Graded multi-label classification: compromise between handling label relations and limiting error propagation

Khalil Laghmari, Christophe Marsala, Mohammed Ramdani

Graded multi-label classification: compromise between handling label relations and limiting error propagation

Khalil Laghmari
Laboratoire Informatique de Mohammedia,
FSTM, Hassan II University of Casablanca,
BP 146 Mohammedia 20650 Maroc.
Sorbonne Universités,
UPMC Univ Paris 06,
CNRS, LIP6 UMR 7606,
4 place Jussieu 75005 Paris, France.
laghmari.khalil@gmail.com

Christophe Marsala
Sorbonne Universités,
UPMC Univ Paris 06,
CNRS, LIP6 UMR 7606,
4 place Jussieu 75005 Paris, France
christophe.marsala@lip6.fr

Mohammed Ramdani
Laboratoire Informatique de Mohammedia,
FSTM, Hassan II University of Casablanca,
BP 146 Mohammedia 20650 Maroc.
ramdani@fstm.ac.ma

***Abstract*—In graded multi-label classification (GMLC), each data can be assigned to multiple labels according to a degree of membership on an ordinal scale, and with respect to label relations. For example, in a movie catalog web page, a *five stars* action movie should be at least a *one star* suspense movie. Ignoring those relations can lead to inconsistent predictions, but if they are considered, then a prediction error for one label will be propagated to all related labels. Most of existing approaches either ignore label relations, or can learn only relations fitting a predefined imposed structure. This paper is motivated by the lack of a study analysing the compromise between handling label relations and limiting error propagation in GMLC, and by the fact that there is no known approach giving a control on that compromise to allow such a study. In this paper, a new meta-classifier with two main advantages is proposed for GMLC. Firstly, no predefined structure is imposed for learning label relations, and secondly, the meta-classifier is based on three measures giving control on the studied compromise. The studied compromise is analysed according to its impact on the classifier complexity and on hamming-loss evaluation measure. A comparison to three existing approaches shows that the proposed meta-classifier is competitive according to hamming-loss evaluation measure, and it is the most stable classifier according to hamming-loss standard deviation.**

I. Introduction

Graded multi-label classification (GMLC) is the task of assigning one or more labels to each data according to an ordinal scale of membership degrees M. It was recently introduced [1] as a generalization of the multi-label classification task (MLC) [2].

The most known source for GMLC data is catalogue web pages, where movies or animes are assigned to different categories such as *action*, *suspense*, and *humour* using a *one-to-five* star rating. Data involving the task of GMLC can be found in many other domains such as chemistry, where molecules have multiple odours with different intensities ranging from *very weak* to *very strong*.

Relations can exist between labels according to M. For example, *high ranked* action movies should contain *at least little* suspense, and molecules having a *very strong* intensity of jasmine odour *can not* have the smell of musk simultaneously.

Those relations, if well learned, are supposed to give a better understanding of hidden knowledge in data, but actually they lead to a serious dilemma:

- On the one hand, considering label relations has the advantage of making consistent predictions, but it has the disadvantage of allowing error propagation because a prediction error for one label will be propagated to all related labels.
- On the other hand, learning a classifier while ignoring those relations has the advantage of making independent predictions which prevents error propagation, but it has the disadvantage of making inconsistent predictions since label relations are ignored.

Label relation dilemma is an inherent challenge in all MLC tasks and not only specific to GMLC. However, this challenge has not received enough attention in the literature: first MLC approaches assume label independence, and later approaches focused only on how to handle label relations.

Learning all label relations can lead to cyclic dependencies between labels. Most of existing MLC approaches avoid this problem by setting a non cyclic dependence structure, and then try to learn only relations fitting the predefined structure.

Another limitation of existing MLC approaches is that they can handle only co-occurrence relations, while GMLC data can encapsulate also order relations based on the ordinal scale of membership degrees.

To overcome the limitations of existing MLC approaches, and to answer the challenge of label relation dilemma in GMLC, we propose a new meta-classifier named *PSI-MC* with two main advantages: Firstly, it allows label relation learning from GMLC data without imposing a predefined

978-1-5090-5781-8/16/$31.00 2016 IEEE

structure, and secondly, based on three measures named *pre-selection*, *selection*, and *interest* measures, it allows controlling the compromise between learning label relations and limiting error propagation.

