also a base classifier in the Adaptive Ensemble of Self-Adjusting Nearest Neighbor Subspaces (AESAKNNS) algorithm proposed in [24].

It uses Poisson distribution to distribute a unique subset of features and samples for training to each classifier within the ensemble. The ADWIN detector also monitors classifiers. Li et al. [25] have introduced an effective classification of multi-label data streams with high-dimensional attributes and concept drifts.

The proposed algorithm utilizes an ensemble of classifiers and includes a feature selection and concept drift detection based on the max-relevance and min-redundancy approach. Law and Ghosh [26] have proposed a Multi-Label Binary Tree of Classifiers (ML-BTC) that maintains the label dependencies and class imbalance.

ML-BTC aims to construct a tree of classifiers for multi-label classification with a novel label-space partitioning method. During the training phase, the tree grows, and the decision at any node depends on the multi-label entropy and sample cardinality. Also, the unnecessary branching of imbalanced classes is restricted.

The final label set is assigned using the specific classifiers at the tree leaf nodes. Wu et al. [27] have proposed a weighted ensemble for multi-label classification called the Weighted Ensemble classification algorithm based on the Nearest Neighbors for Multi-Label data stream (WENNML).

An Active Candidate Ensemble Classifier is trained using a data stream block to generate the classifier and monitored by the well-known drift detector, ADWIN. All the detected instances will be stored in a data block, where a threshold sets its size. If the number of instances exceeds the threshold, a Passive Candidate Ensemble Classifier will be trained over the data in the block to generate a classifier; otherwise, no training is conducted. Both generated classifiers are dynamically updated using geometric and weighting techniques, replacing the old classifiers with more representative classifiers.

4.2 Q2: What Multi-label Classification Methods were Proposed ForIoT Streams?

IoT is one of the main data stream sources; however, not all the methods found have been evaluated on a streaming IoT in a non-stationary environment. Flood monitoring is one of the event monitoring applications in smart cities. Real-time monitoring of gully and drainage blockage is needed to avoid the flood.

Thus, Mishra et al. [28] have proposed an image-cropping-based DCNN model. The main task of the proposed model is to classify blockage images according to their severity. It has been trained over a dataset of 1200 images, drains, and gullies collected by the authors from Google images and YouTube videos.

Anomaly detection is a popular research field; it plays a vital role in cyber security for smart city services. Anomaly detection methods were considered binary classification problems with normal and abnormal classes.

However, such methods cannot always satisfy the demands of the smart home. Xu et al. [29] have proposed an algorithm to leverage the data produced by IoT services for anomaly detection considering the concept drift phenomenon, named Improved Long Short-Term Memory (I-LSTM).

It detects anomalies for IoT services using a Recurrent Neural Network (RNN) and then classifies the anomaly's specific category, which can belong to more than one category. It also adapts the drifts to increase detection accuracy effectively.

The authors used a real communication dataset collected from the IoT environment, which contains the communication between intelligent devices in the smart home. The Internet of Things has changed how manufacturing machinery works, as several machines can communicate and transfer data. Dalle Pezze et al. [30] have proposed a method called Balance Among Tasks Optimizing Class Distribution in Memory (BAT-OCDM).

The proposed framework is based on multi-label classification for packaging equipment monitoring. BAT-OCDM has been validated on the industrial Alarm Forecasting task originating from monitoring packaging equipment, which mainly uses the alarm logs for predictive maintenance purposes. The experimental results can effectively handle the presence of class imbalance and the distribution shifts in a stream of machines.

Human activity recognition has gained increased scientific interest due to the necessity of monitoring the multi-resident daily activities concurrently acting in smart homes. To deal with the multi-resident human activity recognition problem, Jethanandani et al. [31] have used the Classifier Chain model (CC) to predict human activities in smart homes.

The model has been applied to a real dataset of human activities in two houses with multiple residents. The model has been implemented using four classifiers, including Bernoulli Naïve Bayes, Decision Tree, K-nearest Neighbors, and Logistic Regression as base classifiers.

The results of the experiments showed that the developed Classifier Chain model with a Decision Tree outperformed the others and successfully dealt with the issue of recognizing multi-resident activities in a smart home. Similarly, Li et al.[32] have adopted a multi-label Markov Logic Network classification method (Ml-MLN) to recognize different individuals. Ml-MLN can identify resident types in multi-resident family homes based on their daily activity patterns and preferences.

