Pattern Recognition 135 (2023) 109190

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Multilabel Prototype Generation for data reduction in K-Nearest Neighbour classification

Jose J. Valero-Mas[a,*], Antonio Javier Gallego[a], Pablo Alonso-Jiménez[b], Xavier Serra[b]

[a] University Institute for Computer Research, University of Alicante, Spain
[b] Music Technology Group, Universitat Pompeu Fabra, Spain

ARTICLE INFO

Article history:
Received 19 April 2022
Revised 11 October 2022
Accepted 13 November 2022
Available online 15 November 2022

Keywords:
Multilabel classification
Prototype generation
Efficient *k*NN

ABSTRACT

Prototype Generation (PG) methods are typically considered for improving the efficiency of the *k*-Nearest Neighbour (*k*NN) classifier when tackling high-size corpora. Such approaches aim at generating a reduced version of the corpus without decreasing the classification performance when compared to the initial set. Despite their large application in multiclass scenarios, very few works have addressed the proposal of PG methods for the multilabel space. In this regard, this work presents the novel adaptation of four multiclass PG strategies to the multilabel case. These proposals are evaluated with three multilabel *k*NN-based classifiers, 12 corpora comprising a varied range of domains and corpus sizes, and different noise scenarios artificially induced in the data. The results obtained show that the proposed adaptations are capable of significantly improving—both in terms of efficiency and classification performance—the only reference multilabel PG work in the literature as well as the case in which no PG method is applied, also presenting statistically superior robustness in noisy scenarios. Moreover, these novel PG strategies allow prioritising either the efficiency or efficacy criteria through its configuration depending on the target scenario, hence covering a wide area in the solution space not previously filled by other works.

1. Introduction

The *k*-Nearest Neighbour (*k*NN) classifier represents one of the most well-known algorithms for non-parametric supervised classification, mostly due to its conceptual simplicity and good statistical error properties [1]. For a given query, this method hypothesises about its category by querying the *k* nearest neighbours of a reference corpus, following a specified similarity measure [2]. In this regard, this classification strategy has been largely considered in a wide range of disparate fields as, for instance, diabetes detection [3], musical key estimation [4] or handwritten signature verification [5], among others.

However, as a representative case of the *lazy* learning paradigm, *k*NN does not derive a model out of the reference corpus [6]. In contrast, for every query, this method requires an exhaustive search among the elements of the aforementioned corpus, thus entailing low-efficiency figures in both classification time and memory usage [7]. Note that, while this inefficiency issue may be obviated in scenarios with limited amounts of data, when considering large data collections, *k*NN becomes intractable [8].

Data Reduction (DR) stands as one of the most popular approaches in the related literature for tackling this drawback [9]. This group of methods aims to reduce the size of the reference set for improving the efficiency of the model while keeping—or even increasing—the classification performance obtained with the original data. Among them, the Prototype Generation (PG) family represents one of the most competitive alternatives due to its remarkable reduction capabilities compared to other DR strategies [10]. In a broad sense, PG derives an alternative reference set for the classifier by performing different selection and merging operations on the elements of the initial corpus following certain heuristics [11].

Due to the relevance of PG for efficient *k*NN-based classification, a considerable amount of research effort has been invested in proposing novel strategies as well as improving the existing ones [12]. However, such research works have typically addressed *multiclass* scenarios—classification tasks in which every single query is assigned to one category out of a set of mutually excluding labels—, hence neglecting the more general *multilabel* scenario—case in which an undetermined number of categories is assigned to each query [13].

The work by Ougiaroglou et al. [14] represents one of the scarce works of a PG strategy devised to address multilabel scenarios. More precisely, this work proposes the adaptation of the

[*] Corresponding author.
E-mail address: jjvalero@dlsi.ua.es (J.J. Valero-Mas).

https://doi.org/10.1016/j.patcog.2022.109190

state-of-the-art Reduction through Homogeneous Clustering (RHC) method [15] to the multilabel space, obtaining the so-called Multilabel Reduction through Homogeneous Clustering (MRHC). The authors not only conclude that such adaptation remarkably improves the efficiency of the *k*NN classifier in multilabel scenarios but also state the need for contriving multilabel PG strategies due to the shortage of existing alternatives.

In this context, the present work further explores the proposal and use of PG methods for improving the efficiency of *k*NN classification in multilabel scenarios. More precisely, we introduce the novel adaptation of four PG strategies from their original multiclass formulation to the multilabel case. These proposals have been comprehensively evaluated considering several multilabel classification approaches based on *k*NN with a wide variety of corpora. Additionally, different percentages of label-level noise particularly devised for this multilabel framework—artificial alterations of the classes or labels of the data—have been induced in the corpora to assess the robustness of the proposals and their capability of dealing with such adverse scenarios. The results obtained report a statistically significant improvement in terms of both reduction capabilities and classification performance for all scenarios and noise levels contemplated compared to the exhaustive search carried out by the base *k*NN method and the reference MRHC reduction approach. In this regard, these novel proposals not only fill a gap in the scarce multilabel PG literature but also reportedly outperform the only existing strategy in the field, the commented MRHC algorithm.

The rest of the work is structured as follows: Section 2 provides the theoretical background of the work; Section 3 presents the proposed PG methods; Section 4 introduces the experimental setup; Section 5 shows and discusses the results; and finally, Section 6 concludes the work and poses future research lines to pursue.

2. Background

To adequately describe multilabel classification, we initially introduce the multiclass framework, as it conceptually represents a simpler task. Formally, let $\mathcal{X} \in \mathbb{R}^f$ denote an f-dimensional feature space and $\mathcal{Y}_{mc}$ a set of discrete labels. Additionally, let $\mathcal{T}_{mc} = \{(\boldsymbol{x}_i, y_i) : \boldsymbol{x}_i \in \mathcal{X}, y_i \in \mathcal{Y}_{mc}\}_{i=1}^{|\mathcal{T}_{mc}|}$ represent an annotated collection of data where each datum $\boldsymbol{x}_i \in \mathcal{X}$ is related to label $y_i \in \mathcal{Y}_{mc}$ by an underlying function $h_{mc} : \mathcal{X} \rightarrow \mathcal{Y}_{mc}$. The goal of multiclass classification is retrieving the most accurate approximation $\hat{h}_{mc}(\cdot)$ to that underlying function.

Among the different alternatives for performing such an approximation task, the well-known *k*NN stands as one of the most common choices given its relevance in the Pattern Recognition field [16]. Formally, given a query $q \in \mathcal{X}$, this method models $\hat{h}_{mc}$ as:

$$\hat{h}_{mc}(q) = \text{mode}\left(\mathcal{Y}_{mc}^{k}\left(\underset{\boldsymbol{x}_i \in \mathcal{T}_{mc}}{\arg\min_k}\{d(q, \boldsymbol{x}_i)\}\right)\right) \qquad (1)$$

where k stands for the number of neighbours considered, $d : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}_0^+$ is a dissimilarity measure, $\text{mode} : \mathcal{Y}_{mc} \rightarrow \mathcal{Y}_{mc}$ denotes the mode operator, and $\mathcal{Y}_{mc}^k$ is the set of labels retrieved from the closest k elements to the query q.

As previously introduced, the multilabel paradigm constitutes a generalisation of the multiclass framework in which each individual instance may be associated with more than a single label [17]. Formally, the set of multilabel data $\mathcal{T}_{ml} = \{(\boldsymbol{x}_i, \dagger_i) : \boldsymbol{x}_i \in \mathcal{X}, \dagger_i \subseteq \mathcal{Y}_{ml}\}_{i=1}^{|\mathcal{T}_{ml}|}$ relates datum $\boldsymbol{x}_i \in \mathcal{X}$ to a subset of classes $\dagger_i \subseteq \mathcal{Y}_{ml}$, namely labelset, where $\mathcal{Y}_{ml} = \{\lambda_1, \lambda_2, \ldots, \lambda_L\}$ is an L-size collection of mutually non-exclusive labels [18]. As in the multiclass case, the goal is retrieving the most accurate approximation $\hat{h}_{ml}(\cdot)$ to the underlying function $h_{ml} : \mathcal{X} \rightarrow \mathcal{Y}_{ml}$.

To leverage the advantages of multiclass classifiers in multilabel scenarios, the literature considers two main approaches [19]: *problem transformation* and *algorithm adaptation*. We now describe these paradigms and report some commonly considered methods within them for *k*NN schemes as it represents the focus of the work.

The *problem transformation* paradigm disentangles the multilabel task into several single-label problems for then applying a multiclass *k*NN-based strategy for performing the classification task. Some of the most common alternatives are: the *Binary Relevance k*NN (BR*k*NN), which decomposes the task into L independent binary classification problems [20]; the *Label Powerset k*NN (LP-*k*NN), which derives an alternative single-label corpus where each labelset is considered as a different class [21]; and Random *k*-Labelsets (RA*k*EL), which divides the initial set of labels into a number of small random subsets for then performing LP-*k*NN and creating an ensemble-based classifier [22].

In contrast, the *algorithm adaptation* approach focuses on modifying the base multiclass classifier to fit the multilabel scenario. In this regard, the *Multilabel k*NN (ML-*k*NN) proposed by Zhang and Zhou [23] expands the base *k*NN method resorting to a maximum-a-posteriori principle to determine the labelset of the query based on its neighbouring instances. Some extensions to this approach are the *Dependent ML-kNN* [24], which models the different dependencies among the set of labels, the IBLR-ML method [25], which expands the base ML-*k*NN one by combining it with logistic regression, or the combination of ensembles and ML-*k*NN as in the work by Zhu et al. [26].

Nevertheless, while these transformations and adaptations allow the use of *k*NN in multilabel classification tasks, the inherent efficiency issue of these classifiers has been neglected in the literature. Note that, while some multilabel schemes such as the ML-*k*NN depict similar inefficiency figures to that of the multiclass *k*NN formulation since they explore the entire reference $\mathcal{T}_{ml}$ set, the BR*k*NN case is of particular relevance as it requires iterating through the $\mathcal{T}_{ml}$ set L different times.

The Prototype Generation (PG) family of methods stands as one of the most successful approaches for efficient *k*NN classification in multiclass cases [9]. As a representative case of DR strategy, PG aims to obtain an alternative set $\mathcal{R}_{mc}$ by performing certain combinations and transformations on the elements of $\mathcal{T}_{mc}$ so that $|\mathcal{R}_{mc}| < |\mathcal{T}_{mc}|$ while keeping—or even improving—the classification performance. However, as aforementioned, to the best of our knowledge, there is a remarkable lack of methods for performing PG in multilabel scenarios. The sole exception to this assertion is the work by Ougiaroglou et al. [14] where the state-of-the-art multiclass PG method RHC was adapted to the multilabel space. In that work, the authors experimentally proved the usefulness of their PG proposal to improve the efficiency of the multilabel classification and stated the need for devising other alternatives to fill this existing gap in the literature.

In this context, the present work proposes a novel adaptation to the multilabel space of four well-known multiclass PG algorithms. More precisely, we consider the classic Chen reduction algorithm [27] as well as the three different versions of the reference Reduction through Space Partitioning (RSP) strategy by Sánchez [28]. For this first-time adaptation to the multilabel space of such PG algorithms, this work proposes several mechanisms for both partitioning and integrating the labels of the multilabel prototypes of the initial corpus for eventually generating the instances of the reduced multilabel set. These novel methods are thoroughly compared, in terms of both performance and efficiency, to the state-of-the-art proposal by Ougiaroglou et al. [14] and to the case in which no reduction is performed considering different multilabel *k*NN-based classifiers, corpora, and noise scenarios. Such a study shall provide insights on whether the proposed multilabel

PG methods cope with the commented efficiency issue without decreasing the classification performance and on their robustness as well as data cleansing capabilities in cases depicting the presence of noise in the data.

3. Prototype generation in the multilabel space

This section presents the proposed PG methods for the multilabel space. As commented, we focus on the first-time adaptation of four algorithms originally devised for multiclass cases: the Chen method [27] and the three versions of the Reduction through Space Partitioning (RSP) strategy [28]. In this regard, the first part of the section introduces the original multiclass formulations of these algorithms and the second one presents their respective multilabel adaptations proposed in this work.