The paper is organized as follows: in Section II we review the most used MLC approaches, in Section III we review some well-known GMLC approaches, and in Section IV we describe the proposed meta-classifier PSI-MC. Experiments on real datasets are discussed in Section V. Conclusions and ideas for future works are presented in Section VI.

II. Multi-label classification

Let $C = \{c_l\}_{1 \leq l \leq k}$ be the set of labels, and $X = \{x_i\}_{1 \leq i \leq n}$ be the set of data. Each data x_i is a vector $(x_{ij})_{1 \leq j \leq p}$ where x_{ij} is the value corresponding to the j^{th} attribute. $y_i \subseteq C$ is the set of labels associated to x_i. The MLC task is to learn a classifier $H : X \rightarrow \mathcal{P}(C)$ mapping each data to the correct set of associated labels. The MLC task is answered either by extending a mono-label classifier, or by transforming multi-label data to mono-label data.

Many mono-label classifiers were adapted to handle multi-label data, such as the K-Nearest Neighbours algorithm [3] [4], the naive bayes algorithm [5], the Support Vector machines [6] [7], the decision trees [8] [9], and the neural networks [10] [11]. In this paper we are more interested in transformation methods [12] because they can be used with any mono-label classifier.

There are three families of learning approaches based on transformation methods: Learning one multi-classes classifier, learning a binary classifier for each label, and learning a binary classifier for each two different labels.

A. Learning one multi-classes classifier

Label power set (LP) is a straightforward approach considering each label set as a single distinct label. One disadvantage of this approach is the class imbalance problem encountered when some label sets are not frequent. The idea behind pruned problem transformation approach (PPT) [13] is to remove infrequent label sets in order to overcome the class imbalance problem, but this leads to an information loss due to label truncation. A pruned problem transformation with no information loss (PPT-n) [13] is an improvement of PPT. Instead of removing infrequent label sets, PPT-n divide them to frequent subsets of labels. A common disadvantage between LP, PPT, and PPT-n approaches is that they can not predict an unseen label set in the training set. This limitation is overcome by the PPT-ext approach [13].

B. Learning a binary classifier for each label

In binary relevance approach (BR), a binary classifier $H_l : X \rightarrow \{0, 1\}$ is learned for each label $c_l \in C$. The labels relevant to predict are given by $H(x) = \{c_l, H_l(x) = 1\}_{1 \leq l \leq k}$. This approach learns independent classifiers and therefore can not learn label relations. Classifier chains approach (CC) [14] overcome this limitation using a chained structure depending on a predefined order for labels. Each classifier H_l is allowed to use the prediction result of all previous classifiers $\{H_{l'}\}_{1 \leq l' < l}$ in order to make its own prediction. In classifier treillis approach (CT) [15] a directed graph is built instead of a chained structure. Label correlation is computed between each pair of labels, then classifiers are placed on nodes according to label correlation. Each classifier is allowed to use the prediction result of all classifiers corresponding to parent nodes.

C. Learning a binary classifier for each two different labels

In ranking by pairwise comparison approach (RPC) [16], a binary classifier $H_{ll'}$ is learned for each label pair $c_l \neq c_{l'}$. Each $H_{ll'}$ is used to predict whether a data is associated to the label c_l or $c_{l'}$. A label ranking is outputted using a majority vote aggregation but with no separation between relevant and irrelevant labels. This problem is solved by calibrated label ranking approach (CLR) [17]. The idea is to train k more classifiers using an additional label c_0. For each classifier $H_{l0}, l \in [1, k]$, data not associated with the label c_l is considered as associated with the additional label c_0. Labels predicted more times then c_0 are relevant labels, and the remaining are considered as irrelevant labels.

III. Graded multi-label classification

A. GMLC and fuzzy sets

GMLC data can be viewed as fuzzy sets [18]. The only difference is that in fuzzy sets a numeric scale is used for membership degrees $M = [0, 1]$, while in GMLC an ordinal scale is used $M = \{m_1 < \ldots < m_s\}$. The interest of GMLC is related to data acquisition because it is easier for annotators to give their opinions on an ordinal scale. Each data x_i is associated with a fuzzy set y_i where $\lambda_i : C \rightarrow M$ is the mapping function of x_i associating each label $c_l \in C$ to its membership grade.