Using Internet of Things technology, human motion detection systems have been developed to track and monitor people suffering from mobility difficulties. Wang et al. [33] have proposed a real-time walking motion detection system based on a multi-label imbalanced classification method, mobile phone, and motion stick.

For MLC, the authors have introduced MFGBoost, a combination of focal loss and LightGBM, mainly to classify human motions. Multi-label classification and IoT have also been used to improve the tourist experience in smart cities. Cepeda-Pacheco and Domingo have introduced a multi-label classifier to suggest the places, activities, or attractions that fit the tourists' profiles in real-time.

A deep neural network is trained with tourist profiles, and the data is collected using IoT devices in a smart city, including the visited attractions, location, time, and weather. Several sites are predicted based on the number of staying days, taking into account the nearest locations and the weather forecast.

Web Application Programming Interfaces (APIs) are well-known for providing services in Internet-of-Things (IoT) ecosystems. One of the main features of Web APIs is annotation or tagging, which plays a vital role in managing a huge number of services.

Xu et al. [34] have proposed a holistic framework utilizing multi-label classification for automatically tagging mobile and edge services in IoT systems. The framework employs neural network models based on a novel multi-head self-attention mechanism.

Such a mechanism learns the hidden correlation among annotations. Similarly, in things categorization, things are annotated with semantic labels, which can be used for searching and recommendation purposes. Chen et al. [35] have developed a novel method utilizing a binary support vector machine classifier for each label. It can produce semantic labels for a given thing.

Non-intrusive load monitoring is a technique used to monitor the total energy consumption of a building. With the emergence of IoT devices, handling massive data in such systems is challenging. Nalmpantis and Vrakas [36] have proposed a new framework tackling the power disaggregation issue called multi-label NILM.

The framework uses the dimensionality reduction method and addresses the disaggregation problem by identifying many appliances. The embedded microcontroller unit (MCU) is a vital component in supporting real-time processing in IoT.

Programmers usually face many challenges in developing code for microcontroller units and searching for sample codes online. Thus, Zhou et al. [37] proposed a tag-correlated-based classifier to help programmers query desired codes using tags. It has two machine-learning channels for processing the code description and analyzing the code content.

Table 1. Versatile multi-label classification methods

Algorithm	Objectives	Address concept drift?	Address class imbalance?	Reference
WENNML (2023)	To detect drifting data during multi-label classification.	✓	X	[26]
High-Dimensional MLC (2023)	To detect drift and select optimal features for multi-label classification.	✓	X	[24]
ML-INC (2023)	Toaddressclassimbalance, incompletelabels, irrelevantfeatures, and noisy data in MLC.	X	✓	[17]
BAT-OCDM (2023)	Utilizemulti-labelclassification in thepackagingindustry.	✓	✓	[29]
AESAKNNS (2022)	To propose an ensemble MLC method for drifting data stream.	✓	X	[23]
Semantic Annotation (2022)	Togenerateannotationsthathelpdeveloperschoosethe appropriate Web APIs, facilitatingthedevelopment of services in IoTsystems.	X	X	[33]
IMLSF (2022)	Totackleclassimbalanceissues in multi-labelclassificationbyassigningweightstolabelled and unlabeledinstances.	✓	X	[20]
MFGBoost (2022)	Toclassify human motions in a real-time walkingmotiondetectionsystem.	X	✓	[32]
ML-BTC (2022)	To enable an ensemble of tree classifiers to preserve the label dependencies and handle the class imbalance in MLC.	X	✓	[25]
Tourist Attraction RS (2022)	To predict sites that fit tourists in a smart city leveraging IoT devices.	X	X	[1]
ML-KELM (2022)	To apply multi-label classification on the data stream in SIoT.	✓	X	[37]
MLSAkNN (2021)	To predict the complete set of labels from the constantly changing stream.	✓	X	[19]
A tag correlated with MLC (2021)	To assist embedded programmers in finding microcontroller unit codes.	X	X	[36]
IC-based DCNN (2020)	To monitor floods in smart cities using MLC of gullies and drainage images.	X	X	[27]
I-LSTM (2020)	To classify the anomaly category in the IoT environment.	✓	✓	[28]
CC (2020)	To predict human activities in smart homes with multi-residents.	X	X	[38]
Things categorization (2020)	To predict the labels of a given thing.	X	X	[34]
Ml-MLN (2020)	To recognize humans in multi-resident homes.	X	X	[31]
ML-NILM (2020)	To enable the NILM framework to identify many appliances.	X	X	[35]
LIIML(2020)	To address class imbalance in multi-label datasets.	X	✓	[21]
ML-CIB (2019)	To address multi-label data issues such as class imbalance, label correlation, and irrelevant features	X	✓	[15]
MLAW (2019)	To propose a classification framework that predicts the class of the new incoming instance. It also tackles concept drift.	✓	X	[22]
MLC via label correlation (2019)	To predict a set of labels of a stream instances and tackle the concept drift.	✓	X	[14]
MLSAMPkNN (2019)	To propose a versatile MLC for drifting data streams for diverse learning scenarios	✓	✓	[18]
MLC based on assessments of cost and value (2018)	To solve the problem of class imbalance in MLC.	X	✓	[16]

