Graded Multilabel Classification by Pairwise Comparisons

Christian Brinker
Technische Universität Darmstadt
christian.brinker@stud.tu-darmstadt.de

Eneldo Loza Mencía
Technische Universität Darmstadt
eneldo@ke.tu-darmstadt.de

Johannes Fürnkranz
Technische Universität Darmstadt
juffi@ke.tu-darmstadt.de

***Abstract*—The task in multilabel classification is to predict for a given set of labels whether each individual label should be attached to an instance or not. Graded multilabel classification generalizes this setting by allowing to specify for each label a degree of membership on an ordinal scale. This setting can be frequently found in practice, for example when movies or books are assessed on a one-to-five star rating in multiple categories. In this paper, we propose to reformulate the problem in terms of preferences between the labels and their scales, which can then be tackled by learning from pairwise comparisons. We present three different approaches which make use of this decomposition and show on three datasets that we are able to outperform baseline approaches. In particular, we show that our solution, which is able to model pairwise preferences across multiple scales, outperforms a straight-forward approach which considers the problem as a set of independent ordinal regression tasks.**

***Keywords*—graded multilabel classification, ordinal classification, learning by pairwise comparisons**

I. Introduction

Multilabel Classification (MLC), the task of learning to assign multiple labels to a single data item, has received a lot of attention in the recent machine learning literature [1] because it has many real-world applications such as tagging of messages in blogs, annotating images, or assigning keywords to scientific papers. However, often we need to predict a degree or grade of membership to a particular category or label, instead of only whether this label is present or not. Cheng, Dembczyński, and Hüllermeier [2] introduced this task as *Graded Multilabel Classification* (GMLC). For example, TV guides often rate a movie on a scale from one to five stars in several different categories such as 'fun', 'action', 'sex', or 'suspense', as is shown in Table I. Users may find the additional information in the form of grades of memberships in contrast to simple binary assignments of genres very useful, and appreciate it for choosing their individual TV programs. Another application is the prediction of answers from questionnaires, where a common setting is to ask the probands to answer a series of questions and to respond on a graded scale of agreement, frequency, importance, quality or likelihood.

Although superficially similar, this task differs from a classical recommendation task [3]. While in both cases one essentially needs to make ordinal predictions that correspond to ratings, in recommender systems the training information is a sparsely populated rating matrix and the task is to predict (some of) the missing values. In contrast, the training information for GMLC is a complete matrix where each of the objects in the lines is characterized with a set of features (e.g., features that characterize the respective movie), and the task is to predict the entries for a new line, given the features that correspond to this new entry.

In this paper, an extended version is available as [4], we assume an inherent preference structure between the labels in combination with their grade of membership, and propose pairwise preference learning as a suitable technique to exploit this structure. To this end, we generalize calibrated label ranking, a technique for tackling multilabel classification in a pairwise fashion [5], to the case where we have multipartite instead of bipartite preference information. In particular, we show how the use of a calibration label, which indicates the separation between relevant and irrelevant labels in the predicted ranking, can be generalized to multiple such labels. As a result, we investigate and experimentally compare three different variations of this principled approach.

II. Preliminaries

We represent an instance or object as a vector $\mathbf{x}$ in a feature space $\mathbb{X}$. Each instance can be associated with a point $y_{\mathbf{x}}$ in the target space $\mathbb{Y}$. A training set is a finite set of tuples $(\mathbf{x}, y_{\mathbf{x}}) \in \mathbb{X} \times \mathbb{Y}$ drawn independently from an unknown probability distribution on $\mathbb{X} \times \mathbb{Y}$. The goal is to learn a classifier $H : \mathbb{X} \to \mathbb{Y}$ which predicts $y_{\mathbf{x}}$ for a given $\mathbf{x}$. We will denote the prediction of H with a circumflex, i.e. $\hat{y} = H(\mathbf{x})$. Depending on the form of $\mathbb{Y}$ we face different problems and assumptions. In the simplest case, binary classification, we have $\mathbb{Y} = \{0, 1\}$. Ordinal classification generalizes this problem by extending the value space to a discrete and ordered

TABLE I. Example of ratings of some movies according to the German TV guide TVSpielfilm.de

Movie title	'fun'	'action'	'sex'	'suspense'
The other guys	⋆ ⋆ ⋆	⋆⋆		
A few good men		⋆		⋆ ⋆ ⋆
Once upon a time in the west		⋆ ⋆ ⋆	⋆	⋆ ⋆ ⋆
Dirty dancing		⋆	⋆	

This is the authors' version of the paper, which appeared in *2014 IEEE International Conference on Data Mining (ICDM 2014)*, pp. 731–736, DOI: 10.1109/ICDM.2014.102. A longer version is available as technical report at http://www.ke.tu-darmstadt.de/publications/reports/tud-ke-2014-01.pdf. For the present version, the following applies:

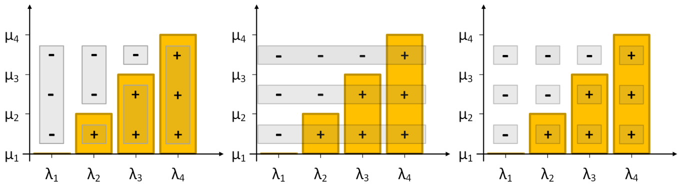


Fig. 1. Different decompositions of graded multilabel classification: vertical (*left*), horizontal (*center*), and complete (*right*). The illustration shows the decompositions for a training instance for which label λ_i has grade μ_i.