A fuzzy set can be described by the vertical representation using the membership function, or by the horizontal representation using α-cuts. In a similar way, the GMLC task can be solved using either a vertical or an horizontal decomposition, or both [1].

B. Decomposing the GMLC problem

Using the vertical decomposition, k classifiers are trained, one for each label $c_l \in C$, so that each classifier $H_l : X \rightarrow M$ predicts the membership grade for the label c_l.

Using the horizontal decomposition, $s - 1$ classifiers are trained, one for each grade $m_g \in \{m_2, \ldots, m_s\}$, so that each classifier $H_g : X \rightarrow \mathcal{P}(C)$ predicts the set of associated labels according to a membership grade at least equals to m_g. There is no need to train a classifier for m_1 because the corresponding label set is $H_1(x) = C$.

If a label c_l is predicted by a classifier H_g, it is expected from all classifiers $\{H_{g'}\}_{1 \leq g' \leq g}$ to predict c_l as well. In regard to this hierarchical property, the trained classifiers are not completely independent, hence the horizontal decomposition in theory, can learn better label relations than the vertical decomposition. However, it is not guaranteed that the hierarchical property is satisfied by trained classifiers. An aggregation function is generally used to handle this case.

Combining both decompositions can be done starting first by the vertical decomposition, then applying an horizontal decomposition, or the opposite. Indeed, the task of each vertical classifier is an ordinal classification [19] [20] and can be solved horizontally using $s-1$ binary classifiers, and the task of each horizontal classifier can be solved vertically using the BR approach.

C. Solving the GMLC problem using a pairwise approach

The task of GMLC can be solved also using a pairwise approach like CLR. The three methods named Horizontal CLR, Full CLR, and Joined CLR are all based on the idea of using multiple calibration labels [21]. The key idea is to use $s-1$ virtual labels: $V = \{v_g\}_{1\leq g\leq s-1}$ with fixed membership grades: $\forall i \in [1,n] : m_g < \lambda_i(v_g) < m_{g+1}$. v_g is used to denote (for simplicity) both the virtual label and its corresponding membership grade: $\lambda_i(v_g) = v_g$.

For the Horizontal CLR approach, an horizontal decomposition is first performed, then each MLC task g from the obtained $s-1$ tasks is solved using CLR, with v_g as the cutting point between relevant and irrelevant labels.

For the Full CLR approach, a classifier is trained for each label pair in $C \cup V$. All virtual labels are considered as cutting points. The membership grade m_g is predicted for a label $c_l \in C$ if m_g is the highest membership grade for which c_l is predicted more times than v_g.

One drawback of Full CLR approach, is that for a classifier $H_{ll'}$, a data x_i is considered positive if $\lambda_i(c_l) > \lambda_i(c_{l'})$, and negative otherwise, regardless of the difference between membership grades.

Joined CLR answers this problem by combining both Horizontal CLR and Full CLR approaches. Indeed, each MLC task for the Horizontal CLR approach is solved using all virtual labels as cutting points, instead of using only one cutting point.

The three CLR based approaches discussed in this section are also used in experiment section (V), where our proposed meta-classifier is compared to them.

IV. A new meta-classifier for graded multi-label classification

A. Key ideas

The first key idea of the proposed meta-classifier is to learn an initial set of k multi-class classifiers $H^0 = \{H_l\}_{1\leq l\leq k}$, one for each label, where the training set for a classifier H_l includes membership grades of labels $\{c_{l'}\}_{l'\neq l}$ as descriptive numeric attributes, which means that the prediction by a classifier H_l can depend on the prediction result of other classifiers $\{H_{l'}\}_{l'\neq l}$. This allows learning label relations considering membership grades without fixing a predefined structure, but it does not prevent learning cyclic dependencies.

The second key idea is to avoid cyclic dependencies by replacing involved classifiers. The compromise between handling label relations and limiting error propagation is defined by the way we select which classifier to replace first, and by the way we select alternative classifiers that the new classifier can depend on. The final set of classifiers without cyclic dependencies is the one used to make predictions $\mathbb{H} = \{\mathbb{H}_L\}_{1\leq L\leq k}$.

Initially, the final classifier set is empty: $\mathbb{H} \leftarrow \emptyset$, and the set of classifiers not yet added to $\mathbb{H}$ is $H^0 - \mathbb{H} = H$. Our proposed meta-classifier PSI-MC removes iteratively classifiers from H, and adds them directly or after being replaced to $\mathbb{H}$, until $|H| = 0$ and consequently $|\mathbb{H}| = k$.