It also addresses the correlation between them to determine which function modules will mostly be used for a specific application.

The proposed method employs seven well-known classifiers, such as Binary Relevance, ML-KNN, and Classifier Chains. In addition, it was evaluated using an embedded code dataset built by the researchers. The idea of the Social Internet of Things (SIoT) is formed by integrating the Internet of Things with social networking platforms.

Information can be disseminated more quickly when multi-label categorization in SIoT provides multi-dimensional search terms for an object. For example, when a user uses labels or keywords to query for an object, the object content can be sent from the source to the user who is interested in it.

Nevertheless, this procedure is more difficult in a data stream environment than in a stationary setting. Working in a streaming setting, Luo et al. [38] have developed a multi-label approach based on Kernel Extreme Learning Machine (ML-KELM).

The adaptive threshold and kernel extreme learning machine form the foundation of ML-KELM. It makes use of the Sherman-Morrison-Woodbury algorithm and the Cholesky decomposition method. Additionally, ML-KELM can adjust to concept drift in the data stream.

4.3 Q3: Do Multi-label Classification Methods in the Literature Address Concept Drift and Class Imbalance?

Most surveyed studies have addressed concept drift in MLC tasks in data streams. The few that have not addressed them are all in the context of IoT. On the other hand, class imbalance has not been studied enough when dealing with MLC tasks in data streams generally and in the context of IoT. Table 1 shows the multi-label classification methods in the literature arranged by the dates from the most recent to the earliest. It also presents their objectives and whether class imbalance and concept drift were addressed

4.4 Q4: What Are The Main Multi-label Classification Datasets and Tools?

Datasets and software implementation of the MLC algorithms are required to carry out a benchmarking comparison of several methods [39]. The datasets adopted in the literature to evaluate the performance of the algorithms in multi-label streaming environments are classified into two types:

- **Real-world Datasets:** These are collected by authors and correspond to real-world applications. Examples of real-world datasets for MLC are listed below:
- **ARAS (IoT dataset):** Is an available online dataset consisting of truth labels for 27 human activities collected from real people in real houses [40]. The two houses have seven ambient sensors, with two residents in each. Residents did not follow any specific scenario during the data collection, which took two months.
- **Sensor (IoT dataset):** is a dataset collected using a TinyDB in-network query processing system [41]. Fifty-four sensors were deployed in the Intel Berkeley Research lab over 37 days. The dataset includes four attributes: humidity, temperature, light, and voltage, and one class isdivided into four labels.
- **MediaMill: C**onsists of 43907 data instances of video frames; each has 120 features and 101 class labels . The main task is to annotate video frames with the correct class labels semantically.
- **IMDB**: Is a dataset of 120,919 records of movie plot text summaries. It is employed for text classification, where each record has 1,001 attributes and 28 class labels.
- **Ohsumed:** Is a dataset of 2,417 records of peer-reviewed medical articles. It has 103 attributes and 23 class labels, each representing disease categories.

Synthetic Datasets: generators from software tools produce artificial datasets: Such datasets are useful as they provide the ground truth of the data. Unlike real datasets, drift detection could be evaluated in synthetic datasets since the drift location, type, and duration are determined.

- **Waveform Generator:** Consists of three classes and 40 numeric attributes; each class is formed by combining two of three basewaves [6]. Concept drifts can be added by exchanging the positions of the attributes, where each attribute represents a particular context.