finite space $\mathbb{Y} = \mathbb{M} = \{\mu_1, \ldots, \mu_m\}$ that is structured with a total order $\prec$, such that $\mu_1 \prec \mu_2 \prec \ldots \prec \mu_m$. On the other hand, multilabel classification extends the label space to n binary dimensions, i.e. $\mathbf{y}_{\mathbf{x}} = (y^1_{\mathbf{x}}, \ldots, y^n_{\mathbf{x}}) \in \mathbb{Y} = \{0,1\}^n$. Alternatively, we may view this as a mapping from $\mathbf{x}$ to a subset $P_{\mathbf{x}} \subseteq \mathcal{L}$, where $\mathcal{L}$ is a finite set of predefined, non-mutually exclusive labels $\{\lambda_1, \ldots, \lambda_n\}$. y_i is 1 if $\lambda_i \in P_{\mathbf{x}}$ and 0 otherwise. The labels in $P_{\mathbf{x}}$ are usually said to be *relevant* or *positive*, whereas $\mathcal{L}$. $N_{\mathbf{x}} = \mathbb{L} \backslash P_{\mathbf{x}}$ is called the set of *irrelevant* or *negative* labels for $\mathbf{x}$.

III. Graded Multilabel Classification

In graded multilabel classification [2], each label λ in the set of relevant labels $P_{\mathbf{x}}$ of instance $\mathbf{x} \in \mathbb{X}$ is no longer only relevant or not ($\mathbb{M} = \{0,1\}$), but has output values $\mathbb{M} = \{\mu_1, \ldots, \mu_m\}$ with an ordered scale $\mu_1 \prec \mu_2 \prec \ldots \prec \mu_m$ as in ordered classification. It is assumed that the same ordinal scale is used for all labels, i.e. $\mathbb{Y} = \{\mu_1, \ldots, \mu_m\}^n$, μ_1 denoting the lowest and μ_n the highest degree of relevance of a label. This is a strong restriction but is motivated on real applications such as those sketched in the introduction. On the other hand, this assumption induces a (limited) comparability between the grades of the different labels which cannot be assumed in the more general setting of multi-target ordinal regression.

Following [2], we define the auxiliary membership function $L_{\mathbf{x}} : \mathbb{L} \to \mathbb{M}$ as $L_{\mathbf{x}}(\lambda_i) = y^i_{\mathbf{x}}$ which returns the grade of a specific label and instance. Let $P'^i_{\mathbf{x}} = \{\lambda \mid \mu_i = L_{\mathbf{x}}(\lambda)\}$ be the set of labels with grade μ_i, and $P^i_{\mathbf{x}} = \{\lambda \mid \mu_i \preceq L_{\mathbf{x}}(\lambda)\}$ the labels that are at least as relevant as μ_i. The latter set allows to model the assumption that if a label has a membership degree of μ_i, it also has all grades $\mu_j \prec \mu_i$ associated to it. Thus, since μ_1 is the lowest possible grade, it follows that $P^1_{\mathbf{x}} = \mathbb{L}$.

Cheng et al. [2] introduce three straight-forward *reduction* schemes in order to decompose the original problem into a set of well-known and solvable subproblems. Figure 1 illustrates these reductions on an example where we have four possible labels $\mathbb{L} = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$ on a scale $\mu_1 \prec \mu_2 \prec \mu_3 \prec \mu_4$.

Vertical Reduction: In the vertical reduction, the original problem of learning $H \colon \mathbb{X} \to \mathbb{M}^n$ is reduced to n ordinal classification problems of learning $[H]_{\lambda_1}, \ldots, [H]_{\lambda_n}$, $[H]_{\lambda_i} : \mathbb{X} \to \mathbb{M}$, one for each label $\lambda_1, \ldots, \lambda_n$ (cf. Figure 1 (left)). The aggregation of the individual predictions is trivially given by $H(\mathbf{x}) = ([H]_{\lambda_1}(\mathbf{x}), \ldots, [H]_{\lambda_n(\mathbf{x})})$. A simple yet effective decomposition strategy for solving the individual resulting ordinal problems was proposed by Frank and Hall [6]: the original problem is decomposed into $n-1$ independent binary subproblems, each of which contains all instances with a class value $\prec \mu_i$ as positive examples and all others as negative examples. The probabilistic estimations of the base classifiers are then combined into a distribution $Pr(\mu_i) = Pr(\prec \mu_{i+i}) - Pr(\prec \mu_i)$ over the possible class grades. Obviously, such independent classifiers can not model interdependencies and correlations between the different labels, which is the main disadvantage of this approach.

Horizontal Reduction: In contrast, the horizontal reduction transforms the original problem into $m = |\mathbb{M}|$ multilabel classification problems. For each grade μ_i, $i = 1 \ldots m$ we learn a classifier $[H]^i : \mathbb{X} \to \mathcal{P}(\mathcal{L})$ using $(\mathbf{x}, P^i_{\mathbf{x}})$ as training information. As $P^1_{\mathbf{x}} = \mathbb{L}$, we can ignore grade μ_1.

An additional challenge for this approach is that it cannot be guaranteed that $[H]^j(\mathbf{x}) = \hat{P}^j_{\mathbf{x}} \subseteq [H]^i(\mathbf{x}) = \hat{P}^i_{\mathbf{x}}$, $\mu_j \prec \mu_i$ holds, although by definition it holds that $P^j_{\mathbf{x}} \subseteq P^i_{\mathbf{x}}$ for $\mu_j \prec \mu_i$. Cheng et al. attempt to address this problem by weighting the evidence for a higher grade higher than the evidence for a lower grade, effectively proposing to resolve contradictions by taking for each label λ_i the maximum predicted grade $\max_{\prec}\{\mu_j \in \mathbb{M} \mid \lambda_i \in \hat{P}^j_{\mathbf{x}}\}$, where $\max$ is defined with respect to the total order relation $\prec$.

Unlike the vertical scheme, the horizontal reduction scheme conserves dependencies between labels because each multilabel subproblem allows to model the label dependencies at a certain degree of membership. This information can be taken into account by algorithms like IBLR-ML [2].