B. Measures

A **pre-selection measure** $\mathbb{P} : H \rightarrow \{0,1\}$
is used to fill the set of candidate classifiers for replacement: $\{H_l \in H, \mathbb{P}(H_l) = 1\}_{1\leq l\leq k}$. Note that our objective is not only to solve cyclic dependencies but also to answer the challenge of label relation dilemma. Hence, in order to reduce the effect of error propagation, even if a classifier is not involved in a cyclic dependency, it can be a candidate for replacement if it depends on too many other classifiers.

Classifiers not candidates for replacement are added directly to the final classifier set:
$\mathbb{H} \leftarrow \mathbb{H} \cup \{H_l \in H, \mathbb{P}(H_l) = 0\}_{1\leq l\leq k}$.

A **selection measure** $\mathbb{S} : \mathcal{P}(H) \rightarrow H$ is used to select one classifier to be replaced from the set of candidate classifiers. For example, it can be either the classifier that depends on the lowest number of classifiers to reduce information loss, or the classifier that depends on the highest number of classifiers to reduce error propagation. It could also be the classifier whose the highest number of classifiers depend on to resolve more cyclic dependencies.

A **measure of chaining interest** $\mathbb{I} : \mathbb{H} \rightarrow \{0,1\}$ is used to decide whether the new classifier to build can depend on a final classifier $\mathbb{H}_L$: $\mathbb{I}(\mathbb{H}_L) = 1$ or not: $\mathbb{I}(\mathbb{H}_L) = 0$.

pre-selection, selection, and chaining interest measures are called the **PSI-measures**. In the following, we investigate their impact on synthetic data before analysing experiment results on real datasets.

C. Analysing the impact of PSI-measures using synthetic data

In TABLE I, $X = \{x_i\}_{1\leq i\leq 10}$ is the training set, $C = \{c_l\}_{1\leq l\leq 5}$ is the label set, $M = \{m_g\}_{1\leq g\leq 4} = \{0,1,2,3\}$ is the set of membership grades, and $\{a_1, a_2\}$ is the set of descriptive attributes.

	a_1	a_2	c_1	c_2	c_3	c_4	c_5
x_1	20	20	0	0	3	0	0
x_2	30	40	1	0	3	0	0
x_3	20	30	0	0	3	0	0
x_4	20	10	0	0	0	0	3
x_5	50	40	2	3	0	1	2
x_6	50	20	2	3	0	1	2
x_7	10	10	0	1	2	2	3
x_8	10	30	0	3	1	2	2
x_9	10	10	0	1	2	2	3
x_{10}	10	50	0	3	1	2	2

TABLE I
GMLC DATA

In the following, we use decision trees as base classifiers because they are easily interpreted, however the proposed meta-classifier can be used with any other base classifier.

Fig. 1 shows the obtained decision trees for H^0 using the weka implementation [22] of the *C4.5* algorithm [23], and Fig. 2 shows the corresponding dependency graph.