3.1. Reference multiclass PG

The considered multiclass PG strategies—the Chen method as well as the different RSP versions—constitute representative examples of the so-called *space splitting* policy [29], which typically follows a two-step approach: a first stage, *space partitioning*, divides the feature space of the multiclass set $\mathcal{T}_{mc}$ into different regions using certain heuristics; after that, the *prototype merging* stage computes new prototypes from each region attending to different criteria, producing the reduced set $\mathcal{R}_{mc}$. The existing PG strategies under this framework, therefore, essentially differ in the particular splitting and prototype generation heuristics considered.

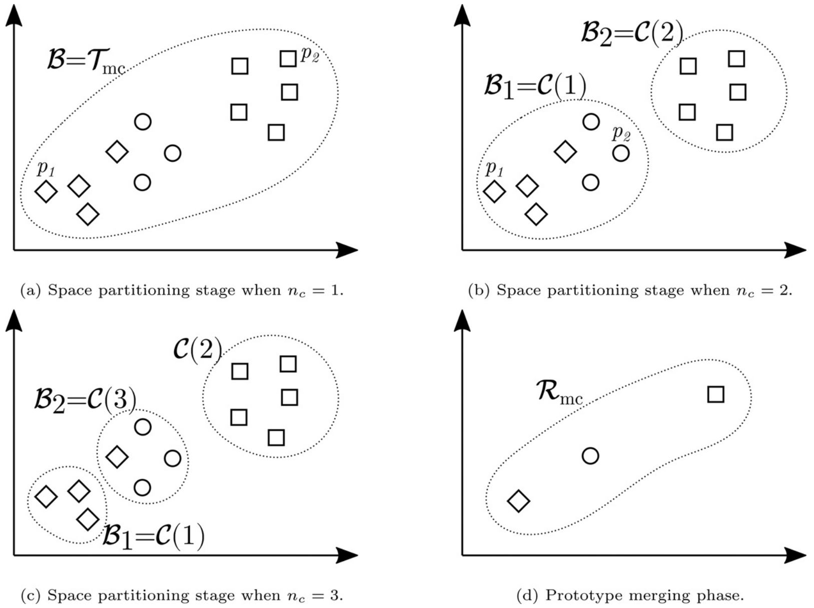
In the specific case of the Chen and RSP PG families, the aforementioned heuristics depict some similarities. In this regard, we first present the particular approach followed by the Chen method in Algorithm 1, aided by the graphical illustration in Fig. 1, for then commenting on the different points on which the three RSP strategies differ from it.

As it may be observed in the algorithm, the method iteratively divides the feature space of $\mathcal{T}_{mc}$ into n_d—user parameter—disjoint subsets which are denoted as $\mathcal{C}_{mc}(i)$ where $\bigcup_{i=1}^{n_d} \mathcal{C}_{mc}(i) = \mathcal{T}_{mc}$. For that, the largest subset in each iteration is divided in two attending to the distance between the two farthest prototypes in it. Eventually, for each of the n_d regions, a new prototype is obtained as the median of the features of the elements in it and labelled after the most common class. Hence, the size of the resulting reduced set equals the number of partitions selected by the user, *i.e.*, $|\mathcal{R}_{mc}| = n_d$.

Algorithm 1: Chen algorithm for multiclass PG [27].

```
Input  : $\mathcal{T}_{mc} \subset \mathcal{X} \times \mathcal{Y}_{mc} \leftarrow$ Multiclass corpus
         $n_d \leftarrow$ Number of resulting partitions
         $d(\cdot,\cdot) \leftarrow$ Dissimilarity measure
Output: $\mathcal{R}_{mc} \leftarrow$ Reduced set
Let $n_c = i = 1$, $\mathcal{C}_{mc} = \emptyset$, $\mathcal{B} = \mathcal{T}_{mc}$          ▷ Space partitioning
Let $p_1, p_2$ be the farthest prototypes in $\mathcal{B}$
while $n_c < n_d$ do
    Divide $\mathcal{B}$ into subsets:
        $\mathcal{B}_1 = \{p \in \mathcal{B} : d(p, p_1) \leq d(p, p_2)\}$
        $\mathcal{B}_2 = \{p \in \mathcal{B} : d(p, p_1) > d(p, p_2)\}$
    Set $n_c = n_c + 1$, $\mathcal{C}_{mc}(i) = \mathcal{B}_1$, and $\mathcal{C}(n_c) = \mathcal{B}_2$
    Divide $\mathcal{C}_{mc}$ into subsets:
        $\mathcal{I}_1 = \{i : |\{y \in C_{mc}(i)\}| > 1\}$
        $\mathcal{I}_2 = \{j : j \leq n_c\} - \mathcal{I}_1$
    Let $\mathcal{I} = \mathcal{I}_1$ if $\mathcal{I}_1 \neq \emptyset$ else $\mathcal{I}_2$
    Find farthest points $q_1(i), q_2(i)$ in $\mathcal{C}_{mc}(i)$ $\forall\ i \in \mathcal{I}$
    Let $j = argmax_{j\in[1,i]} d(q_1(j), q_2(j))$
    Set $\mathcal{B} = \mathcal{C}_{mc}(j)$, $p_1 = q_1(j)$, and $p_2 = q_2(j)$
end while
Compute $\mathcal{R}_{mc} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n_d}$ as:          ▷ Prototype merging
    $\boldsymbol{x}_i = \text{median}(\{\boldsymbol{x} \in \mathcal{C}_{mc}(i)\})$
    $y_i = \text{mode}(\{y \in \mathcal{C}_{mc}(i)\})$
```

The RSP family, as commented, builds upon Chen's proposal by modifying some of the space partitioning and/or prototype merging stages. The first RSP version—RSP1—considers the Chen algorithm prone to discard underrepresented classes due to its pro-


$\mathcal{B}$=$\mathcal{T}_{mc}$
p_1
p_2
$\mathcal{B}_2$=$\mathcal{C}(2)$
$\mathcal{B}_1$=$\mathcal{C}(1)$
p_1
p_2
$\mathcal{C}(2)$
$\mathcal{B}_2$=$\mathcal{C}(3)$
$\mathcal{B}_1$=$\mathcal{C}(1)$
$\mathcal{R}_{mc}$

(a) Space partitioning stage when $n_c = 1$.

(b) Space partitioning stage when $n_c = 2$.

(c) Space partitioning stage when $n_c = 3$.

(d) Prototype merging phase.

Fig. 1. Graphical illustration of the multiclass Chen PG method. The example depicts the results of the space partitioning process (cases 1 a to 1 c) and the prototype merging phase (case 1 d) when considering $n_d = 3$ subsets. Symbols p_1 and p_2 denote the two furthest prototypes in the cluster to be divided.

totype merging policy (lines 16–18 in Algorithm 1). Thus, instead of computing a single prototype for each of the n_d regions and labelling them after the most represented class in each partition, RSP1 only merges prototypes sharing the same label. Hence, each region is now represented by as many prototypes as the number of classes it contains. In this case, therefore, the size of the reduced set may not be known in advance but accomplishes $|\mathcal{R}_{mc}| \geq n_d$.

The second version of RSP—RSP2—expands RSP1 by modifying the criterion for selecting the region to split (lines 12–13 in Algorithm 1). RSP2 considers the overlapping degree criterion, which is defined as the ratio of the average distance between instances belonging to different classes and the average distance between instances that are from the same class. The region with the largest overlapping degree is the one to be divided.

The third RSP reduction heuristic—RSP3—is based on the idea that each resulting region should represent a cluster of instances belonging to only one class. Thus, this approach modifies the Chen method so that it iteratively performs the space partitioning stage (line 3 in Algorithm 1) until all resulting sets are homogeneous in terms of class representation, remaining the prototype merging phase of the algorithm unaltered. Hence, unlike the RSP1 and RSP2 strategies, the RSP3 approach does not require the n_d parameter related to the number of resulting regions since the method exclusively relies on this class homogeneity criterion to accomplish the space partitioning stage.

3.2. Multilabel PG proposals

Having introduced the four reference PG methods in their multiclass formulation, we now present their respective proposed multilabel adaptations.

The multilabel space splitting PG framework may be formulated in an analogous manner to that of the multiclass case. Initially, the *space partitioning* phase divides the multilabel set $\mathcal{T}_{ml} \subset \mathcal{X} \times \mathcal{Y}_{ml}$ into n_d non-overlapping multilabel regions C_{ml} such that $\bigcup_{i=1}^{n_d} \mathcal{C}_{ml}(i) = \mathcal{T}_{ml}$. After the convergence of this stage, the *prototype merging* step retrieves the multilabel set of data $\mathcal{R}_{ml}$ generated out of these C_{ml} clusters by following a certain approach, where $|\mathcal{R}_{ml}| \leq |\mathcal{T}_{ml}|$. Within this framework, we introduce the different modifications proposed for accommodating the presented multiclass PG methods to such a scenario.

Our first proposal is the adaptation of the Chen algorithm, namely *Multilabel Chen* or *MChen*. Since the space partitioning stage (lines 1–15) computes the set of clusters $\mathcal{C}_{mc}$ only relying on the set of features $\mathcal{X}$, no adaptation is required for its multilabel formulation to obtain set $\mathcal{C}_{ml}$. Oppositely, given that the prototype merging stage (lines 16–18) usually requires combining elements from different classes, the question arises about the proper approach to do so in multilabel spaces since the simple selection of the most common label in the $\mathcal{C}_{ml}$ cluster is not suitable for the considered scenario.

In this regard, we resort to the policy devised by Ougiaroglou et al. [14] for the MRHC method in which the resulting prototype keeps the labels present in at least half of the instances of the cluster. Mathematically, the labelset assigned to the resulting element in cluster $\mathcal{C}_{ml}(i)$ is given by:

$$\dagger_i = \left\{ \lambda : |\mathcal{C}_{ml}(i)|_\lambda \geq \frac{|\mathcal{C}_{ml}(i)|}{2} \quad \forall\ \lambda \in \mathcal{C}_{ml}(i) \right\} \tag{2}$$

where $|\mathcal{C}_{ml}(i)|_\lambda$ denotes the cardinality of label λ in subset $\mathcal{C}_{ml}(i)$. This expression replaces that in line 18 of Algorithm 1 whereas the policy followed for obtaining the set of features (line 17) is not modified. Fig. 2c provides a graphical example of this merging procedure considering the space partitioning result shown in Fig. 2 b.

The second proposal is the *Multilabel RSP1* or *MRSP1*. As aforementioned, the RSP1 states that, during the prototype merging stage and for each cluster $\mathcal{C}_{mc}(i)$, one prototype must be retrieved for each class present in it. The MRSP1 adapts such stage by resorting to a labelset approach (lines 16–18), *i.e.* each labelset is considered a different class and the instances with the same labelset are merged and assigned to it. Mathematically, set $\mathcal{R}_{ml}$ is obtained as:

$$\mathcal{R}_{ml} = \left\{ \left(\text{median}\left(\left\{ \boldsymbol{x}_j : (\boldsymbol{x}_j, \dagger_j) \in \mathcal{C}_{ml}(i), \dagger_j = \dagger_k \right\} \right), \dagger_k \right) \right\}_{i=1}^{n_d} \tag{3}$$

where $k = |\{\dagger \in \mathcal{C}_{ml}(i)\}|$ is the number of labelsets in the i-th cluster $\mathcal{C}_{ml}(i)$ and $j \in [1, |\mathcal{C}_{ml}(i)|]$. Fig. 2 d provides a graphical example of this procedure based on the space partitioning result depicted in Fig. 2 b.

The *Multilabel RSP2* or *MRSP2* proposal generalises the space partitioning approach based on the overlapping degree from the RSP2 method to the multilabel space (lines 12–13). For that, as in the MRSP1 proposal, we resort to a labelset approach: each labelset is considered a different class and the overlapping degree Φ_i of the i-th $C_{ml}(i)$ region is computed as the ratio of the average distance between instances belonging to different labelsets—$D^{\neq}$—and the average distance between instances of the same labelset—$D^{=}$.