Complete Reduction: The complete reduction learns a single classifier $[H]_{\lambda_i \mu_j} : \mathbb{X} \to \{0,1\}$ for each of the $n \cdot m$ possible label/grade combinations using training information $(\mathbf{x}, \mathbb{I}(\mu_j \preceq y^i_{\mathbf{x}}))$ where $\mathbb{I}$ is the indicator function ($\mathbb{I}(x) = 1$ if x is true, and 0 otherwise).

IV. Graded Multilabel Classification by Pairwise Comparison

Learning by pairwise decompositions is based on the idea of modeling preferences between labels [7]. These preferences are either derived from the label structure (e.g. a hierarchy) or given for the training instances at hand, e.g. in the form of a total or partial, often multipartite ranking. Moreover, pairwise decomposition implicitly takes label dependencies into account to some extent, since it explicitly models the cases of pairwise exclusions. We hence believe that pairwise decomposition is well suited to the setting of graded multilabel classification. In particular, we build upon *calibrated label ranking* (CLR), a pairwise approach to solving multilabel problems, which we describe in more detail in the following. Thereafter, we will introduce three different approaches for generalizing CLR to the graded case, which are all based on the idea of working with multiple calibration labels.

A. Calibrated Label Ranking

The pairwise decomposition of multilabel problems interprets the training information as bipartite rankings $N_{\mathbf{x}} \prec P_{\mathbf{x}}$, i.e., we can deduce explicit preference statements $\lambda_u \prec \lambda_v$ for all $\lambda_u \in N_{\mathbf{x}}, \lambda_v \in P_{\mathbf{x}}$. These preferences are learned by training classifiers $H_{uv} : \mathbf{x} \to \{0,1\}$ for each of the possible pairs of labels, $1 \leq u < v \leq n$. Hence, the problem is decomposed into $\frac{n(n-1)}{2}$ smaller binary sub-problems. For each pair of labels (λ_u, λ_v), only examples belonging to either

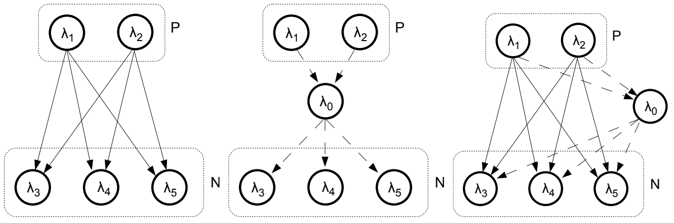


Fig. 2. Preferences in calibrated label ranking: on the left, we see all preferences between the relevant labels $P_{\mathbf{x}} = \{\lambda_1, \lambda_2\}$ and the irrelevant labels $N_{\mathbf{x}} = \{\lambda_3, \lambda_4, \lambda_5\}$, the center graph shows the position of the virtual label $v = \lambda_0$, and the right graph shows all generated preferences (the union of the previous two graphs).

λ_u or λ_v are used to train the corresponding classifier $H_{u,v}$. More precisely, classifier $H_{u,v}$ receives all $\mathbf{x}$ where λ_u is a relevant label and λ_v is irrelevant as positive training examples $(\mathbf{x}, 1)$, and those where $\lambda_v \in P_{\mathbf{x}}$ and $\lambda_u \in N_{\mathbf{x}}$ as negative examples $(\mathbf{x}, 0)$. All other examples are ignored. For making a prediction, all $\frac{n(n-1)}{2}$ base classifiers predict a vote for one of the two corresponding classes. Adding these votes results in a full ranking over the labels.

To convert the resulting ranking of labels into a multilabel prediction, we use the *calibrated label ranking* (CLR) approach [5]. This technique avoids the need for learning a threshold function for separating relevant from irrelevant labels, which is often performed as a post-processing phase after computing a ranking of all possible classes. The key idea is to introduce an artificial *calibration label* $v = \lambda_0$, which represents the split-point between relevant and irrelevant labels. Thus, v is assumed to be preferred over all irrelevant labels, but all relevant labels are preferred over v (cf. Figure 2).

During prediction, the virtual label is treated like any other label. Its position in the predicted ranking then denotes a natural cutting point for dividing the label ranking into two sets.[1]

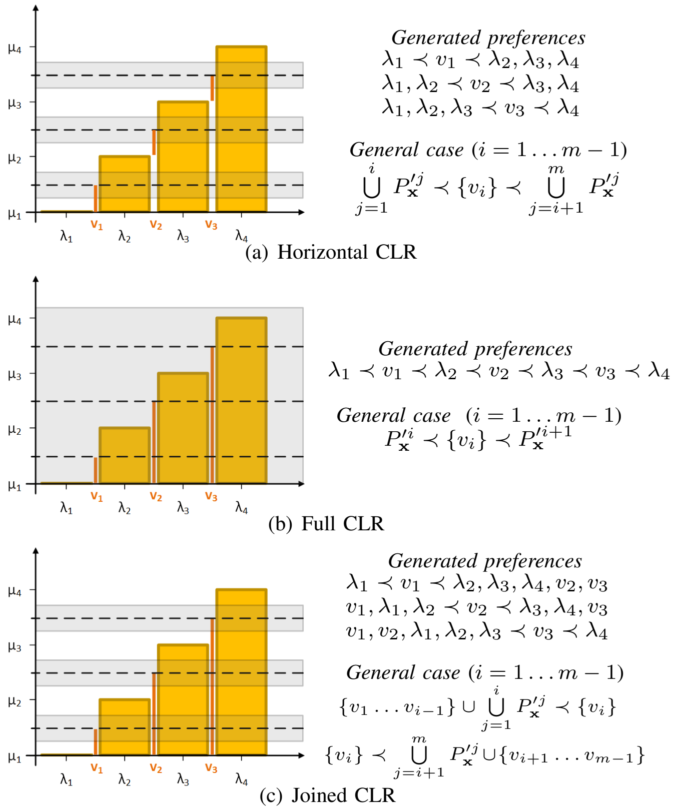
B. Multiple Calibration Labels

The key idea of the proposed pairwise approach to graded multilabel classification is to generalize calibrated label ranking to the case of multiple calibration labels $\mathbb{V} = \{v_1, \ldots, v_{m-1}\}$, where each label represents an intermediate grade v_i between the original grades μ_i and μ_{i+1}. Hence, we obtain $\mathbb{M}_v = \mathbb{M} \cup \mathbb{V}$ with the inner structure