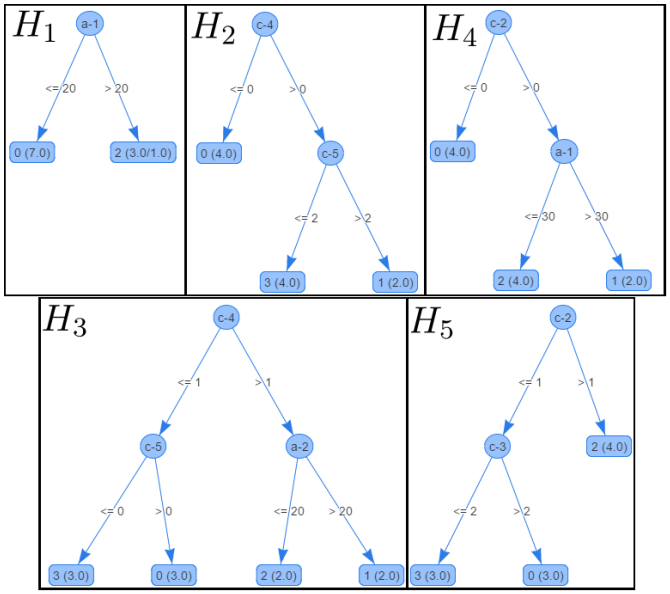

Fig. 1. Decision trees for H^0

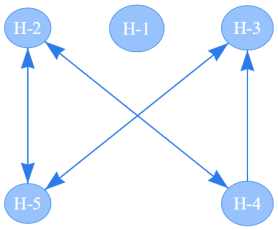

Fig. 2. Dependency graph for H^0

Let $D^{\rightarrow} : H \rightarrow \mathcal{P}(H)$ be the function giving for each classifier H_l the set of classifiers depending on it, and $D^{\leftarrow} : H \rightarrow \mathcal{P}(H)$ be the function giving for each classifier H_l the set of classifiers that it depends on.

In this example, we choose to make all non independent classifiers candidates for replacement:
$\mathbb{P}(H_l) = 0$ *if* $D^{\leftarrow}(H_l) = \emptyset$, 1 *otherwise*.

H_1 is independent, hence it is removed from H and added directly to $\mathbb{H}$ (Fig. 3).

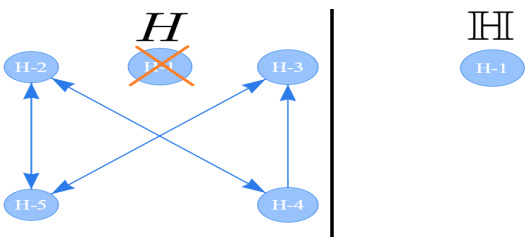

Fig. 3. Moving H_1 from H to $\mathbb{H}$

The remaining classifiers are all dependent. One of them should be selected according to a selection measure. In this example we choose to select the classifier allowing us to solve the highest number of cyclic dependencies:
$\mathbb{S}(H) = \underset{H_l \in H}{argmax}(|D^{\rightarrow}(H_l)|)$.

We have $|D^{\rightarrow}(H_3)| = 1 < |D^{\rightarrow}(H_2)| = |D^{\rightarrow}(H_4)| = |D^{\rightarrow}(H_5)| = 2$. The selection measure outputs the first classifier with the highest value which is H_2 in this case.

H_2 is to be replaced by another classifier H'_2, and we have the choice to chain it with $\mathbb{H}_1$ or to make it independent. In this example the chaining interest measure is given by $\mathbb{I}(\mathbb{H}_L) = 0$, $\forall \mathbb{H}_L \in \mathbb{H}$.

Fig. 4 shows the updated dependency graphs for H and $\mathbb{H}$ after replacing H_2 by H'_2.

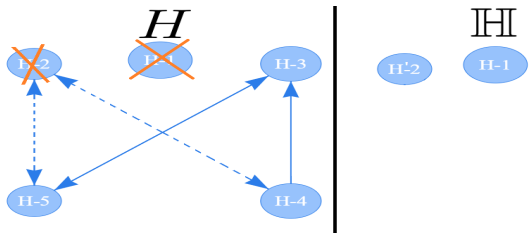

Fig. 4. Replacing H_2 by H'_2

Note that after removing H_2 from H, H_4 becomes independent and then it is added directly to $\mathbb{H}$ (Fig. 5). Also note that since H_4 was depending on H_2 in H, it is now depending on H'_2 in $\mathbb{H}$. This shows that the selection measure choosing which classifier to replace first, has also an impact on the learned label relations.

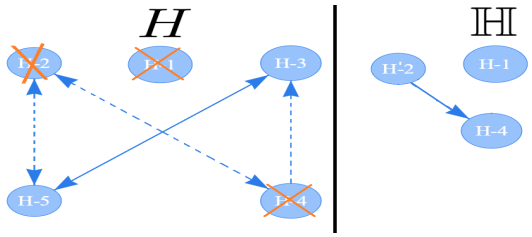

Fig. 5. Moving H_4 from H to $\mathbb{H}$

We have $D^{\leftarrow}(H_3) = D^{\leftarrow}(H_5) = 1$, hence $\mathbb{P}(H_3) = \mathbb{P}(H_5) = 1$. H_3 and H_5 are both candidates for replacement, and since we have $|D^{\rightarrow}(H_3)| = |D^{\rightarrow}(H_5)| = 1$, then $\mathbb{S}(H) = H_3$.

H_3 is replaced by an independent classifier H'_3 according to the chaining interest measure $\mathbb{I}$ (Fig. 6).