- **Random Tree:** Generates a data stream based on a randomly generated tree [42]. It also allows for customizing the number of attributes and classes. The tree is developed by selecting the attributes as split nodes and assigning classes to them randomly. Concept drift can be added by changing the parameters, such as the average number of labels per sample or the label dependencies. Moreover, there are a lot of open-source tools or libraries for machine learning, which provide many methods, including classification, regression, clustering, and many others. These tools also allow the researchers to implement and evaluate their proposed algorithms directly.
- **Massive Online Analysis (MOA):** Is a well-known framework for data stream learning [43]. MOA is based on the WEKA library and written in Java. It provides various popular machine learning algorithms, MLC algorithms and data stream generators. MOA also allows drift simulation and provides several evaluation methods.
- **Scikit-multiflow:** Is an open-source framework for learning from data streams in Python. [44]. It extends the well-known sci-kit-learn designed to accommodate several data stream algorithms. Scikit-multiflow provides several stream generators, learners, evaluators, and popular change detectors.
- **Mulan is an Open-source library developed in Java that applies to** multi-label learning. It contains several state-of-the-art algorithms concerning multi-label learning. Mulan also provides evaluation measures by hold-out and cross-validation.
- **MEKA:** is a framework built on WEKA, a machine learning tool . It provides several multi-label learning methods.
- **Hadoop SPARK:** is a software designed to process large-scale data supporting many languages such as Python, Java, Scala, and R [45]. It involves a built-in library that includes SQL, Spark Streaming, MLlip, and GraphX. MLlip is a machine-learning library containing many algorithms for classification, regression, clustering, association rules, etc.

5 Discussion

Smart systems generate data streams in varying forms, at high speeds, and in massive volumes. Most traditional machine learning algorithms assume that each instance is associated with only a single class label, which is not true in many real-world cases. The methods under versatile multi-label classification have been evaluated regarding music/ image categorization, text categorization, bioinformatics, and many others [46].

Most studies discovered different types of data changes that occur suddenly or gradually. However, the class imbalance was not extensively inspected, although it is a well-known challenge in real-world data stream classification problems. Moreover, many methods showed a high complexity and long classification time.

The Internet of Things (IoT) is an expanding and growing field that has attracted researcher attention in the last decade. Methods explored in the literature showed the diversity of IoT applications and the importance of multi-label classification, specifically in smart cities. MLC methods can monitor floods in smart cities utilizing gully and drainage images [47]. They also can predict sites that fit tourists by leveraging IoT devices. Besides, MLC methods are used to recognize several activities of residents in smart homes [48].

However, most studies did consider drift detection and class imbalance. Drift is one of the main characteristics of the data stream, and class imbalance is very common in the IoT data stream. Therefore, MLC models leveraging IoT data stream must tackle these challenges to avoid wrong predictions.

Despite the extensive research on the MLC of data stream, the joint treatment of concept drift and class imbalance in the context of IoT or data stream is still not largely explored. Furthermore, concept evolution is another MLC challenge that could be further researched.

It occurs when completely new labels appear where they have never emerged before. Features also evolve; known features might fade, and new ones emerge. Still, existing MLC methods do not significantly recognize the evolution of the stream's class labels and features.

6 Conclusion and Future Works

Data stream Classification has many challenges, including concept drift, class imbalance, and multi-label data. In many real-world applications, data instances are associated with many labels, where single classification models are unsuitable.

Also, classification models without a drift detector won't be able to capture new data distribution; therefore,they will keep learning from prior training data, resulting in inaccurate predictions. Furthermore, classification models should be able to deal with imbalanced data to avoid bias toward majority classes and deteriorate the model's performance.

This article reviews the state-of-the-art versatile multi-label classification. It also provides the MLC methods developed for the IoT stream. However, some methods have not addressed drift detection, especially those developed for IoT.

Furthermore, most studies did not consider the data imbalance. The joint treatment for data imbalance, multi-label, and concept drift remains unexplored.Future works of IoT data streams can be integrated with edge computing, predictive analytics and data fusion techniques for effective system monitoring.