$$\mu_1 \prec v_1 \prec \mu_2 \prec v_2 \prec \mu_3 \prec \ldots \prec v_{m-1} \prec \mu_m$$

As a consequence, we obtain an extended set of labels $\mathbb{L} \cup \mathbb{V}$. Note that we use $\mathbb{V}$ to denote both, labels and grades, which conveniently emphasizes the fixed mapping between grade and label v_i, i.e. it generally holds $L(v_i) = v_i$.

Furthermore, in order to cover the case that some training instances may be ignored by certain pairwise classifiers, we introduce the projection function $[p]_{rp} : \mathbf{x} \to \{0, 1, \varnothing\}$ which indicates to use a training example $\mathbf{x}$ either as positive (1), negative (0) example or not at all ($\varnothing$) for the given decomposition rp. Let us further also assume that the pairwise base classifiers are symmetric, i.e. $[H]_{\lambda_u,\lambda_v} = 1 - [H]_{\lambda_v,\lambda_u}$


μ4
μ3
μ2
μ1
λ1
v1
λ2
v2
λ3
v3
λ4
Generated preferences
$\lambda_1 \prec v_1 \prec \lambda_2, \lambda_3, \lambda_4$
$\lambda_1, \lambda_2 \prec v_2 \prec \lambda_3, \lambda_4$
$\lambda_1, \lambda_2, \lambda_3 \prec v_3 \prec \lambda_4$
General case $(i = 1 \ldots m-1)$
$\bigcup_{j=1}^{i} P'^{j}_{\mathbf{x}} \prec \{v_i\} \prec \bigcup_{j=i+1}^{m} P'^{j}_{\mathbf{x}}$
μ4
μ3
μ2
μ1
λ1
v1
λ2
v2
λ3
v3
λ4
Generated preferences
$\lambda_1 \prec v_1 \prec \lambda_2 \prec v_2 \prec \lambda_3 \prec v_3 \prec \lambda_4$
General case $(i = 1 \ldots m-1)$
$P'^{i}_{\mathbf{x}} \prec \{v_i\} \prec P'^{i+1}_{\mathbf{x}}$
μ4
μ3
μ2
μ1
λ1
v1
λ2
v2
λ3
v3
λ4
Generated preferences
$\lambda_1 \prec v_1 \prec \lambda_2, \lambda_3, \lambda_4, v_2, v_3$
$v_1, \lambda_1, \lambda_2 \prec v_2 \prec \lambda_3, \lambda_4, v_3$
$v_1, v_2, \lambda_1, \lambda_2, \lambda_3 \prec v_3 \prec \lambda_4$
General case $(i = 1 \ldots m-1)$
$\{v_1 \ldots v_{i-1}\} \cup \bigcup_{j=1}^{i} P'^{j}_{\mathbf{x}} \prec \{v_i\}$
$\{v_i\} \prec \bigcup_{j=i+1}^{m} P'^{j}_{\mathbf{x}} \cup \{v_{i+1} \ldots v_{m-1}\}$

(a) Horizontal CLR

(b) Full CLR

(c) Joined CLR

Fig. 3. The three different approaches for a pairwise decomposition of a graded multilabel problem, showing also exemplarily the generated preferences and the general case $(i = 1 \ldots m - 1)$.

C. Horizontal Calibrated Label Ranking

The first, simple approach to generalize calibrated label ranking to the graded case is to use the horizontal decomposition as described in Section III, and to solve each of the resulting multilabel problems with CLR. Thus, in order to learn each $[H]^i$, we choose grade v_i as our cutting point, i.e. we only differentiate between grades greater or smaller than v_i. Translated to CLR, v_i becomes the calibrating label and $\cup_{v_i \prec \mu_j} P'^{j}_{\mathbf{x}}$ and $\cup_{\mu_i \prec v_j} P'^{j}_{\mathbf{x}}$ our positive and negative set of labels, respectively, as is illustrated in Figure 3(a).

More precisely, we train each $[H]^i_{\lambda_u,\lambda_v}, \ \lambda_u \neq \lambda_v \in \mathbb{L} \cup \{v_i\}$ using training examples $(\mathbf{x}, [p]^i_{\lambda_u,\lambda_v}(\mathbf{x}))$ given by

$$[p]^i_{\lambda_u,\lambda_v}(\mathbf{x}) = \begin{cases} 1 & \text{if } [L]^i_{\mathbf{x}}(\lambda_v) \prec [L]^i_{\mathbf{x}}(\lambda_u) \\ 0 & \text{if } [L]^i_{\mathbf{x}}(\lambda_u) \prec [L]^i_{\mathbf{x}}(\lambda_v) \\ \varnothing & \text{if } [L]^i_{\mathbf{x}}(\lambda_u) = [L]^i_{\mathbf{x}}(\lambda_v) \end{cases} \tag{1}$$

and

$$[L]^i(\lambda_u) = \begin{cases} \mu_i & \text{if } \lambda_u \prec v_u \\ \mu_{i+1} & \text{if } v_u \prec \lambda_u \end{cases} \tag{2}$$

For making a prediction for a test instance $\mathbf{x}$, the votes $h_{\mathbf{x}}(\lambda_u) = \sum_{\lambda_u \neq \lambda_v} [H]^i_{\lambda_u,\lambda_v}(\mathbf{x})$ are summed up for each label $u \in \mathbb{L} \cup \{v_i\}$, and λ_u is predicted as relevant if $h_{\mathbf{x}}(\lambda_u) > h_{\mathbf{x}}(\lambda_{v_i})$. The final graded prediction is obtained by using the maximum predicted score for each label, as described in Section III.