After removing H_3 from H, H_5 becomes independent and it is added directly to $\mathbb{H}$ (Fig. 7). Since H_5 was depending on both H_2 and H_3 in H, it is now depending on H'_2 and H'_3 in $\mathbb{H}$.

Fig. 8 shows the dependency graph for H^0, compared to the one for $\mathbb{H}$ using a chaining interest measure always equals to 0 (the same used in the example), and to the one for $\mathbb{H}$ using a chaining interest measure always equals to 1. Note

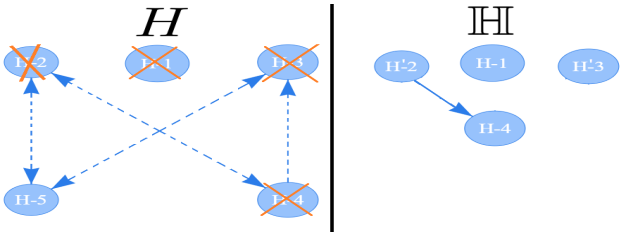

Fig. 6. Replacing H_3 by H'_3

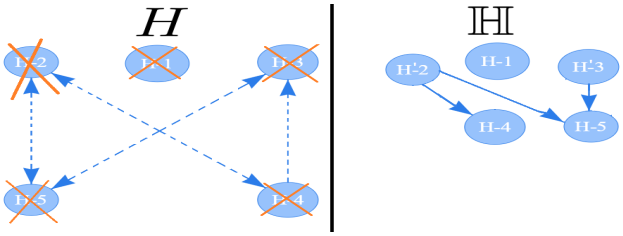

Fig. 7. Moving H_5 from H to $\mathbb{H}$

that with $\mathbb{I} = 1$ new label relations are learned instead of the initial cyclic relations.

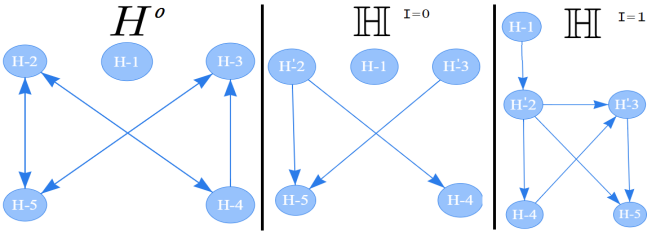

Fig. 8. Dependency graphs

In summary, by marking classifiers to be replaced, the pre-selection measure $\mathbb{P}$ can reduce learned label relations to minimize the risk of error propagation. The learned relation structure can change according to the order of replaced classifiers, which is determined by the selection measure $\mathbb{S}$. The chaining interest measure $\mathbb{I}$ controls allowed label relations: predictions are supposed to be more consistent by allowing replaced classifiers to learn many relations, and by doing the opposite the risk of error propagation is reduced.

V. Experiments

dataset	instances	attributes	labels	grades
BelaE-5	1930	45	5	$\{0,1,2,3,4\}$
BelaE-10	1930	40	10	$\{0,1,2,3,4\}$
molecules	2600	15	81	$\{0,1,2,3,4,5,6\}$

TABLE II
Description of used datasets

TABLE II describes datasets used in our experiments. The original BelaE data [1] is collected from the answers of 1930 graduate students about the importance of 48 properties of their future jobs. Descriptive attributes are age and sex of students, and labels are the 48 properties. To overcome the problem of insufficient attributes, 50 datasets are randomly generated considering only k labels as target labels and the remaining labels as descriptive attributes. In order to compare our proposed meta-classifier results to reported results of Horizontal CLR, Full CLR, and Joined CLR approaches [21], we used the same 50 datasets for $k = 5$, the same 50 datasets for $k = 10$, and the same base classifier (weka implementation of C4.5) as in [21]. Results averaged over 10 folds cross-validation are shown in TABLE III, where hamming-loss is extended to the GMLC case as described in [21].