References

1. **Cepeda-Pacheco, J. C., Domingo, M. C. (2022).** Deep learning and internet of things for tourist attraction recommendations in smart cities. Neural Computing and Applications, Vol. 34, No. 10, pp. 7691–7709. DOI: 10.1007/s00521-021-06872-0.
2. **Nguyen, H., Woon, Y., Ng, W. (2014).** A survey on data stream clustering and classification. Knowledge and Information Systems, Vol. 45, No. 3, pp. 535–569. DOI: 10.1007/s10115-014-0808-1.
3. **Hurbean, L., Danaiata, D., Militaru, F., Dodea, A., Negovan, A. (2021).** Open data based machine learning applications in smart cities: a systematic literature review. Electronics, Vol. 10, No. 23, pp. 2997. DOI: 10.3390/electronics10232997.
4. **Rivera, G., Florencia, R., García, V., Ruiz, A., Sánchez-Solís, J. P. (2020).** News classification for identifying traffic incident points in a spanish-speaking country: a real-world case study of class imbalance learning. Applied Sciences, Vol. 10, No. 18, pp. 6253. DOI: 10.3390/app10186253.
5. **Zheng, X., Li, P., Chu, Z., Hu, X. (2020).** A survey on multi-label data stream classification. IEEE Access, Vol. 8, pp. 1249–1275. DOI: 10.1109/access.2019.2962059.
6. **Endut, N., Hamzah, W. M. A. F., Ismail, I., Kamir-Yusof, M., Abu-Baker, Y., Yusoff, H. (2022).** A systematic literature review on multi-label classification based on machine learning algorithms. TEM Journal, pp. 658–666. DOI: 10.18421/tem112-20.
7. **Herrera, F., Charte, F., Rivera, A. J., del-Jesus, M. J. (2016).** Multilabel classification. Springer International Publishing. DOI: 10.1007/978-3-319-41111-8.
8. **Sanghi, G., Kanungo, N., Deshmukh, S., Agarwal, S. (2017).** Automatic multi-label image annotation for smart cities. IEEE Region 10 Symposium, Vol. 14, pp. 1–4. DOI: 10.1109/tenconspring.2017.8069997.
9. **Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., Bouchachia, A. (2014).** A survey on concept drift adaptation. ACM Computing Surveys, Vol. 46, No. 4, pp. 1–37. DOI: 10.1145/2523813.
10. **Wang, P., Jin, N., Fehringer, G. (2020).** Concept drift detection with false positive rate for multi-label classification in IoT data stream. International Conference on UK-China Emerging Technologies, Vol. 11, pp. 1–4. DOI: 10.1109/ucet51115.2020.9205421.
11. **García, V., Sánchez, J., Marqués, A., Florencia, R., Rivera, G. (2020).** Understanding the apparent superiority of over-sampling through an analysis of local information for class-imbalanced data. Expert Systems with Applications, Vol. 158, pp. 113026. DOI: 10.1016/j.eswa.2019.113026.

12. **Bolívar, A., García, V., Florencia, R., Alejo, R., Rivera, G., Sánchez-Solís, J. P. (2022).** A preliminary study of smote on imbalanced big datasets when dealing with sparse and dense high dimensionality. Pattern Recognition, Vol. 13264, pp. 46–55. DOI: 10.1007/978-3-031-07750-0_5.

13. **Wolfswinkel, J. F., Furtmueller, E., Wilderom, C. P. M. (2013).** Using grounded theory as a method for rigorously reviewing literature. European Journal of Information Systems, Vol. 22, No. 1, pp. 45–55. DOI: 10.10 57/ejis.2011.51.

14. **Nguyen, T. T., Nguyen, T. T. T., Luong, A. V., Nguyen, Q. V. H., Liew, A. W., Stantic, B. (2019).** Multi-label classification via label correlation and first order feature dependance in a data stream. Pattern Recognition, Vol. 90, pp. 35–51. DOI: 10.1016/j.patcog.2019. 01.007.

15. **Braytee, A., Liu, W., Anaissi, A., Kennedy, P. J. (2019).** Correlated multi-label classification with incomplete label space and class imbalance. ACM Transactions on Intelligent Systems and Technology, Vol. 10, No. 5, pp. 1–26. DOI: 10.1145/3342512.

16. **Ding, M., Yang, Y., Lan, Z. (2018).** Multi-label imbalanced classification based on assessments of cost and value. Applied Intelligence, Vol. 48, No. 10, pp. 3577–3590. DOI: 10.1007/s10489-018-1156-8.