[1] We break ties in the final counting in favor of the virtual label.

D. Full Calibrated Label Ranking

The idea of the full calibrated label ranking approach is to consider the targets in a GMLC problem as a multipartite ranking $P_{\mathbf{x}}'^{1} \prec P_{\mathbf{x}}'^{2} \ldots \prec P_{\mathbf{x}}'^{m}$ (cf. Figure 3(b)). Enriched by the virtual labels we eventually obtain

$$P_{\mathbf{x}}'^{1} \prec \{v_1\} \prec P_{\mathbf{x}}'^{2} \ldots \prec \{v_{m-1}\} \prec P_{\mathbf{x}}'^{m}$$

Obviously, for $m = 2$, this reduces to calibrated label ranking with $P_{\mathbf{x}} = P_{\mathbf{x}}'^{1}$ and $N_{\mathbf{x}} = P_{\mathbf{x}}'^{2}$.

The projection function for base classifiers $[H]_{\lambda_u,\lambda_v}$, $\lambda_u \neq \lambda_v$, $\lambda_u, \lambda_v \in \mathbb{L} \cup \mathbb{V}$ only slightly changes in comparison to (1), namely into

$$[p]_{\lambda_u,\lambda_v}(\mathbf{x}) = \begin{cases} 1 & \text{if } L(\lambda_v) \prec L(\lambda_u) \\ 0 & \text{if } L(\lambda_u) \prec L(\lambda_v) \\ \varnothing & \text{if } L(\lambda_u) = L(\lambda_v) \end{cases} \tag{3}$$

Note that in contrast to the horizontal decomposition in Sec. IV-C we can sum up the votes across the grades, obtaining one global ranking over all labels and grades. After querying all $(n+m-1)(n+m-2)/2$ base classifiers, we then predict $\hat{y}^j = \arg\max_{\mu_i} h_{\mathbf{x}}(\lambda_j) > h_{\mathbf{x}}(\lambda_{v_i})$ for λ_j.

A possible disadvantage of this approach is that the algorithm is prone to producing many ties in the ranking since $n+m-1$ labels have to be ordered on a scale of 0 to $n+m-2$ obtainable votes. This can potentially be remedied using a different voting function like weighted voting. However, we observed that predicting accurate and comparable scores such as confidences or probabilities is not a trivial task. Hence, 0-1 voting is more robust and makes the fewest assumptions on the base classifiers, and we restrict ourselves to this approach in this paper. Another, related problem is that *preference intensities* are not considered, i.e., the difference between the grades of two compared labels is ignored, for training as well as during prediction. The Joined CLR approach, described in the next section, provides a solution to this.

E. Joined Calibrated Label Ranking

On the one hand, Full CLR is not able to capture different degrees of preference intensities since the preference between two labels λ_u, λ_v is only obtained in a binary way. On the other hand, we recall that in the horizontal approach we learn each discriminating classifier $[H]^i_{\lambda_u,\lambda_v}$ exactly $m-1$ times, once for every grade transition. In fact, the number of classifiers λ_u vs. λ_v which use a training instance $\mathbf{x}$ depends on the difference between the grades of the labels, more precisely, it is exactly $|y^u_{\mathbf{x}} - y^v_{\mathbf{x}}|$. We can hence expect that the difference in the number of votes between both labels correlates with the difference in the true grades. A solution, which takes such predictions with varying intensity into account, would be to compute a common, joint ranking across degrees and labels, i.e. to compute $s(\lambda_u) = \sum_{\mu_i} \sum_{\lambda_v \neq \lambda_u} [H]^i_{\lambda_u,\lambda_v}$ for all $\lambda_u, \lambda_v \in \mathbb{L} \cup \mathbb{V}$. Although this would possibly produce a good ranking over the labels in $\mathbb{L}$, it cannot be expected to provide a good ranking over the virtual labels because each of the virtual labels only appears in one horizontal sub-problem and can therefore only obtain at most n votes. In contrast, each of the real labels can obtain up to $n(m-1)$ votes.

Joined CLR solves this problem by generalizing the horizontal decomposition introduced above, so that all virtual labels are always used in all horizontal sub-problems. More precisely, it decomposes the initial problem into $m-1$ bipartite (three-partite if we count the virtual label) ranking problems with one main calibrating label v_i on each grade transition. In this regard, joined CLR is equivalent to horizontal calibrated label ranking and all pairwise base classifiers learned by horizontal CLR are also learned in exactly the same manner by joined CLR. On the other hand, as shown in Figure 3(c), joined CLR also adds all remaining virtual labels $v_j \neq v_i$ into these bipartite ranking problems allowing them to accumulate the necessary voting mass. The resulting problem remains bipartite, since we map all grades to μ_i and $\mu_i + 1$ as in horizontal CLR. Using a simplified informal representation, this basically means that in addition to the comparisons

$$\mu_1, \ldots, \mu_i \prec v_i \prec \mu_{i+1}, \ldots, \mu_m$$

each horizontal subproblems is enriched with the following preferences:

$$\begin{gathered} \mu_1, \ldots, \mu_i \prec v_{i+1}, \ldots, v_{m-1} \\ v_1, \ldots, v_{i-1} \prec \mu_{i+1}, \ldots, \mu_m \\ v_1, \ldots, v_{i-1} \prec v_i \prec v_{i+1}, \ldots, v_{m-1} \end{gathered}$$