In formal terms, for the i-th region, these pairwise distance values $D^{\neq}$ and $D^{=}$ are respectively computed as:

$$D^{\neq} = \left\{ d(\boldsymbol{x}_j, \boldsymbol{x}_k) : (\boldsymbol{x}_j, \boldsymbol{y}_j) \wedge (\boldsymbol{x}_k, \boldsymbol{y}_k) \in \mathcal{C}_{ml}(i), j \neq k, \boldsymbol{y}_j \neq \boldsymbol{y}_k \right\} \tag{4}$$

$$D^{=} = \left\{ d(\boldsymbol{x}_j, \boldsymbol{x}_k) : (\boldsymbol{x}_j, \boldsymbol{y}_j) \wedge (\boldsymbol{x}_k, \boldsymbol{y}_k) \in \mathcal{C}_{ml}(i), j \neq k, \boldsymbol{y}_j = \boldsymbol{y}_k \right\} \tag{5}$$

with $1 \leq j,\ k \leq n_d$. Based on this, the overlapping degree Φ_i for the same i-th region is eventually obtained as:

$$\Phi_i = \frac{\sum_{j=1}^{|D^{\neq}|} D^{\neq}(j)}{\sum_{k=1}^{|D^{=}|} D^{=}(k)} \cdot \frac{|D^{=}|}{|D^{\neq}|} \tag{6}$$

Note that, after the convergence of the space partitioning stage, the prototype merging policy in Eq. 3 introduced for MRSP1 is applied.

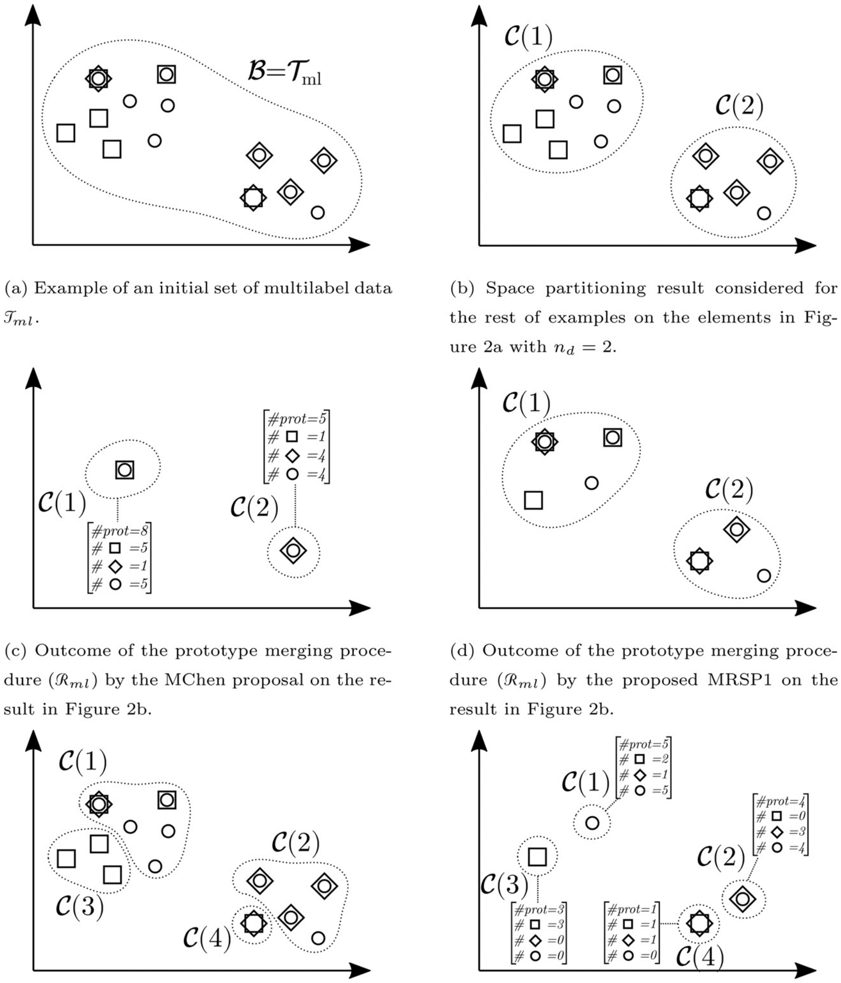
The last proposal is the *Multilabel RSP3* or *MRSP3*. In this case, we must generalise the cluster homogeneity concept of the RSP3 method to automatically estimate the n_d number of clusters. For that, we resort to the criterion posed by Ougiaroglou et al. [14] which states that a set of multilabel data is considered to be homogeneous if there is, at least, one common label among all the prototypes in the set, *i.e.* $\exists\ \lambda \in \mathcal{C}_{ml}(i) \text{s.t.} |\mathcal{C}_{ml}(i)|_\lambda = |\mathcal{C}_{ml}(i)|$. This substitutes the condition in line 3 in Algorithm 1 so that the process finishes when this homogeneity criterion is accomplished by all regions. After this space partitioning stage, the set of clusters $\mathcal{C}_{ml}$ is further processed following the prototype merging approach of the MChen proposal in Eq. 2.

Finally, Figs. 2 e and 2 f respectively show the result of the space partitioning and prototype merging phases of the introduced MRSP3 proposal.

4. Experimental set-up

This section presents the experimental scheme designed for comparatively assessing the proposed multilabel PG methods. For an easier description, this procedure is graphically illustrated in Fig. 3.

During the training phase of the procedure, the set of train data $\mathcal{T}_{ml} \subset \mathcal{X} \times \mathcal{Y}_{ml}$ is altered to induce certain noise level in the instances controlled by the user parameter $\theta \in [0, 1]$, retrieving set $\mathcal{T}'_{ml}$. Then, this latter data collection $\mathcal{T}'_{ml}$ is processed by a multilabel PG method to obtain a reduced version of the set—namely $\mathcal{R}_{ml}$—that is used as the reference set for the multilabel kNN-based classifier. It must be noted that the noise induction process represents an optional stage in the posed pipeline. Hence, as it will be shown, the first experimental part does not induce any noise by



(a) Example of an initial set of multilabel data $\mathcal{T}_{ml}$.

(b) Space partitioning result considered for the rest of examples on the elements in Figure 2a with $n_d = 2$.

(c) Outcome of the prototype merging procedure ($\mathcal{R}_{ml}$) by the MChen proposal on the result in Figure 2b.

(d) Outcome of the prototype merging procedure ($\mathcal{R}_{ml}$) by the proposed MRSP1 on the result in Figure 2b.

(e) Result of the space partitioning phase obtained by the MRSP3 proposal on the prototypes in Figure 2a.

(f) Outcome of the prototype merging procedure ($\mathcal{R}_{ml}$) by the proposed MRSP3 on the result in Figure 2e.

Fig. 2. Graphical illustration of the multilabel PG proposals introduced in the work. Fig. 2(a) represents a multilabel set of train data $\mathcal{T}_{ml}$ to be reduced. Fig. 2(b) shows the space partitioning results on which the different reduction proposals are based, except for the MRSP3 one, whose case is illustrated in Fig. 2(e). Prototype merging graphs 2(c) and 2(f) depict the number of prototypes—denoted as *#prot*—and the cardinality of labels—#□, #○, and #◇—for each of the original clusters.

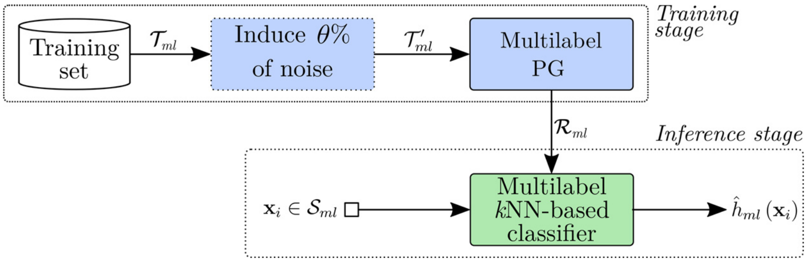

Training set
$\mathcal{T}_{ml}$
Induce θ% of noise
$\mathcal{T}'_{ml}$
Multilabel PG
Training stage
$\mathcal{R}_{ml}$
Inference stage
$\mathbf{x}_i \in \mathcal{S}_{ml}$
Multilabel kNN-based classifier
$\hat{h}_{ml}(\mathbf{x}_i)$

Fig. 3. Experimental scheme for the comparative assessment of the PG methods.

setting $\theta = 0$ while the second one will analyse the robustness and data cleansing capabilities of the reduction methods to the data corruption process by considering $\theta > 0$.

During the inference stage, a test set of multilabel data $\mathcal{S}_{ml} \subset \mathcal{X} \times \mathcal{Y}_{ml}$ drawn from the same distribution as the train data $\mathcal{T}_{ml}$ but disjoint from it is considered for evaluating the method. Using a $\hat{h}_{ml}(\cdot)$ prediction function from the particular multilabel kNN-based classification strategy at hand, each sample $\boldsymbol{x}_i \in \mathcal{S}_{ml}$ is given a labelset that is eventually compared to that in the ground-truth based on certain evaluation criteria.

The remainder of the section presents the corpora used for assessing the multilabel PG proposals, the noise induction procedure used, the considered kNN-based classification strategies, and the contemplated evaluation protocol.

4.1. Corpora

We have considered 12 multilabel corpora from the Mulan repository [30] comprising a varied range of domains, corpus sizes, initial space dimensionalities, and target label spaces. The precise

Table 1
Summary of the corpora considered for the experimentation. Each corpus is described in terms of its data domain, partition sizes, dimensionality of input data (features) and output space (labels), cardinality, and density.

Name	Domain	Corpus size		Dimensionality		Cardinality	Density	MeanIR
		Train	Test	Features (f)	Labels (L)			
Bibtex	Text	4,880	2,515	1,836	159	2.40	0.015	12.78
Birds	Audio	322	323	260	19	1.01	0.053	6.10
Corel5k	Image	4,500	500	499	374	3.52	0.009	183.29
Emotions	Music	391	202	72	6	1.87	0.311	1.49
Genbase	Biology	463	199	1,186	27	1.25	0.046	31.60
Medical	Text	333	645	1,449	45	1.25	0.028	48.59
rcvsubset1	Text	3,000	3,000	47,236	101	2.88	0.029	191.42
rcvsubset2	Text	3,000	3,000	47,236	101	2.63	0.026	177.89
rcvsubset3	Text	3,000	3,000	47,236	101	2.61	0.026	192.48
rcvsubset4	Text	3,000	3,000	47,229	101	2.49	0.025	170.84
Scene	Image	1,211	1,196	294	6	1.07	0.179	1.33
Yeast	Biology	1,500	917	103	14	4.24	0.303	7.27

details in terms of size, features, and label dimensionality of these sets are provided in Table 1. Note that the *cardinality*—average number of labels associated with each instance—and *density*—ratio of cardinality and label dimensionality of the corpus—measures are provided for each corpus as they represent common descriptors in the multilabel classification field. In addition, we also provide the *mean imbalance ratio* (*MeanIR*) index that estimates the imbalance level of multilabel corpora and is obtained as:

$$MeanIR = \frac{1}{|\mathcal{Y}_{ml}|} \sum_{\lambda \in |\mathcal{Y}_{ml}|} \frac{\max\limits_{\forall \lambda' \in \mathcal{Y}_{ml}} \left(\sum_{i=1}^{|\mathcal{T}_{ml}|} [\![\lambda' \in \dagger_i]\!] \right)}{\sum_{i=1}^{|\mathcal{T}_{ml}|} [\![\lambda \in \dagger_i]\!]} \tag{7}$$

where the descriptor $MeanIR \in [1, \infty)$ reports sharper imbalance rates as the value increases, and $[\![\cdot]\!] \rightarrow \{0, 1\}$ represents the Iverson bracket, which outputs the unit value when the condition in the argument is met and zero otherwise.

Note that, for the sake of reproducible research, we have used the partitions defined by Szymański and Kajdanowicz in these particular corpora [31].

4.2. Noise induction procedure

To examine the actual robustness of both the existing and the proposed multilabel PG methods, we artificially introduce noise in the data. Note that, to our best knowledge, no previous work has assessed the robustness of multilabel PG methods by performing a noise induction process. Hence, we resort to the procedure by Natarajan et al. [32] that is commonly considered in the multiclass PG literature: noise is introduced in the data by swapping the labels of pairs of prototypes randomly chosen from the train partition. Algorithm 2 provides a formal description of the adaptation of this procedure to the multilabel space, in which the user parameter $\theta \in [0, 1]$ represents the induced noise rate, *i.e.*, the percentage of prototypes that change their label.

As aforementioned, the particular case of $\theta = 0$ represents that in which no noise is induced in the corpus, hence being $\mathcal{T}'_{ml} = \mathcal{T}_{ml}$. In the subsequent experimentation, we will assess the proposals presented in this work considering both a noise-free scenario ($\theta = 0$) as well as under different levels of induced noise typically considered in the related literature.