17. **Ji, X., Tan, A., Wu, W., Gu, S. (2023).** Multi-label classification with weak labels by learning label correlation and label regularization. Applied Intelligence, Vol. 53, No. 17, pp. 20110–20133. DOI: 10.1007/s10489-023-04562-z.

18. **Roseberry, M., Krawczyk, B., Cano, A. (2019).** Multi-label punitive kNN with self-adjusting memory for drifting data streams. ACM Transactions on Knowledge Discovery from Data, Vol. 13, No. 6, pp. 1–31. DOI: 10.11 45/3363573.

19. **Roseberry, M., Krawczyk, B., Djenouri, Y., Cano, A. (2021).** Self-adjusting k nearest neighbors for continual learning from multi-label drifting data streams. Neurocomputing, Vol. 442, pp. 10–25. DOI: 10.1016/j.neucom. 2021.02.032.

20. **Rastogi, R., Mortaza, S. (2022).** Imbalance multi-label data learning with label specific features. Neurocomputing, Vol. 513, pp. 395–408. DOI: 10.1016/j.neucom.2022.09.085.

21. **Sadhukhan, P., Palit, S. (2020).** Lattice and imbalance informed multi-label learning. IEEE Access, Vol. 8, pp. 7394–7407. DOI: 10.1109/ access.2019.2962201.

22. **Sun, Y., Shao, H., Wang, S. (2019).** Efficient ensemble classification for multi-label data streams with concept drift. Information, Vol. 10, No. 5, pp. 158. DOI: 10.3390/info10050158.

23. **Alberghini, G., Barbon-Junior, S., Cano, A. (2022).** Adaptive ensemble of self-adjusting nearest neighbor subspaces for multi-label drifting data streams. Neurocomputing, Vol. 481, pp. 228–248. DOI: 10.1016/j.neucom. 2022.01.075.

24. **Li, P., Zhang, H., Hu, X., Wu, X. (2022).** High-dimensional multi-label data stream classification with concept drifting detection. IEEE Transactions on Knowledge and Data Engineering, Vol. 35, No. 8, pp. 8085–8099. DOI: 10.1109/tkde.2022.3200068.

25. **Law, A., Ghosh, A. (2022).** Multi-label classification using binary tree of classifiers. IEEE Transactions on Emerging Topics in Computational Intelligence, Vol. 6, No. 3, pp. 677–689. DOI: 10.1109/tetci.2021.3 075717.

26. **Wu, H., Han, M., Chen, Z., Li, M., Zhang, X. (2023).** A weighted ensemble classification algorithm based on nearest neighbors for multi-label data stream. ACM Transactions on Knowledge Discovery from Data, Vol. 17, No. 5, pp. 1–21. DOI: 10.1145/3570960.

27. **Mishra, B. K., Thakker, D., Mazumdar, S., Neagu, D., Gheorghe, M., Simpson, S. (2020).** A novel application of deep learning with image cropping: a smart city use case for flood monitoring. Journal of Reliable Intelligent Environments, Vol. 6, No. 1, pp. 51–61. DOI: 10.1007/s40860-020-00099-x.

28. **Xu, R., Cheng, Y., Liu, Z., Xie, Y., Yang, Y. (2020).** Improved long short-term memory based anomaly detection with concept drift adaptive method for supporting IoT services. Future Generation Computer Systems, Vol.

112, pp. 228–242. DOI: 10.1016/j.future.2020.05.035.

29. **Dalle-Pezze, D., Deronjic, D., Masiero, C., Tosato, D., Beghi, A., Susto, G. A. (2023).** A multi-label continual learning framework to scale deep learning approaches for packaging equipment monitoring. Engineering Applications of Artificial Intelligence, Vol. 124, pp. 106610. DOI: 10.1016/j.engappai.2023.106610.

30. **Jethanandani, M., Sharma, A., Perumal, T., Chang, J. (2020).** Multi-label classification based ensemble learning for human activity recognition in smart home. Internet of Things, Vol. 12, pp. 100324. DOI: 10.1016/j.iot.2020.100324.

31. **Li, Q., Huangfu, W., Farha, F., Zhu, T., Yang, S., Chen, L., Ning, H. (2020).** Multi-resident type recognition based on ambient sensors activity. Future Generation Computer Systems, Vol. 112, pp. 108–115. DOI: 10.1016/j.future.2020.04.039.