More formally, we learn classifiers $[H]^i_{\lambda_u,\lambda_v}$ using $[p]$ and $[L]$ from Eq. (1) and (2), but in this case for each $\lambda_u \neq \lambda_v$, $\lambda_u, \lambda_v \in \mathbb{L} \cup \mathbb{V}$. Note that the training signal between two virtual labels is always fixed. Hence, we can set $[H]^i_{v_u,v_v}(\mathbf{x}) = 0$ if $v_u \prec v_v$, 1 otherwise, for $v_u \neq v_v$, $v_u, v_v \in \mathbb{V}$. During prediction, the votes for each label are aggregated across all grade transitions as proposed in the beginning of this subsection.

Note that fixing the predictions between virtual labels can introduce a bias since these predictions are always perfect, whereas the remaining predictions depend on the classification performance of a classifier trained on potentially noisy data. This problem can be alleviated e.g. by allowing different fixed values than 0 and 1 or by removing some comparisons. We are currently developing such methods and leave the investigation for further work.

V. Experiments

In this section, we describe the data and setup of the experiments, followed by the results.

A. Datasets

An overview over the used datasets is given in Table II. The BeLa-E benchmark was used in previous work, whereas Movies and Medical are two new real-world datasets.[2]

BeLa-E: The BeLa-E dataset results from a questionnaire in which 1930 students rated the importance of certain properties of their future jobs from ‘1’ to ‘5’. We replicated the setup of Cheng et al. [2] by choosing a random subset of the n questions as target labels and the remaining $50-n$ as instance attributes. The selection was done for $n = 5$ and $n = 10$, and in each case repeated 50 times, resulting in 50 different dataset for each value of n.

[2] Datasets and details at www.ke.tu-darmstadt.de/resources/GMLC and [4].

TABLE II. OVERVIEW OF DATASETS USED IN THE EXPERIMENTS. SHOWN ARE THE TOTAL NUMBER OF INSTANCES, ATTRIBUTES, UNIQUE LABELS n, DIFFERENT GRADES m, THE AVERAGE GRADE INDEX AND THE FREQUENCY OF THE SPECIFIC GRADES μ_i APPEARING IN THE LABEL–INSTANCE MAPPINGS.

					Avg.	Distribution of grades $\mu_i, i =$				
Dataset	Instances	Attributes	Labels	Grades	Grade	1	2	3	4	5
BELA-E n=5	1930	45	5	5	2.50	7.95	13.04	23.89	31.43	23.69
BELA-E n=10	1930	40	10	5	2.50	7.95	13.04	23.89	31.43	23.69
MOVIES	1967	27002	5	4	0.72	50.26	31.13	15.18	3.43	–
MEDICAL	1953	1602	204	4	0.02	99.08	0.31	0.24	0.37	–

Movies: We collected a dataset from the German TV program guide www.TVSpielfilm.de which rates movies by assigning grades from '0' to '3' to the categories 'fun', 'action', 'sex', 'suspense' and 'sophistication' rather than giving an overall rating. For characterizing the 1967 movies, we extracted the titles, the associated summary texts and other information from www.imdb.org and applied stemming, stop word removal and TF-IDF weighting.

Medical: The MEDICAL dataset consists of 1953 free text radiology reports. Three expert companies were asked to annotate them with a set of ICD-9-CM diagnosis codes. In contrast to the original multilabel dataset, we generated a GMLC dataset by considering the level of agreement as grade of assignment. Note that it lies in the nature of the problem that labels are very likely to be absent. The texts were processed as for MOVIES but we used the absolute term frequency in contrast to TF-IDF.

B. Experimental setup

All proposed approaches were implemented as part of the LPCforSOS framework,which is an extension of Weka,[3] except for IBLR-ML, which we obtained from the authors. IBLR-ML, a combination of instance-based learning with logistic regression taking into account label dependencies, is a state-of-the-art multilabel learner proposed by Cheng et al. [2] for solving the horizontal decomposition. We used J48/C4.5 of Weka as binary base classifier. The complete reduction approach was implemented by using horizontal reduction with binary relevance decomposition (BR). For ordinal classification in the vertical decomposition (F&H), we used Weka's implementation of the method of Frank and Hall [6]. All losses (see below) are computed individually on the instances, averaged first over all examples in a test fold, and then over all 10 test folds. In addition, on the BELA-E datasets, we averaged the results over the 50 versions of each dataset. For calculating the rank losses for the complete reduction approaches (BR and F&H), IBLR-ML and horizontal calibrated label ranking (H-CLR), the predicted grade is used as the score.

C. Losses

For the GMLC problem, Cheng et al. [2] generalized several common losses for multilabel classification, including *Hamming Loss* (avg. deviation from correct grades), *Vertical 0-1 Loss* (percentage of labels with incorrectly assigned grades) and the ranking measure *C-Index* (pairwise ranking error). We follow this setup, except for the following slight modification of the *One Error*. In addition, we propose *optimistic Hamming loss* as a new loss function.

One Error Rank Loss: This metric is the generalization of the *one error* loss for rankings in multilabel classification. It compares if the highest real grade corresponds to the highest predicted grade. Contrary to [2], we use a version that can be zero:

$$\text{ONEERR}\left(\hat{y}^i_{\mathbf{x}}, y^i_{\mathbf{x}}\right) = \frac{1}{m-1} AE\left(\max_{1\leq i\leq n} \hat{y}^i_{\mathbf{x}}, \max_{1\leq j\leq n} y^j_{\mathbf{x}}\right)$$

with $AE : \mathbb{M} \times \mathbb{M} \to \mathbb{N}, AE\left(\mu_i, \mu_j\right) = |i - j|$.