Note that, while this particular noise induction policy may be deemed simplistic, it constitutes a first approximation to assess the robustness of multilabel PG methods in the context of label-level distortions. Nevertheless, other procedures that contemplate the multilabel nature of these data may also provide some additional insights about the performance of these methods, such as swapping only part of the labels between pairs of instances, randomly including or eliminating classes for each prototype, or simply duplicating labelsets among elements in the corpus, and will be explored in future research.

Algorithm 2: Noise induction procedure.

```
Input  : $\mathcal{T}_{ml} \subset \mathcal{X} \times \mathcal{Y}_{ml} \leftarrow$ Multilabel train corpus
         $\theta \leftarrow$ Noise level parameter
Output: $\mathcal{T}'_{ml} \subset \mathcal{X} \times \mathcal{Y}_{ml} \leftarrow$ Noisy multilabel train corpus
Let $\Theta = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{\theta \cdot |\mathcal{T}_{ml}|} \in_R \mathcal{T}_{ml}$    ▷ Random sampling of set $\mathcal{T}_{ml}$
Let $\mathcal{T}'_{ml} = \mathcal{T}_{ml} - \Theta$
for $i \in [0, \ldots, |\Theta|/2]$ do
    Save labelset of the $i$-th element in set $\Theta$: $\boldsymbol{y}' = \boldsymbol{y} \in \Theta_i$
    Put labelset in $|\Theta| - i$ in the $i$-th sample:
    $\boldsymbol{y} \in \Theta_i = \boldsymbol{y} \in \Theta_{|\Theta|-i}$
    Set $\boldsymbol{y}'$ as the labelset of the $|\Theta| - i$-th element:
    $\boldsymbol{y} \in \Theta_{|\Theta|-i} = \boldsymbol{y}'$
end for
Let $\mathcal{T}'_{ml} = \mathcal{T}'_{ml} \cup \Theta$
```

4.3. Classification strategies

We have selected three reference multilabel techniques based on *k*NN as classification methods: BR*k*NN and LP-*k*NN from the transformation paradigm as well as ML-*k*NN based on the algorithm adaptation premise. In all cases, the Euclidean distance has been used as the dissimilarity measure.

Regarding the k parameter representing the number of neighbours, we have considered the values $k \in \{1, 3, 5, 7\}$. Note that this parameter is not optimised by any means during the experimentation since the aim is to examine its influence on the overall classification performance in relation to the PG mechanisms.

4.4. Evaluation metrics

To assess the goodness of the proposals, we consider two criteria: classification performance and efficiency figures.

With respect to the former criterion, we resort to the Hamming Loss (HL) as it constitutes a commonly considered approach for measuring the goodness of multilabel classifiers [33]. This metric, which is defined as the fraction of the wrong predicted labels with respect to the total number of labels, can be mathematically posed as:

$$\mathrm{HL} = \frac{1}{|\mathcal{S}_{ml}|} \sum_{i=1}^{|\mathcal{S}_{ml}|} \frac{1}{L} \cdot \left| \dagger_i \, \Delta \, \hat{h}_{ml}(\boldsymbol{x}_i) \right| \tag{8}$$

Table 2
Results in terms of HL and resulting size for both the reference methods (exhaustive search, denoted as ALL, and MRHC) and our proposals (MChen, MRSP1, MRSP2, and MRSP3) when considering the different kNN-based classifiers. Non-dominated solutions per classifier are highlighted in bold type. Underlined values denote the best performance rates per PG scheme and classifier.

	Size	**BR*k*NN**				**LP-*k*NN**				**ML-*k*NN**			
		1	3	5	7	1	3	5	7	1	3	5	7
Reference													
ALL ■	100	9.09	7.94	7.69	7.56	9.09	8.71	8.54	8.45	9.09	7.92	7.72	7.66
MRHC ▼	59.62	8.76	7.70	7.49	7.51	8.76	8.47	8.57	8.68	8.76	7.87	7.91	7.85
Proposals													
MChen$_{10}$ ◆	**9.98**	7.92	**7.74**	7.78	7.83	7.92	**7.90**	7.98	7.97	7.92	7.87	**7.75**	7.86
MChen$_{30}$ ◆	**29.94**	8.00	7.58	**7.51**	7.55	8.00	7.91	**7.86**	7.87	8.00	7.70	7.77	7.81
MChen$_{50}$ ◆	49.96	8.29	7.71	7.53	7.38	8.29	8.30	8.06	8.10	8.29	7.85	7.63	7.57
MChen$_{70}$ ◆	69.97	8.51	7.72	7.62	7.46	8.51	8.35	8.40	8.49	8.51	7.90	7.82	**7.45**
MChen$_{90}$ ◆	89.02	8.73	7.62	7.29	7.17	8.73	8.35	8.16	8.33	8.73	7.61	7.53	7.55
MRSP1$_{10}$ ▲	61.88	8.95	7.96	7.60	7.40	8.95	8.88	9.03	8.97	8.95	8.21	7.95	7.68
MRSP1$_{30}$ ▲	**74.51**	8.77	7.76	7.36	**7.17**	8.77	8.55	8.34	8.50	8.77	7.84	**7.44**	7.56
MRSP1$_{50}$ ▲	80.11	8.80	7.78	7.44	7.27	8.80	8.41	8.34	8.50	8.80	7.84	7.50	7.54
MRSP1$_{70}$ ▲	84.37	8.86	7.77	7.45	7.30	8.86	8.43	8.32	8.46	8.86	7.79	7.54	7.52
MRSP1$_{90}$ ▲	90.78	8.84	7.71	7.33	7.17	8.84	8.46	8.27	8.46	8.84	7.68	7.51	7.57
MRSP2$_{10}$ ▲	61.09	8.85	7.90	7.61	7.36	8.85	8.92	8.93	8.87	8.85	8.04	7.86	7.79
MRSP2$_{30}$ ▲	78.07	8.79	7.87	7.53	7.34	8.79	8.52	8.26	8.39	8.79	7.88	7.68	7.47
MRSP2$_{50}$ ▲	**83.59**	8.74	7.76	7.45	7.27	8.74	8.59	8.36	8.34	8.74	7.79	7.58	**7.43**
MRSP2$_{70}$ ▲	**87.21**	8.80	7.65	7.38	**7.12**	8.80	8.44	8.36	8.30	8.80	7.69	7.66	7.52
MRSP2$_{90}$ ▲	89.28	8.76	7.68	7.49	7.34	8.76	8.47	8.51	8.34	8.76	7.80	7.51	7.50
MRSP3 ▲	66.88	8.53	7.62	7.50	7.34	8.53	8.29	8.12	8.23	8.53	7.94	7.78	7.71

where $\mathcal{S}_{ml} \subset \mathcal{X} \times \mathcal{Y}_{ml}$ denotes the multilabel set of test data, Δ is the symmetric difference of ground-truth $\dagger_i$ and predicted $\hat{h}_{ml}(\boldsymbol{x}_i)$ sets, and L is the number of labels.

As commonly done in the DR field, efficiency is assessed by comparing the size of the reduced set $\mathcal{R}_{ml}$ normalised by that of the training set $\mathcal{T}_{ml}$ [34]. Computation time, which is typically discarded as an evaluation metric due to its variability depending on the load of the computing system, is additionally reported as a supplementary figure of merit for the analysis of the particular implementations provided in this work.

It must be noted that PG methods for kNN seek to simultaneously optimise two contradictory goals, set size reduction and classification performance, being not possible to achieve a global optimum. Hence, as in reference works from the literature [35,36], we address it as a Multi-objective Optimisation Problem in which the two aforementioned objectives are meant to be optimised. The different solutions under this framework—there may exist more than one—are retrieved by resorting to the concept of non-dominance: one solution is said to dominate another if it is better or equal in each goal function and, at least, strictly better in one of them. Those elements, typically known as non-dominated, constitute the Pareto frontier in which all elements are deemed as optimal solutions without any order among them.

5. Results

This section introduces and discusses the results obtained by the proposed multilabel PG methods with the evaluation methodology considered. For comparison purposes, the reference MRHC method and the case in which no reduction process is applied—denoted as ALL—are included. Also, let subscript m represent the input parameter of the PG methods when required, *i.e.* MChen$_m$, MRSP1$_m$, and MRSP2$_m$, which relates to the number of partitions as $n_d = m \cdot |\mathcal{T}_{ml}|/100$. For assessing its influence in the scheme, we considered different values of this input parameter as $m \in \{10, 30, 50, 70, 90\}$.

The remainder of the section presents four particular experiments: (i) a first part in which the PG methods are comparatively evaluated obviating the noise induction process; (ii) a second one whose focus is the noise robustness and data cleansing capabilities of these PG schemes; (iii) a third passage that assesses the PG methods from the perspective of class-imbalance data; and (iv) a last part that benchmarks these strategies in terms of their execution time.

The implementation of the proposed PG methods and the experimental procedure considered is publicly available in: https://github.com/jose-jvmas/multilabel_PG. In addition, all obtained results for each individual corpus, configuration, and scenario contemplated are available at Mendeley Data (https://doi.org/10.17632/rbcnc6jcf3) for every single experiment performed in the work.

5.1. Comparative assessment of multilabel PG strategies

In this first experiment, we thoroughly compare the different reduction strategies using the aforementioned multilabel kNN-based classifiers as individual scenarios. In this regard, Table 2 and Fig. 4 show the results obtained in which the performance and reduction figures constitute the average of the individual values obtained for the corpora considered.

A first remark that may be observed is that the proposed methods fill a region in the space of possible solutions not previously occupied by existing multilabel PG methods. This is because some of the proposals (MChen, MRSP1, and MRSP2) allow selecting the size of the reduced set through a parameter. Note that, while this may be considered a drawback, such a feature allows prioritising either the reduction rate or classification performance depending on the particular application considered.

It can be also checked that, for all cases, MChen achieves the highest reduction rates, even when other parameter-based multilabel PG proposals consider the same m value. The main reason for such an effect is that, for a given m reduction setting, the MChen performs a rather aggressive reduction—especially with low m values—as it only retrieves a single prototype per region. On the contrary, the merging procedure for both MRSP1 and MRSP2 softens the single-prototype policy by the MChen proposal to compute an instance per existing labelset in the partitions, hence increasing the size of the R_{ml} resulting set. Regarding the MRSP3 method, the fact that the resulting set size may be deemed as medium-to-high (a 66.88% of the size of the ALL case) is due to the region-based homogeneity requirement of the space partitioning phase; in this sense, a more relaxed criterion (*e.g.*, allowing only partial label matches) should result in sharper reduction rates.