32. **Wang, J., Jiang, X., Meng, Q., Saada, M., Cai, H. (2022).** Walking motion real-time detection method based on walking stick, IoT, copod and improved lightgbm. Applied Intelligence, Vol. 52, No. 14, pp. 16398–16416. DOI: 10.1007/s10489-022-03264-2.

33. **Xu, Y., Xiao, W., Yang, X., Li, R., Yin, Y., Jiang, Z. (2023).** Towards effective semantic annotation for mobile and edge services for internet-of-things ecosystems. Future Generation Computer Systems, Vol. 139, pp. 64–73. DOI: 10.1016/j.future.2022.09.021.

34. **Chen, Y., Zhang, J., Xu, L., Guo, M., Cao, J. (2020).** Modeling latent relation to boost things categorization service. IEEE Transactions on Services Computing, Vol. 13, No. 5, pp. 915–929. DOI: 10.1109/tsc.2017.2715159.

35. **Nalmpantis, C., Vrakas, D. (2020).** On time series representations for multi-label NILM. Neural Computing and Applications, Vol. 32, No. 23, pp. 17275–17290. DOI: 10.1007/s00521-020-04916-5.

36. **Zhou, Y., Cui, S., Wang, Y. (2021).** Machine learning based embedded code multi-label classification. IEEE Access, Vol. 9, pp. 150187–150200. DOI: 10.1109/access.2021.3123498.

37. **Luo, F., Liu, G., Guo, W., Chen, G., Xiong, N. (2022).** ML-KELM: A kernel extreme learning machine scheme for multi-label classification of real time data stream in SIoT. IEEE Transactions on Network Science and Engineering, Vol. 9, No. 3, pp. 1044–1055. DOI: 10.1109/tnse.2021.3073431.

38. **Kasubi, J. W., Huchaiah, M. D. (2021).** Human activity recognition for multi-label classification in smart homes using ensemble methods. Artificial Intelligence and Sustainable Computing for Smart City, Communications in Computer and Information Science, Vol. 124, pp. 282–294. DOI: 10.1007/978-3-030-82322-1_21.

39. **Madden, S. (2022).** Intel Lab Data. db.csail.mit.edu/labdata/labdata.html

40. **Museba, T., Nelwamondo, F., Ouahada, K. (2021).** An adaptive heterogeneous online learning ensemble classifier for nonstationary environments. Computational Intelligence and Neuroscience, Vol. 2021, No. 1. DOI: 10.1155/2021/6669706.

41. **Bifet, A., Read, J., Holmes, G., Pfahringer, B. (2018).** Streaming data mining with massive online analytics (MOA). Series in Machine Perception and Artificial Intelligence, Data Mining in Time Series and Streaming Databases, pp. 1–25. DOI: 10.1142/9789813228047_0001.

42. **Montiel, J., Read, J., Bifet, A., Abdessalem, T. (2018).** Scikit-multiflow: A multi-output streaming framework. Journal of Machine Learning Research, Vol. 19, No. 72, pp. 1–5.

43. **Apache SparkTM (2022).** Unified engine for large-scale data analytics. spark.apache.org/

44. **Ganji, R. N., Dadkhah, C., Tohidi, N. (2023).** Improving sentiment classification for hotel recommender system through deep learning and data balancing. Computación y Sistemas, Vol. 27, No. 3, pp. 811–825. DOI: 10.13053/cys-27-3-4655.

45. **Moreno-Escobar, J. J., Pérez-Franco, V. J., Coria-Páez, A. L., Morales-Matamoros, O., Aguilar-del-Villar, E. Y., Castillo-Perez, M. D. (2023).** Multivariate data analysis of

consumer behavior of functional products: a neuroscience and neuromarketing approach to improve decision-making. Computación y Sistemas, Vol. 27, No. 4, pp. 1027–1046. DOI: 10.13053/cys-27-4-4690.

46. **Gelbukh, A., Pérez-Alvarez, D. A., Kolesnikova, O., Chanona-Hernandez, L., Sidorov, G. (2024).** Multi-instrument based n-grams for composer classification task. Computación y Sistemas, Vol. 28, No. 1, pp. 85–98. DOI: 10.13053/cys-28-1-4903.

Article received on 28/02/2024; accepted on 15/05/2024.
**Corresponding author is Mashail Althabiti.*