Optimistic Hamming Loss: Under some circumstances, CLR tends to under- or overestimate the correct position of the virtual label. In order to be independent of such an effect, we follow the idea of [5] and propose to evaluate the ranking performance by *cheating* on the correct positioning of the virtual labels: we place the cutting points in hindsight so that the distribution of grades corresponds to the real one. In a way, this allows us to compute the *regret* of using a specific cutting technique. More precisely, we find the partitioning $\hat{P}''^1_{\mathbf{x}}, \hat{P}''^2_{\mathbf{x}}, \ldots$ over the predicted ranking such that $|\hat{P}''^i_{\mathbf{x}}| = |P'^i_{\mathbf{x}}|$ and $s_{\mathbf{x}}(\lambda_u) \leq s_{\mathbf{x}}(\lambda_v)$ if $\lambda_u \in P'^i_{\mathbf{x}}, \lambda_v \in P'^j_{\mathbf{x}}, \mu_i \prec \mu_j$. Given the corresponing prediction $\hat{\mathbf{y}}''$, we obtain the optimistic Hamming loss as $\text{OPTHAMMLOSS} = \text{HAMMLOSS}\left(\hat{\mathbf{y}}''_{\mathbf{x}}, \mathbf{y}_{\mathbf{x}}\right)$.

D. Results

The experimental results are summarized in Table III. The first observation is that BR, i.e., the complete reduction using horizontal and vertical cuts, is usually outperformed by the pairwise approaches, even for Hamming loss. Moreover, BR is always outperformed by F&H, even though both classifiers are trained equally. The difference is due to the different aggregation strategies of the predictions of the binary classifiers (see Sec. III), and obviously, the more sophisticated approach by Frank and Hall pays off for these datasets.

The next observation is that the approach using IBLR-ML shows even worse results than BR. This is surprising, since it does not correspond to the results reported by Cheng et al. [2], where BR is beaten by IBLR-ML, although we used the code provided by the authors. A reason might be that the 50 sub-datasets are obviously not exactly equal due to the random initialization. Furthermore, we used a different base learner for BR which explains the differences for this algorithm, but not the ones for IBLR-ML, which was used exactly the same way as in Cheng et al. [2].

Still, our results for C-Index and one error seem more reasonable to us since IBLR-ML uses the same overestimating aggregation as BR. H-CLR also uses this aggregation but pairwise classification is an ensemble method and thereby is more robust to noise predictions of single classifiers.

Interestingly, the approach using vertical reduction (F&H) seems to perform quite competitive w.r.t. other approaches, especially for Hamming and vertical 0-1 loss. This may show that preserving and focusing on the information about the grades (vertical) is more important for GMLC than considering the relations between the labels at each grade (horizontal). On the other hand, horizontal CLR outperforms F&H on exactly these both losses (except for MEDICAL, where they perform equally). On the BELA-E datasets, all approaches are pairwise statistically significantly different with $\alpha = 0.01$ (sign test).

[3] Cf. http://www.lpcforsos.sf.net and http://www.cs.waikato.ac.nz/ml/weka/.

TABLE III. Results of the three pairwise graded multilabel algorithms in comparison to IBLR-ML and two benchmarks. In addition to the results of the five different loss functions in terms of percentage (×100), we show the standard deviation for BeLa-E and the average rank of each algorithm on the particular dataset in parenthesis.

Dataset	Evaluation Measure	IBLR-ML	BR	F&H	Full CLR	Joined CLR	Horizontal CLR
BeLa-E $n = 5$	Hamming Loss	27.23 (4)± 4.51	28.07 (5)± 2.62	16.08 (2)± 1.65	33.97 (6)± 5.79	17.96 (3)± 1.31	**15.77 (1)**± 1.53
	Optimistic Hamming Loss	–	–	–	11.00 (2)± 1.70	**9.62 (1)**± 1.45	–
	Vertical 0-1 Loss	69.39 (5)± 5.39	61.27 (3)± 4.19	51.97 (2)± 3.68	73.44 (6)± 7.58	61.82 (4)± 3.61	**51.90 (1)**± 3.52
	C-Index	49.55 (6)± 8.44	32.63 (5)± 3.19	24.34 (4)± 4.25	20.38 (2)± 4.13	**18.16 (1)**± 3.68	23.88 (3)± 4.11
	One Error Loss	27.80 (6)± 7.46	12.89 (5)± 3.20	11.35 (4)± 2.45	8.50 (2)± 2.25	**7.19 (1)**± 1.82	11.06 (3)± 2.31
BeLa-E $n = 10$	Hamming Loss	27.27 (4)± 3.83	27.77 (5)± 1.83	16.04 (2)± 1.04	35.44 (6)± 3.70	17.92 (3)± 0.87	**15.13 (1)**± 0.95
	Optimistic Hamming Loss	–	–	–	12.70 (2)± 0.94	**12.03 (1)**± 0.91	–
	Vertical 0-1 Loss	69.95 (5)± 4.16	61.17 (3)± 2.69	51.97 (2)± 2.23	75.11 (6)± 4.47	61.76 (4)± 0.87	**50.45 (1)**± 2.15
	C-Index	50.37 (6)± 6.98	32.85 (5)± 3.45	24.14 (4)± 2.68	18.57 (2)± 2.27	**17.58 (1)**± 2.14	22.78 (3)± 2.53
	One Error Loss	34.47 (6)± 9.23	17.03 (5)± 4.38	12.92 (4)± 2.53	8.19 (2)± 1.67	**7.77 (1)**± 1.28	11.56 (3)± 1.93
Movies	Hamming Loss	32.33 (5)	21.94 (3)	18.95 (2)	76.51 (6)	25.32 (4)	**17.73 (1)**
	Optimistic Hamming Loss	–	–	–	9.58 (2)	**8.98 (1)**	–
	Vertical 0-1 Loss	67.34 (5)	50.85 (3)	47.86 (2)	96.50 (6)	67.16 (4)	**44.70 (1)**
	C-Index	33.98 (6)	30.86 (5)	23.12 (4)	15.43 (2)	**14.74 (1)**	21.40 (3)
	One Error Loss	15.43 (5)	18.43 (6)	14.24 (4)	9.30 (2)	**7.75 (1)**	12.21 (3)
Medical	Hamming Loss	1.30 (3)	0.31 (2)	**0.26 (1)**	3.00 (4)	10.34 (5)	**0.26 (1)**
	Optimistic Hamming Loss	–	–	–	**0.23 (1)**	0.31 (2)	–
	Vertical 0-1 Loss	2.07 (3)	0.62 (2)	**0.60 (1)**	3.81 (4)	21.87 (5)	**0.60 (1)**
	C-Index	49.96 (6)	18.40 (5)	10.73 (3)	**3.27 (1)**	5.20 (2)	12.06 (4)
	One Error Loss	90.89 (6)	20.93 (5)	11.76 (3)	**10.44 (1)**	10.65 (2)	12.71 (4)