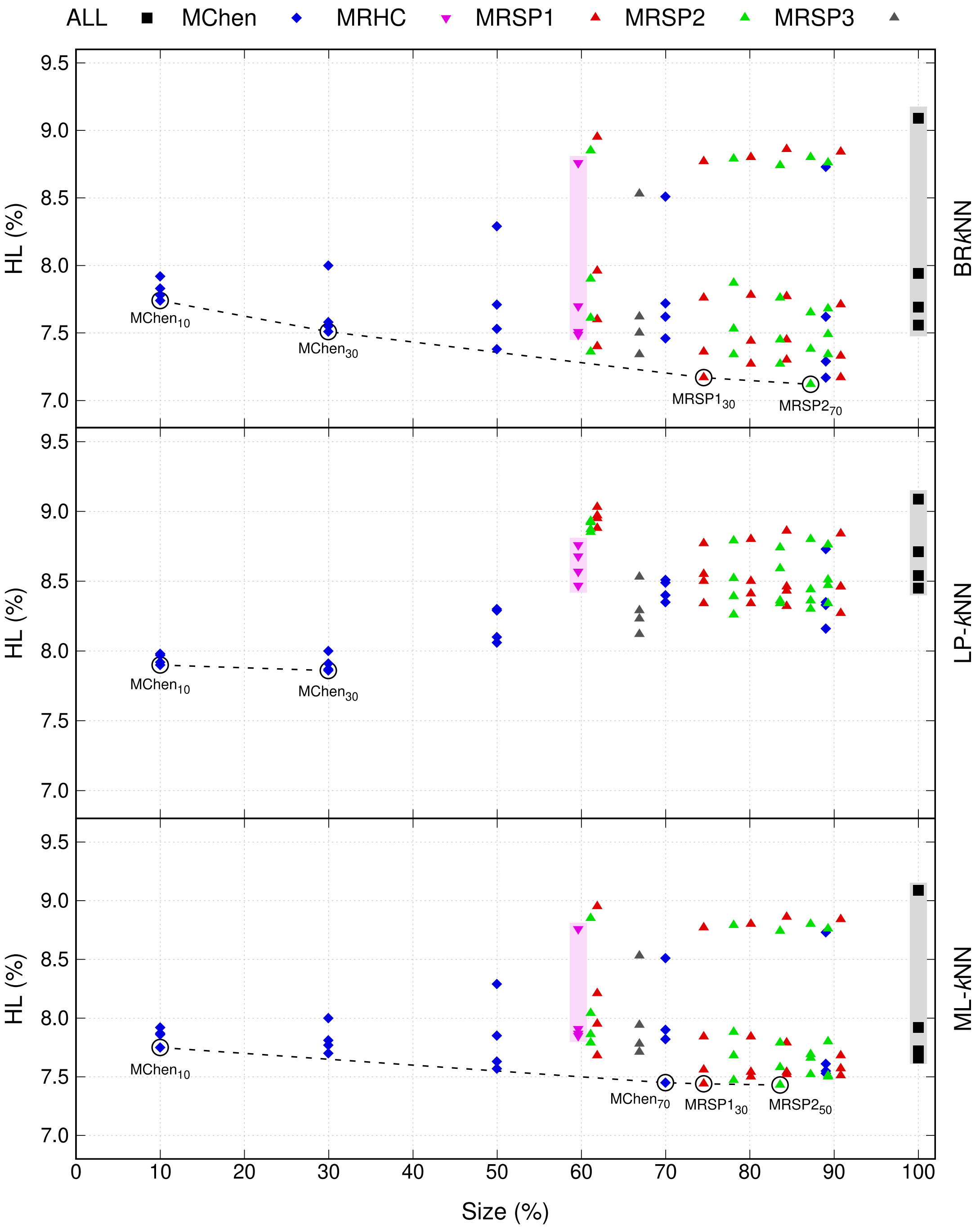


Fig. 4. Results in terms of HL and resulting size obtained with the *k*NN-based classifiers when considering the PG methods and the exhaustive search case (ALL) for the different *k* values tested. Circled methods and dashed lines represent the non-dominated elements and the Pareto frontiers in each scenario, respectively. For easier comparison, shaded areas depict the regions in the solution space occupied by the baseline cases (MRHC and ALL).

In all cases, since the PG process is applied before the classification stage, the resulting set sizes are the same for all scenarios, being the differences in performance only due to the particular capabilities of the classification scheme. It can be observed that LP-*k*NN may be deemed as the least competitive alternative since, for the same reduction scheme, HL figures tend to be higher than the other alternatives. Oppositely, BR*k*NN and ML-*k*NN show similar performance results since the HL figures do not remarkably differ among them. Such performance disparities among the classifiers are most likely due to the restrictiveness of the LP-*k*NN classifier that, in contrast to the BR*k*NN and ML-*k*NN methods, is not able to infer labelsets not seen during the training stage.

From the point of view of the PG strategies, the rather sharp reduction figures depicted by the MChen method—mainly due to the single-prototype policy of the merging stage—generally entails the least competitive classification rates among the PG techniques, being the sole exception found when contemplating the LP-*k*NN classifier. This fact is most probably due to that the resulting prototypes in that scenario only comprise the most relevant labels in the corpora, being hence guaranteed the inference of a representative part of the classes at the expense of missing sporadic labels. The rest of the cases—MRHC and the entire MRSP family—generally achieve better classification rates than the MChen alternative—and especially the MRSP1 and MRSP2 methods—for the different *k*NN-

Table 3
Wilcoxon signed-rank test results with $p < 0.05$ for the classifiers considered. Symbols √, ✗, and = respectively denote that, for each classification scenario, the non-dominated solution in the row significantly improves, worsens or does not differ from the reference one in the column for the performance (HL) or reduction (Size) criterion.

	ALL ■		MRHC ▼	
	HL	Size	HL	Size
BR*k*NN				
MChen_{10} ◆	=	√	=	√
MChen_{30} ◆	=	√	=	√
MRSP1_{30} ▲	√	√	√	=
MRSP2_{70} ▲	√	√	√	✗
LP-*k*NN				
MChen_{10} ◆	=	√	=	√
MChen_{30} ◆	√	√	√	√
ML-*k*NN				
MChen_{10} ◆	=	√	=	√
MChen_{70} ◆	=	√	√	=
MRSP1_{30} ▲	=	√	=	=
MRSP2_{50} ▲	=	√	√	✗

based classifiers. Again, an exception is found when considering the LP-*k*NN one, which is most likely due to the high label variability in the search space that visibly hinders the performance of this classifier.

In terms of non-dominance, it may be noted that the obtained Pareto frontiers in the different classification scenarios considered only comprise examples of the novel multilabel PG strategies proposed in the work: BR*k*NN contains MChen_{10}, MChen_{30}, MRSP1_{30}, and MRSP2_{70}; LP-*k*NN depicts the MChen_{10} and MChen_{30} cases; and ML-*k*NN points out four of them, which are MChen_{10}, MChen_{70}, MRSP1_{30}, and MRSP2_{90}. Hence, the ALL and MRHC cases may not be considered optimal solutions to the task as they are consistently dominated by the novel proposals presented in this work.

Finally, it may be also checked that the classification rates do generally improve as the number of neighbours considered—k parameter of the classifiers—increases. This fact suggests the presence of some noise in the corpora that is somehow palliated by adequately tuning this parameter. Note that, among the different multilabel classifiers studied, LP-*k*NN is the one that shows the least improvement when increasing this k value.

5.1.1. Statistical significance analysis

A significance analysis has been performed to statistically evaluate the results obtained. For that, we have considered the Wilcoxon signed-rank test [37] to assess whether the classification performance and reduction rate of the proposed PG methods significantly improve those of the baseline strategies. More precisely, for each classification scenario, we compare the results obtained by the elements of the particular Pareto frontier against the best figures obtained by the baseline MRHC and ALL methods. For that, we consider the individual results obtained—either performance or reduction—for each contemplated corpus in the experimentation. Table 3 shows the results of such analysis when considering a significance threshold of $p < .05$.

Focusing on the classification performance criterion (HL), it may be observed that the non-dominated elements in the Pareto frontier—exclusively defined by the proposals introduced in the work—statistically equal or improve the results of the baselines considered. However, since the particular conclusions are quite related to the actual classification scheme at hand, we shall now analyse them in a separate manner.

When considering the BR*k*NN classifier, the proposals depicting the highest reduction rates—MChen_{10} and MChen_{30}—show similar performance to both ALL and MRHC baseline cases; on the contrary, those schemes with larger resulting set sizes—MRSP1_{30} and MRSP2_{70}—do improve the reference strategies.

In the case of the LP-*k*NN classifier, a similar trend to that of the BR*k*NN is found: when performing a sharp reduction—MChen_{10} strategy—, the reported classification rate does not statistically differ to those of the baselines; however, when allowing a larger set size—the MChen_{30} method—, this performance indicator does improve those of the reference cases.

The results obtained with the ML-*k*NN classifier, however, do not show a similar tendency to the ones presented. As it may be observed, none of the non-dominated cases is able to statistically outperform the ALL case, while they do obtain similar performance scores with remarkably fewer prototypes. Regarding the MRHC base case, two of the proposals—MChen_{70} and MRSP2_{50}—do significantly improve this base case while the other two non-dominated elements—MChen_{10} and MRSP1_{30}—report statistically similar classification rates.

In relation to the analysis of the reduction capabilities, as expected, the results show that all non-dominated cases statistically improve the ALL case. Oppositely, when compared to the MRHC, there is a larger variability in the results: MChen generally outperforms the reference method except for the case of MChen_{70} in the ML-*k*NN scenario, which shows no statistical difference; the MRSP1_{30}—found in BR*k*NN and ML-*k*NN—also shows alike reduction capabilities to MRHC as the analysis points out no difference; finally, MRSP2_{70} and MRSP2_{50}, respectively found in the BR*k*NN and ML-*k*NN scenarios, stand for the cases in which the reduction results are statistically worse than MRHC, given that these methods do not remarkably reduce the set size of the reference corpus.

5.2. Noise robustness and data cleansing study

In this second experiment, we assess the performance of both the proposed multilabel PG strategies as well as the reference ones in scenarios with noisy data. For that, we consider the labelset swapping procedure introduced in Section 4.2 with $\theta \in \{20\%, 40\%\}$ as they stand as representative noise rates commonly considered in the related literature [32]. For comparative purposes, the case of $\theta = 0$ is also included to assess the base case in which no noise is induced. Note that, the different corpora in each experiment are affected by the same level of induced noise—*i.e.*, the same θ value for all corpora—, being the case of different noise levels per corpus posed as future work.

The results obtained in the different noise scenarios posed are depicted in Table 4 and Fig. 5. Note that, for simplicity, these figures constitute the average performance—both in terms of recognition rate and reduction capabilities—of the individual results per PG method and k classification parameter for the three *k*NN-based algorithms considered.

The induction of noise in the corpora clearly affects the overall performance since, in general, all studied cases depict lower classification rates as the noise level increases. While the use of high k classification values (e.g., $k = 5$ or $k = 7$) somehow palliates this effect, the best performance achieved in these noisy scenarios is indeed lower than that of the non-induced noise case.

Besides, all PG methods generally show worse reduction rates as the noise increases, being the MRHC and MRSP3 strategies particularly affected. Most likely, the induced noise results in a higher labelset diversity—*i.e.*, less label-level homogeneity—in the prototype groups obtained during the space partitioning stage of the methods, hence forcing the merging stage to generate a larger number of elements to satisfy the condition of retrieving as many prototypes as labelsets in the cluster. The sole exception to this as-

Table 4
Results in terms of HL and resulting size for both the reference methods (exhaustive search, denoted as ALL, and MRHC) and our proposals (MChen, MRSP1, MRSP2, and MRSP3) when considering the different noise scenarios posed. Each value constitutes the average performance obtained for the three classification methods considered. Non-dominated solutions per noise scenario are highlighted in bold type. Underlined values denote the best performance rates per PG scheme and noise scenario.

	Noise 0%					Noise 20%					Noise 40%				
	Size	k				Size	k				Size	k			
		1	3	5	7		1	3	5	7		1	3	5	7
Reference															
ALL ■	100	9.09	8.19	7.98	$\underline{7.89}$	100	9.80	8.50	8.22	$\underline{8.03}$	100	10.65	9.17	8.71	$\underline{8.53}$
MRHC ▼	59.62	8.76	8.01	$\underline{7.99}$	8.02	72.95	9.19	8.40	8.13	$\underline{8.02}$	80.67	10.01	8.94	8.59	$\underline{8.48}$
Proposals															
MChen_{10} ◆	**9.98**	7.92	$\underline{\mathbf{7.84}}$	$\underline{7.84}$	7.89	**9.98**	7.94	$\underline{\mathbf{7.84}}$	$\underline{7.84}$	7.93	**9.98**	8.19	7.95	$\underline{\mathbf{7.94}}$	8.01
MChen_{30} ◆	**29.94**	8.00	7.73	$\underline{\mathbf{7.71}}$	7.74	**29.94**	8.27	7.90	$\underline{\mathbf{7.74}}$	7.78	**29.94**	8.53	8.13	$\underline{\mathbf{7.89}}$	7.98
MChen_{50} ◆	**49.96**	8.29	7.95	7.74	$\underline{\mathbf{7.68}}$	49.96	8.55	8.18	7.86	$\underline{7.82}$	49.96	8.77	8.51	8.10	$\underline{8.03}$
MChen_{70} ◆	69.97	8.51	7.99	7.94	$\underline{7.80}$	69.97	8.89	8.31	8.05	$\underline{7.96}$	69.97	9.24	8.89	8.51	$\underline{8.25}$
MChen_{90} ◆	89.02	8.73	7.86	$\underline{7.66}$	7.68	89.02	9.38	8.20	7.87	$\underline{7.77}$	89.02	10.07	9.03	8.60	$\underline{8.40}$
MRSP1_{10} ▲	61.88	8.95	8.35	8.19	$\underline{8.02}$	65.72	9.85	8.83	8.50	$\underline{8.35}$	68.56	10.54	9.14	8.71	$\underline{8.58}$
MRSP1_{30} ▲	74.51	8.77	8.05	$\underline{7.71}$	7.74	**78.89**	9.52	8.39	8.04	$\underline{\mathbf{7.73}}$	81.73	10.38	9.07	8.46	$\underline{8.15}$
MRSP1_{50} ▲	80.11	8.80	8.01	$\underline{7.76}$	7.77	84.29	9.50	8.34	$\underline{8.00}$	7.84	87.23	10.40	9.08	8.51	$\underline{8.33}$
MRSP1_{70} ▲	84.37	8.86	8.00	7.77	$\underline{7.76}$	88.28	9.49	8.25	7.92	$\underline{7.82}$	91.38	10.33	9.08	8.57	$\underline{8.37}$
MRSP1_{90} ▲	90.78	8.84	7.95	$\underline{7.70}$	7.73	92.35	9.56	8.26	7.90	$\underline{7.81}$	93.82	10.31	9.10	8.64	$\underline{8.42}$
MRSP2_{10} ▲	61.09	8.85	8.29	8.13	$\underline{8.01}$	65.13	9.70	8.78	8.41	$\underline{8.30}$	66.44	10.48	9.06	8.70	$\underline{8.48}$
MRSP2_{30} ▲	78.07	8.79	8.09	7.82	$\underline{7.73}$	81.23	9.53	8.38	8.06	$\underline{7.78}$	83.41	10.21	8.78	8.40	$\underline{8.16}$
MRSP2_{50} ▲	83.59	8.74	8.04	7.80	$\underline{7.68}$	87.92	9.38	8.32	8.07	$\underline{7.76}$	89.37	10.34	8.93	8.41	$\underline{8.26}$
MRSP2_{70} ▲	**87.21**	8.80	7.93	7.80	$\underline{\mathbf{7.65}}$	90.53	9.35	8.28	$\underline{7.98}$	7.74	92.67	10.35	8.93	8.55	$\underline{8.26}$
MRSP2_{90} ▲	89.28	8.76	7.98	7.84	$\underline{7.73}$	92.16	9.46	8.16	7.87	$\underline{7.73}$	93.69	10.35	9.04	8.58	$\underline{8.42}$
MRSP3 ▲	66.88	8.53	7.95	7.80	$\underline{7.76}$	75.54	9.15	8.28	7.92	$\underline{7.81}$	81.33	9.84	8.85	8.31	$\underline{8.17}$

sertion is the MChen method whose reduction capabilities remain stable independently of the noise induced in the data since the prototype merging policy of this strategy always retrieves a single instance per partition.

Overall, it can be noted that MChen can be considered the best noise cleansing strategy since the classification schemes trained after that stage achieve the best overall HL performance figures. Most reasonably, since the merging policy of this strategy only keeps the most common labels in each cluster retrieved from the space partitioning stage, the method inherently removes the sporadic presence of uncommon labels in each of these groups, thus showing the aforementioned noise robustness. MRSP proposals, though, prove not to be that competitive against this type of noise since the performance of the classification schemes trained with the set obtained with those methods degrades as the presence of noise increases. Note that, since these merging policies produce as many prototypes as label combinations exist in each data partition, the higher labelset variability due to the noise induction process not only results in the generation of a larger amount of prototypes, but also that they may be incorrectly labelled. Regarding the reference MRHC method, it may be observed a similar performance trend to that of the MRSP family as they are based on similar reduction principles.

In terms of non-dominance, it may be observed that the different Pareto frontiers are entirely defined by the novel PG proposals introduced in this work: MChen_{10}, MChen_{30}, MChen_{50}, and MRSP2_{70} in the non-induced noise scenario; MChen_{10}, MChen_{30}, and MRSP1_{30} when $\theta = 20\%$; and MChen_{10} and MChen_{30} when considering the noisiest scenario of the ones studied in the work. In this regard, it may be concluded that the MChen algorithm proves itself as a considerably robust method—both in terms of efficiency and classification performance—against noisy situations, especially when set to high reduction rates (e.g., $m = 10\%$ or $m = 30\%$).

It must be noted that DR methods based on editing strategies are typically contemplated in multiclass scenarios as a means of removing noisy elements from the data to enhance the performance of a subsequent DR or classification technique [9]. In this context, and according to their reported noise removal capabilities, multilabel editing strategies such as the ones by Kanj et al. [38] or Arnaiz-González et al. [39] may be used for performing such a noise cleansing process before applying any particular multilabel PG method.

Table 5
Wilcoxon signed-rank test results with $p < 0.05$ for the noise scenarios posed. Symbols √, ✗, and = respectively denote that, for each classification scenario, the non-dominated solution in the row significantly improves, worsens or does not differ from the reference one in the column for the performance (HL) or reduction (Size) criterion.

	ALL ■		**MRHC** ▼	
	HL	Size	HL	Size
Noise 0%				
MChen_{10} ◆	=	√	=	√
MChen_{30} ◆	=	√	√	√
MChen_{50} ◆	=	√	√	√
MRSP2_{70} ▲	=	√	√	√
Noise 20%				
MChen_{10} ◆	=	√	√	√
MChen_{30} ◆	=	√	√	√
MRSP1_{30} ▲	√	√	√	√
Noise 40%				
MChen_{10} ◆	√	√	√	√
MChen_{30} ◆	√	√	√	√

5.2.1. Statistical significance analysis

As in the first experiment, we have considered the Wilcoxon signed-rank test to statistically compare the results obtained by the elements of the Pareto frontier against the best results obtained by the baseline MRHC and ALL methods for each noise scenario. Table 5 shows the outcome of such analysis when considering a significance threshold of $p < 0.05$.

As it may be observed, the multilabel PG strategies proposed in the work significantly improve the reduction rate of the baselines considered for all noise scenarios posed. Such a point suggests a remarkable robustness of our methods to the presence of noise in the data: while the reduction capabilities of the refer-

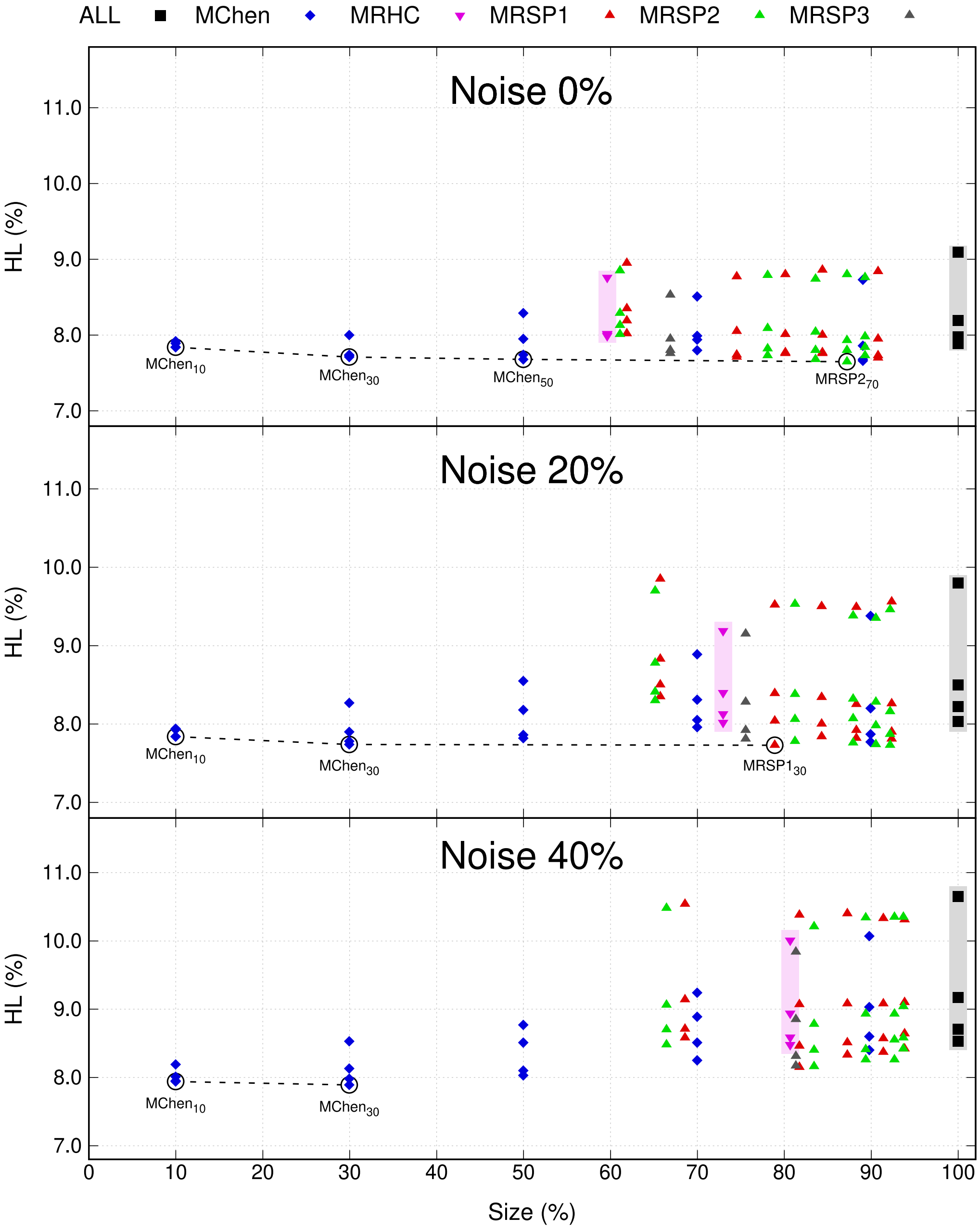


Fig. 5. Results in terms of HL and resulting size for the different noise scenarios when considering the PG methods and the exhaustive search case (ALL) for the k values tested. Note that each sample constitutes the average performance obtained for the three classification methods studied. Circled methods and dashed lines represent the non-dominated elements and the Pareto frontiers in each scenario, respectively. For easier comparison, shaded areas depict the regions in the solution space occupied by the baseline cases (MRHC and ALL).

ence MRHC strategy severely degrade as the noise in the data increases, the MChen is not affected by such an alteration whereas the MRSP1 and MRSP2 strategies do not degrade as much as the MRHC.

In relation to the classification rate, it may be noted that all non-dominated proposals either equal or improve the exhaustive search case with a significantly lower amount of prototypes. More precisely, our proposals improve the ALL case when inducing an elevated level of noise in the data while, when addressing scenarios with low levels of induced noise, the proposed multilabel methods in the Pareto frontier do not significantly differ from the exhaustive search cases.

Regarding the classification performance of the MRHC baseline method, it may be observed that this strategy is remarkably affected by the noise, being significantly outperformed by all the multilabel PG proposals in the non-dominated frontier. The sole exception to this assertion is the MChen$_{10}$ in the Noise 0% scenario that does not significantly differ from the MRHC case.

Overall, this analysis proves the superior robustness and noise cleansing capabilities of the proposed multilabel PG alternatives since, in the worst-case scenario, the classification rate achieved is similar to that of the exhaustive search but with a significantly lower amount of samples. Besides, it is also proved that the only existing multilabel PG method in the literature—the MRHC algorithm—is severely affected by these noisy scenarios—both in terms of efficiency and classification rate—, being hence outperformed by the novel multilabel PG proposals introduced in this work.

Table 6
Results in terms of the F_1^S (%) and AUC^S (%) figures of merit for the two class imbalance ranges considered—denoted as *Moderate* and *High*—as well as for all the entire data collection—designated as *All corpora*—for the non-dominated solutions in Section 5.1 per classification scenario. Figures reported for the reference ALL and MRHC strategies constitute those obtained with the best performing k classification parameter for each particular case. The best performance figures per classifier, imbalance level, and metric are highlighted in bold. The resulting set size is provided for comparative purposes.