The results of the different calibrated label ranking approaches show a high correspondence to their inner structure. Full CLR shows the highest Hamming and vertical 0-1 loss among the approaches. When looking at its optimistic Hamming loss and the quite good C-Index and one error, this seems to be clearly just a problem of the correct positioning of the virtual labels due to the narrowness and thus ties in the rankings (see IV-D). Joined CLR shows a similar behavior. On all but one dataset, it has the best results among the approach for all three ranking losses. The somewhat worse results on the medical dataset suggest that Joined CLR has problems on datasets with many labels being assigned too extreme low or high grades (see Tab. II).

As already mentioned, Horizontal CLR outperforms all other approaches w.r.t. Hamming and vertical 0-1 loss. This is very likely due to the easier positioning of the single calibrating label, especially in comparison to full CLR but also to Joined CLR. On the other hand, Horizontal CLR reveals its disadvantages regarding the prediction of good rankings. It is the worst approach compared to the other pairwise methods w.r.t. C-Index and one error. It seems very obvious that the aggregation strategy of selecting the highest seen grade for each label, also used by BR and IBLR-ML and proposed by Cheng et al., is not advantageous w.r.t. ranking quality.

In summary, the pairwise approaches generally outperform all other approaches on the used ranking losses. Especially the full and joined decompositions provide a clear advantage when good label rankings are important. On the other hand, if we desire good predictions for each label independently (hence for each ordinal problem separately), then Horizontal CLR is the most appropriate method among all evaluated techniques in our experiments.

These two main results make us confident that learning by pairwise comparisons has a natural access to the inner structure of GMLC problems. Moreover, it was shown that pairwise learning provides a flexible adaptation to different objectives by adjusting decomposition and aggregation. The very low optimistic Hamming losses of the CLR approaches additionally promise an even better result through finding a better way of positioning the virtual labels into the global ranking.

VI. Conclusions

In this work, we introduced pairwise comparisons for representing and learning graded multilabel classification (GMLC) problems, which are a combination of ordinal and multilabel classification problems, where each instance is associated with several different grades of relevance to multiple categories. To be able to solve such problems by learning from pairwise comparisons we generalized Calibrated Label Ranking to the case of multiple calibration labels in three different ways, and experimentally compared these approaches to previous work by Cheng et al. [2] on three different datasets. In these experiments, our approaches achieved the best results in all measured losses.

Nevertheless, we believe that we have not yet fully exploited the information that is inherent in GMLC problems. In particular, we believe that pairwise comparisons have the capacity to achieve even better results by improving the way the predicted ranking is separated into grades. In future work, we plan to investigate alternative aggregation strategies to the horizontal reduction, the use of different voting strategies like weighted voting, as well as novel approaches for introducing the virtual labels into the label rankings.

Acknowledgements: *This research has been partially funded by the German Science Foundation (DFG). We would like to thank Weiwei Cheng and Eyke Hüllermeier for fruitful discussions and making their data and algorithms available.*

References

[1] G. Tsoumakas, I. Katakis, and I. P. Vlahavas, "Mining multi-label data," in *Data Mining and Knowledge Discovery Handbook*. Springer, 2010, pp. 667–685.

[2] W. Cheng, K. Dembczyński, and E. Hüllermeier, "Graded multi-label classification: The ordinal case," in *Proceedings of the 27th International Conference on Machine Learning*, 2010.

[3] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender Systems: An Introduction*. Cambridge Univ. Press, 2010.

[4] C. Brinker, E. Loza Mencía, and J. Fürnkranz, "Graded multilabel classification by pairwise comparisons," TU Darmstadt, Tech.

Rep. TUD-KE-2014-01, 2014. [Online]. Available: http://www.ke.tu-darmstadt.de/publications/reports/tud-ke-2014-01.pdf

[5] J. Fürnkranz, E. Hüllermeier, E. Loza Mencía, and K. Brinker, “Multilabel classification via calibrated label ranking,” *Machine Learning*, vol. 73, no. 2, pp. 133–153, Jun. 2008.

[6] E. Frank and M. Hall, “A simple approach to ordinal classification,” in *Proceedings of the 12th European Conference on Machine Learning (ECML-01)*, 2001, pp. 145–156.

[7] E. Hüllermeier, J. Fürnkranz, W. Cheng, and K. Brinker, “Label ranking by learning pairwise preferences,” *Artificial Intelligence*, vol. 172, no. 16-17, pp. 1897–1916, 2008.