	Moderate			High			All corpora		
	F_1^S	AUC^S	Size	F_1^S	AUC^S	Size	F_1^S	AUC^S	Size
BR*k*NN									
ALL ■	23.09	60.88	100	**3.50**	**50.95**	100	21.55	58.95	100
MRHC ▼	22.87	60.60	60.80	3.02	50.83	55.55	20.33	58.43	59.62
$MChen_{10}$ ◆	14.35	56.44	9.95	1.25	50.16	10.00	14.01	55.60	9.98
$MChen_{30}$ ◆	11.77	55.20	29.85	1.44	50.21	30.00	15.03	56.00	29.94
$MRSP1_{30}$ ▲	20.66	59.77	75.44	2.81	50.82	80.86	20.70	58.72	74.51
$MRSP2_{70}$ ▲	**24.10**	**61.47**	88.35	2.49	50.75	87.63	**21.71**	**59.21**	87.21
LP-*k*NN									
ALL ■	35.29	66.44	100	**5.84**	**52.05**	100	26.34	60.87	100
MRHC ▼	**40.67**	**69.34**	60.80	4.67	51.43	55.55	**26.70**	**61.21**	59.62
$MChen_{10}$ ◆	11.71	55.30	9.95	1.00	50.07	10.00	12.78	55.07	9.98
$MChen_{30}$ ◆	13.18	55.91	29.85	1.42	50.19	30.00	16.29	56.45	29.94
ML-*k*NN									
ALL ■	29.04	64.04	100	**4.61**	51.39	100	**23.71**	59.99	100
MRHC ▼	24.58	61.77	60.80	4.24	51.16	55.55	19.75	58.17	59.62
$MChen_{10}$ ◆	15.33	56.92	9.95	1.44	50.25	10.00	13.94	55.62	9.98
$MChen_{70}$ ◆	22.59	60.82	69.98	3.61	51.13	70.00	20.92	58.90	69.97
$MRSP1_{30}$ ▲	27.26	63.52	75.44	4.40	**51.53**	80.86	23.23	**60.15**	74.51
$MRSP2_{50}$ ▲	**30.13**	**64.76**	83.72	3.84	51.14	85.40	23.52	60.08	83.59

5.3. Class imbalance analysis

In order to provide some additional insights regarding the capabilities of the multilabel PG methods, this third experiment assesses their performance by attending to the label-based imbalance ratio of the different corpora used in the work. More precisely, to observe possible relations between the imbalance degree and the overall performance, we have gathered the different corpora based on their respective *MeanIR* score (see Table 1): a first moderate class imbalance group ($10 \leq MeanIR < 100$) comprising the *bibtex, genbase*, and *medical* sets; and a second highly-imbalanced collection with ($MeanIR \geq 100$) containing *Corel5k* and all the *rcv1subset* corpora.

Regarding the evaluation procedures, two representative example-based figures of merit for imbalance data have been considered [40]: the F-measure (F_1^S) and the Area Under the Receiver Operating Characteristic Curve (AUC^S). Based on the notation introduced in this work, these metrics are defined as:

$$F_1^S = \frac{1}{|\mathcal{S}_{ml}|} \cdot \sum_{i=1}^{|\mathcal{S}_{ml}|} \frac{2 \cdot |\dagger_i \cap \hat{h}_{ml}(\S_i)|}{|\dagger_i| + |\hat{h}_{ml}(\S_i)|} \tag{9}$$

$$[7\,pt]AUC^S = \frac{1}{|\mathcal{S}_{ml}|} \cdot \sum_{i=1}^{|\mathcal{S}_{ml}|} \frac{|\hat{h}_{ml}(\S_i)|}{|\dagger_i| \cdot (|\mathcal{Y}_{ml}| - 1)} \tag{10}$$

where $\mathcal{S}_{ml} \subset \mathcal{X} \times \mathcal{Y}_{ml}$ denotes the multilabel set of test data, elements $\dagger_i$ and $\hat{h}_{ml}(\boldsymbol{x}_i)$ respectively stand for the ground-truth and predicted labelsets, and $\mathcal{Y}_{ml}$ denotes the target label space.

Considering all the above, Table 6 presents the results obtained for the aforementioned imbalance-based corpora assortments—namely, *Moderate* and *High*—together with the case of examining all data collections—denoted as *All corpora*—for the two contemplated metrics as well as their average resulting size. Note that, for the sake of conciseness and comparison with the analyses performed in the previous sections, this study assesses the non-dominated solutions per classification scheme obtained in Section 5.1 as well as the best-performing configurations for the baseline cases.

In light of the results obtained, a first remark that may be observed is that the label imbalance in the data severely affects the overall performance of the schemes. More precisely, while the F_1^S score in the *Moderate* scenario gets to achieve values of up to 40% (MRHC with LP-*k*NN), the same metric rarely surpasses a 5% in the *High* scenario (ALL case with LP-*k*NN). Similarly, the AUC^S metric evaluation barely surpasses the random guess (*i.e.*, $AUC^S = 50\%$) when addressing highly-imbalanced data whereas the moderately-imbalance corpora generally achieve AUC^S values over 60%. The *All corpora* case shows an alike behaviour to that of the *Moderate* scenario as the almost-balanced corpora that this assortment incorporates remarkably reduce the overall imbalance ratio. The subsequent analyses thoroughly develop the presented general observations for each of the imbalance-level scenarios.

Focusing on the *Moderate* assortment, it may be checked that the different approaches follow similar trends to those observed in the previous sections: MChen generally depicts the least competitive classification rates due to its sharp reduction whereas the rest of the methods improve these figures as they perform more conservative reduction processes. Moreover, while the ALL case typically represents the best-performing option in terms of classification rate, some of the PG methods do prove to outperform its results—*e.g.*, the MRHC with the LP-*k*NN classifier or the $MRSP2_{30}$ with the ML-*k*NN one. Such an effect is mostly due to the inherent noise cleansing capabilities of the different PG alternatives, which prove to work in these relatively imbalanced scenarios.

In addition, and as previously discussed, these experiments also state a dependency between the reduction technique and the classification strategy. More precisely, MRHC does report the best performance when paired with the LP-*k*NN method whereas the MRSP family is particularly relevant in the BR*k*NN and ML-*k*NN scenarios. Such an insight should be further analysed in future work with the aim of devising a PG strategy that adequately exploits the individual advantages of each *k*NN-based multilabel classification algorithm.

Regarding the *High* imbalance scenario, it can be observed that the classification performance generally degrades after the reduc-

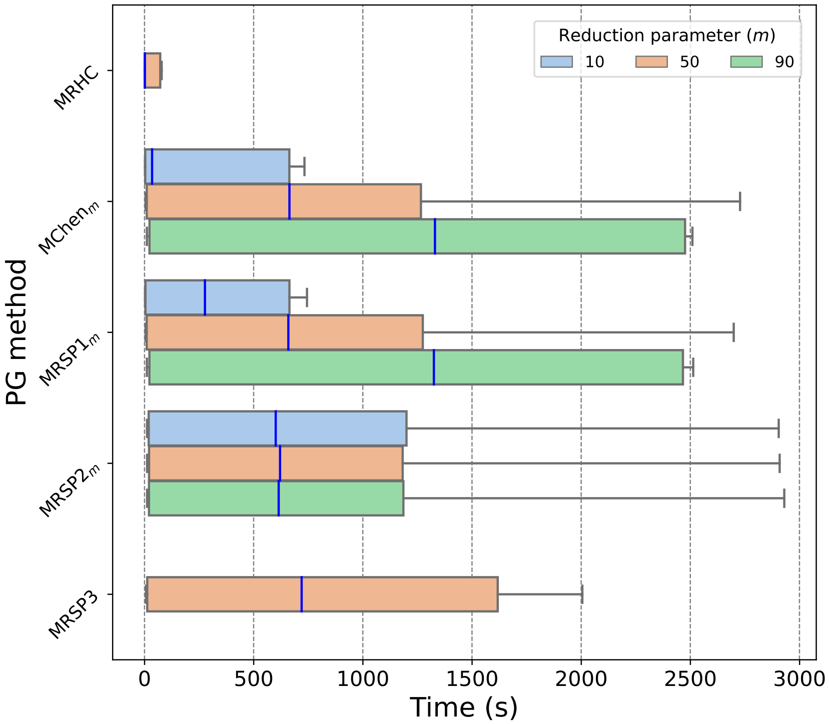


Fig. 6. Results, in seconds, of the execution time for each multilabel PG method and m reduction parameter (when applicable). Each sample of the boxplot graph stands for the reduction time obtained in each particular corpus and fold. Note that the cases of $m = 30$ and $m = 70$ have been omitted for conciseness as they represent intermediate cases of the reported figures.

tion process. While some of the techniques still report competitive figures—*e.g.*, the case of the ML-*k*NN classifier where the MRSP1$_{30}$ reports a decrease of just 0.21% in the F_1^S metric with respect to the ALL case or the BR*k*NN scenario in which the MRHC decreases just 0.48% in the F_1^S score compared to the exhaustive search—it is noticeable that none of the reduced cases outperform the ALL case. The sole exception to this is the case of the MRSP1$_{30}$ strategy that slightly improves the exhaustive search with the ML-*k*NN classifier in the AUC^S figure of merit.

Such a performance decrease in these particular imbalance scenarios relates to the fact that no considerations for such cases were contemplated when devising the methods. In this regard, a severely under-represented label may be assumed as noise, being most likely removed from the resulting set. This can be observed in the different MChen cases, in which the performance remarkably degrades, especially when set to perform sharp reductions (*i.e*, low *m* parameter). Such a limitation is expected to be thoroughly studied with the aim of devising specific multilabel PG policies capable of dealing with class imbalance.

The case of the *All corpora* assortment shows an alike behaviour to that of the *Moderate* imbalance in that the reduction methods generally show slightly lower classification rates than the exhaustive search. An exception to this is the MRSP2$_{70}$ strategy with the BR*k*NN classifier or the MRSP1$_{30}$ and MRSP2$_{50}$ alternatives with the ML-*k*NN model that surpass their respective ALL baseline cases in, at least, one of figures of merit, reinforcing the noise reduction capabilities of the methods. On a final note, it may be observed that, when set to high *m* values, the MRSP family generally achieves similar classification scores to the exhaustive search, while still performing certain size reduction. In this regard, in the event of addressing a given data collection with an unknown imbalance degree, it may be advisable to consider these PG alternatives in contrast to other possibilities such as the MChen or the reference MRHC methods.

5.4. Execution time benchmark

This last experiment assesses the proposed multilabel PG methods in terms of their execution time and compares them with that of the reference MRHC method. For that, we have performed five different executions of all the reduction processes for each particular algorithm configuration and corpus when addressing the case in which no label noise is induced in the data. The results obtained are summarised in Fig. 6, where the samples of the boxplot graphs correspond to the individual execution times obtained in each of the aforementioned reduction scenarios. Note that, this evaluation does not relate to the computational complexity of the studied methods but to the efficiency figures of the precise implementations facilitated in the code repository.

As it can be checked, based on the time execution benchmark, the MRHC implementation stands as the most competitive proposal of all the reduction strategies studied. Such a result is totally reasonable since the space partitioning stage of the MRHC algorithm, which is based on the *k*-means clustering method [1], has been directly drawn from the optimised and efficient-oriented scikit-learn library [41]. The rest of the implementations, however, do not consider any optimised toolkit that may boost the different procedures within, hence depicting higher execution times.

In a more detailed analysis, it is observed that the execution time for all configurable methods generally grows as the *m* reduction parameter increases. The sole exception to this assertion is the case of the MRSP2 method, in which the figures obtained remain relatively stable for all *m* configurations. Taking into account the low computational burden of the MRSP2 prototype merging policy, this stability insight suggests certain independence between the time consumption of the particular space partitioning policy devised for this MRSP2 method and the *m* reduction parameter.

Moreover, the presence of outliers in some of the methods—more precisely, MChen$_{20}$, MRSP1$_{20}$, and all MRSP2 versions—points out some difficulties when addressing particular cases. Most likely, this is due to a rather time-consuming space partitioning phase—typically, a large amount of pair-wise dissimilarity computations involving high-dimensional instances—as opposed to the prototype merging one, due again to the fact that the merging policies introduced in the work do not entail such elevated computation burdens.

In light of the results obtained, certain mechanisms may be introduced to palliate the inefficiency figures observed, such as the pre-calculation of all dissimilarities or the use of optimised libraries (*e.g.*, scikit-learn) to boost some of the intermediate processes. Nevertheless, and as aforementioned, the observed inefficiency issues as well as the conclusions drawn out of them only relate to the particular implementations facilitated as part of this work.

Finally, it must be highlighted that the figures provided in this analysis exclusively reflect the time invested in reducing the set of training data, which may be deemed as the equivalent to the train phase in an eager learning model. Hence, once this set of data is reduced, the inference time only relates to the inherent efficiency of the *k*NN-based classifier as well as the size of the resulting set obtained by the PG method, but not to the execution time of the latter.

6. Conclusions and future work

Prototype Generation (PG) represents one of the most competitive approaches for improving the efficiency of the *k*-Nearest Neighbour (*k*NN) classifier, which is typically related to low-efficiency figures when tackling scenarios with large amounts of data. Nevertheless, while PG methods are commonly considered in multiclass scenarios, very scarce works have addressed such a task in multilabel frameworks.

This work presents the first-time adaptation of four multiclass PG methods to the multilabel case: the reference Chen method [27] and the three versions of the well-known Reduction through Space Partitioning [28]. For that, we generalise to the multilabel space the different criteria considered by each method for gathering sets of prototypes (space partitioning stage) which are then combined according to certain policies (prototype merging). These novel proposals have been evaluated with 3 multilabel *k*NN-based classifiers, 12 multilabel corpora comprising a varied range of domains and corpus sizes, and different noise scenarios obtained by exchanging the labels of the instances in the train partition.

The results obtained show that the proposed adaptations are capable of significantly improving, both in terms of efficiency and efficacy, the only reference work in the literature—Multilabel Reduction through Homogeneous Clustering method by Ougiaroglou et al. [14]—as well as the case in which no PG method is applied—the exhaustive search. It is also proved that some of these adaptations show high robustness and data cleansing capabilities in the presence of noise. More precisely, when set to high reduction rates, the proposed Multilabel Chen strategy allows training classification schemes that statistically outperform those trained with the existing baseline approaches. Moreover, the user parameter of these methods allows prioritising either the efficiency or performance features of the scheme, depending on the particular application.

Future work considers the further analysis of the proposed methods contemplating the particularities of multilabel scenarios such as their performance in relation to the data label cardinality or the possible correlations among labels. The second point of interest is the exploration of alternative criteria for the partitioning and prototype merging stages, including the proposal of novel homogeneity policies, which may result in more efficient and/or robust classifiers as well as tackling the limitations observed in these methods when dealing with imbalanced data. Moreover, we consider that additional insights may be obtained by performing other noise induction policies such as swapping only part of the labels between instances, randomly including or eliminating classes for each prototype, or simply duplicating labelsets among elements in the corpus.

In a more practical sense, the presented methods would remarkably benefit from an efficient implementation so that their execution time could be reduced, which represents one of the main limitations observed. From a more general perspective, given the commented scarcity of multilabel PG strategies, future research contemplates the adaptation of other multiclass PG schemes to this particular scenario. Finally, in light of the noise robustness capabilities of the proposals, they may be considered as preprocessing techniques for other classification schemes such as Support Vector Machine or neural models in task-oriented cases such as *music tagging* or *image classification*, among others.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Multilabel Prototype Generation for Data Reduction in k-Nearest Neighbour classification (Mendeley Data).

Acknowledgments

This research was partially funded by the Spanish Ministerio de Ciencia e Innovación through the MultiScore (PID2020-118447RA-I00) and DOREMI (TED2021-132103A-I00) projects. The first author is supported by grant APOSTD/2020/256 from “Programa I+D+i de la Generalitat Valenciana”.

References

[1] P.E. Hart, D.G. Stork, R.O. Duda, Pattern Classification, Wiley Hoboken, 2000.
[2] C.M. Bishop, Pattern recognition, Mach Learn 128 (9) (2006).
[3] S. Suyanto, S. Meliana, T. Wahyuningrum, S. Khomsah, A new nearest neighbor-based framework for diabetes detection, Expert Syst Appl 199 (2022) 116857.
[4] A. George, X.A. Mary, S.T. George, Development of an intelligent model for musical key estimation using machine learning techniques, Multimed Tools Appl (2022) 1–20.
[5] E. Hancer, I. Hodashinsky, K. Sarin, A. Slezkin, A wrapper metaheuristic framework for handwritten signature verification, Soft comput 25 (13) (2021) 8665–8681.
[6] T. Mitchell, Machine Learning, McGraw Hill Burr Ridge, 1997.
[7] Z. Deng, X. Zhu, D. Cheng, M. Zong, S. Zhang, Efficient kNN classification algorithm for big data, Neurocomputing 195 (2016) 143–148.
[8] A.-J. Gallego, J.R. Rico-Juan, J.J. Valero-Mas, Efficient k-nearest neighbor search based on clustering and adaptive k values, Pattern Recognit 122 (2022) 108356.
[9] S. García, J. Luengo, F. Herrera, Data Preprocessing in Data Mining, volume 72, Springer, 2015.
[10] H.J. Escalante, M. Graff, A. Morales-Reyes, Pggp: prototype generation via genetic programming, Appl Soft Comput 40 (2016) 569–580.
[11] I. Triguero, J. Derrac, S. Garcia, F. Herrera, A taxonomy and experimental study on prototype generation for nearest neighbor classification, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 42 (1) (2012) 86–100.
[12] L. Nanni, A. Lumini, Prototype reduction techniques: a comparison among different approaches, Expert Syst Appl 38 (9) (2011) 11820–11828.
[13] M.-L. Zhang, Z.-H. Zhou, A review on multi-label learning algorithms, IEEE Trans Knowl Data Eng 26 (8) (2013) 1819–1837.
[14] S. Ougiaroglou, P. Filippakis, G. Evangelidis, Prototype generation for multi-label nearest neighbours classification, in: Hybrid Artificial Intelligent Systems, Springer International Publishing, Cham, 2021, pp. 172–183.
[15] S. Ougiaroglou, G. Evangelidis, Efficient dataset size reduction by finding homogeneous clusters, in: Proceedings of the Fifth Balkan Conference in Informatics, 2012, pp. 168–173.
[16] A.-J. Gallego, J. Calvo-Zaragoza, J.J. Valero-Mas, J.R. Rico-Juan, Clustering-based k-nearest neighbor classification for large-scale data with neural codes representation, Pattern Recognit 74 (2018) 531–543.

[17] M. Bello, G. Nápoles, K. Vanhoof, R. Bello, On the generation of multi-label prototypes, Intell. Data Anal. 24 (S1) (2020) 167–183.

[18] J.M. Moyano, E.L. Gibaja, K.J. Cios, S. Ventura, Review of ensembles of multi-label classifiers: models, experimental study and prospects, Information Fusion 44 (2018) 33–45.

[19] E. Gibaja, S. Ventura, Multi-label learning: a review of the state of the art and ongoing research, Wiley interdisciplinary reviews: data mining and knowledge discovery 4 (6) (2014) 411–444.

[20] M.-L. Zhang, Y.-K. Li, X.-Y. Liu, X. Geng, Binary relevance for multi-label learning: an overview, Frontiers of Computer Science 12 (2) (2018) 191–202.

[21] N. Rastin, M.Z. Jahromi, M. Taheri, A generalized weighted distance k-nearest neighbor for multi-label problems, Pattern Recognit 114 (2021) 107526.

[22] G. Tsoumakas, I. Katakis, I. Vlahavas, Random k-labelsets for multilabel classification, IEEE Trans Knowl Data Eng 23 (7) (2010) 1079–1089.

[23] M.-L. Zhang, Z.-H. Zhou, ML-KNN: A lazy learning approach to multi-label learning, Pattern Recognit 40 (7) (2007) 2038–2048.

[24] Z. Younes, F. Abdallah, T. Denøeux, Multi-label classification algorithm derived from k-nearest neighbor rule with label dependencies, in: 2008 16th European Signal Processing Conference, IEEE, 2008, pp. 1–5.

[25] W. Cheng, E. Hüllermeier, Combining instance-based learning and logistic regression for multilabel classification, Mach Learn 76 (2) (2009) 211–225.

[26] X. Zhu, C. Ying, J. Wang, J. Li, X. Lai, G. Wang, Ensemble of ML-KNN for classification algorithm recommendation, Knowl Based Syst 221 (2021) 106933.

[27] C.H. Chen, A. Jóźwik, A sample set condensation algorithm for the class sensitive artificial neural network, Pattern Recognit Lett 17 (8) (1996) 819–823.

[28] J.S. Sánchez, High training set size reduction by space partitioning and prototype abstraction, Pattern Recognit 37 (7) (2004) 1561–1564.

[29] F.J. Castellanos, J.J. Valero-Mas, J. Calvo-Zaragoza, Prototype generation in the string space via approximate median for data reduction in nearest neighbor classification, Soft comput 25 (2021) 15403–15415.

[30] G. Tsoumakas, E. Spyromitros-Xioufis, J. Vilcek, I. Vlahavas, Mulan: a java library for multi-label learning, Journal of Machine Learning Research 12 (2011) 2411–2414.

[31] P. Szymański, T. Kajdanowicz, Scikit-Multilearn: A scikit-based Python environment for performing multi-label classification, Journal of Machine Learning Research 20 (1) (2019) 209–230.

[32] N. Natarajan, I.S. Dhillon, P.K. Ravikumar, A. Tewari, Learning with noisy labels, Adv Neural Inf Process Syst 26 (2013).

[33] G. Madjarov, D. Kocev, D. Gjorgjevikj, S. Džeroski, An extensive experimental comparison of methods for multi-label learning, Pattern Recognit 45 (9) (2012) 3084–3104.

[34] J.R. Rico-Juan, J.J. Valero-Mas, J. Calvo-Zaragoza, Extensions to rank-based prototype selection in k-nearest neighbour classification, Appl Soft Comput 85 (2019) 105803.

[35] J. Calvo-Zaragoza, J.J. Valero-Mas, J.R. Rico-Juan, Improving kNN multi-label classification in prototype selection scenarios using class proposals, Pattern Recognit 48 (5) (2015) 1608–1622.

[36] J.J. Valero-Mas, J. Calvo-Zaragoza, J.R. Rico-Juan, J.M. Iñesta, An experimental study on rank methods for prototype selection, Soft comput 21 (19) (2017) 5703–5715.

[37] J. Demšar, Statistical comparisons of classifiers over multiple data sets, Journal of Machine Learning Research 7 (2006) 1–30.

[38] S. Kanj, F. Abdallah, T. Denoeux, K. Tout, Editing training data for multi-label classification with the k-nearest neighbor rule, Pattern Analysis and Applications 19 (1) (2016) 145–161.

[39] Ã. Arnaiz-Gonzlez, J.-F. Dez-Pastor, J.J. Rodrguez, C. Garca-Osorio, Local sets for multi-label instance selection, Appl Soft Comput 68 (2018).

[40] B. Liu, K. Blekas, G. Tsoumakas, Multi-label sampling based on local label imbalance, Pattern Recognit 122 (2022) 108294.

[41] 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, E. Duchesnay, Scikit-learn: machine learning in python, Journal of Machine Learning Research 12 (2011) 2825–2830.

Jose J. Valero-Mas obtained the M.Sc. in Telecommunications Engineering from the University Miguel Hernndez of Elche in 2012, the M.Sc. in Sound and Music Computing from the Universitat Pompeu Fabra in 2013, and the Ph.D. in Computer Science from the University of Alicante in 2017. He is currently a postdoctoral researcher with a grant from the Valencian Government at the Department of Software and Computing Systems of the University of Alicante, Spain. His research interests include Pattern Recognition, Machine Learning, Music Information Retrieval, and Signal Processing for which he has co-authored more than 30 works within international journals, conference communications, and book chapters.

Antonio Javier Gallego is an associate professor in the Department of Software and Computing Systems at the University of Alicante, Spain. He received B.Sc. & M.Sc. degrees in Computer Science from the University of Alicante in 2004, and a Ph.D. in Computer Science and Artificial Intelligence from the same university in 2012. He has been a researcher on 15 research projects funded by the Spanish Government and private companies. He has authored more than 60 works published in international journals, conferences, and books. His research interests include Deep Learning, Pattern Recognition, Computer Vision, and Remote Sensing.

Pablo Alonso-Jiménez holds a B.Sc. in Telecommunications Engineering from the Universidad de Vigo (2016) and an M.Sc. in Sound and Music Computing from the Universitat Pompeu Fabra (2017), where he is currently pursuing the Ph.D. in the Music Technology Group under the supervision of Dr. Xavier Serra. His fields of interest include signal processing and machine learning for audio, speech, and music applications.

Xavier Serra received a Ph.D. degree in computer music from Stanford University, Stanford, CA, USA, in 1989. He is currently a Professor with the Department of Information and Communication Technologies and the Director of the Music Technology Group, Universitat Pompeu Fabra, Barcelona, Spain. His research interests include the computational analysis, description, and synthesis of sound and music signals. Dr. Serra is very active in the fields of audio signal processing, sound and music computing, music information retrieval and computational musicology at the local and international levels, being involved in the editorial board of a number of journals and conferences and giving lectures on current and future challenges of these fields. He was awarded an Advanced Grant from the European Research Council to carry out the project CompMusic aimed at promoting multicultural approaches in music information research.