Dataset	meta-classifier	hamming-loss average and standard deviation
BelaE-5	Full CLR	0.3397 ± 5.79
	Joined CLR	0.1796 ± 1.31
	Horizontal CLR	0.1577 ± 1.53
	PSI-MC ($I = 1$)	0.1891 ± 0.0956
	PSI-MC ($I = 0$)	0.1889 ± 0.0949
BelaE-10	Full CLR	0.3544 ± 3.70
	Joined CLR	0.1792 ± 0.87
	Horizontal CLR	0.1513 ± 0.95
	PSI-MC ($I = 1$)	0.1894 ± 0.0721
	PSI-MC ($I = 0$)	0.1884 ± 0.0709

TABLE III
Evaluation of graded multi-label classifiers using BelaE dataset

Our proposed meta-classifier is at least better than Full CLR according to hamming-loss (TABLE III), and almost as good as Joined CLR, while it is the most stable one according to hamming-loss standard deviation.

We generated 15 descriptive attributes for odorous molecule data [24] based on the name and formula of molecules. TABLE IV as TABLE III shows that there is no significant difference between using the measure $\mathbb{I} = 0$ and the measure $\mathbb{I} = 1$ according to hamming-loss. Indeed, by allowing more label relations to be learned ($\mathbb{I} = 1$) more true positive labels are predicted, hence the sensitivity (true positive rate) is increased and hamming-loss is decreased, but due to error propagation, more false positive labels are predicted and consequently hamming-loss is increased. This explains why using $\mathbb{I} = 1$ is not always better.

According to the low sensitivity obtained in TABLE IV, the 15 generated attributes are not relevant to discern labels (molecule odours). The prediction error rate in overlapping regions may be high. This problem can be answered by outputting for each data a set of possible labels for each membership grade [25], because experts prefer to have one true prediction in a small set of possible predictions instead of one prediction that could be totally wrong.

meta-classifier	hamming-loss average and standard deviation	sensitivity average and standard deviation
PSI-MC ($I = 1$)	0.0350 ± 0.0155	0.1923 ± 0.2736
PSI-MC ($I = 0$)	0.0306 ± 0.0128	0.1523 ± 0.2563

TABLE IV
The impact of chaining interest measure in molecule data

Label relation dilemma has also an impact on classifier complexity. TABLE V shows the average number of nodes, leafs, and dependent nodes over all decision trees in each meta-

[1] http://www.ke.tu-darmstadt.de/resources/GMLC

classifier, including the initial meta-classifier with cyclic dependencies H^0. When descriptive attributes are good enough to discern labels (BelaE-5 and BelaE-10), the classifier complexity (number of nodes and leafs) is reduced by learning more label relations (number of dependent nodes), but the opposite happens when descriptive attributes can not discern labels well enough (molecule data), because more dependent labels will be needed and consequently the complexity is increased.

Dataset	meta-classifier	node average	leaf average	dependent node average
BelaE-5	H^0	752	376	26
	PSI-MC(I=1)	757	379	13
	PSI-MC(I=0)	761	381	5
BelaE-10	H^0	748	375	58
	PSI-MC(I=1)	761	381	31
	PSI-MC(I=0)	772	386	6
molecules	H^0	115	61	36
	PSI-MC(I=1)	95	51	19
	PSI-MC(I=0)	46	25	4

TABLE V
THE IMPACT OF CHAINING INTEREST MEASURE ON CLASSIFIER COMPLEXITY

VI. Conclusions and future work

In this paper, we proposed a new meta-classifier for GMLC named PSI-MC. It is based on three measures named PSI-measures for pre-selection, selection and chaining interest measure. PSI-MC has two main advantages: it allows learning label relations without fixing a predefined relation structure, and it allows controlling the compromise between handling label relations and limiting error propagation. Experiment results on real datasets shows that PSI-MC is competitive with other existing approaches according to hamming-loss, and it is the most stable one according to hamming-loss standard deviation.

In this paper, the impact of PSI-measures is not fully analysed, there is many interesting measures to be studied, and there is even more if we use base classifiers outputting attribute weights such as neural networks. For future work, we plan to study further our proposed classifier, by analysing different PSI-measures, with different base classifiers and for different datasets.

References

[1] W. Cheng, K. Dembczynski, and E. Hllermeier., "Graded multilabel classification: The ordinal case," in *Proceedings of LWA2010 - Workshop-Woche: Lernen, Wissen & Adaptivitaet*, M. Atzmller, D. Benz, A. Hotho, and G. Stumme, Eds., Kassel, Germany, 2010.

[2] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *Int J Data Warehousing and Mining*, vol. 2007, pp. 1–13, 2007.

[3] E. A. Cherman, N. Spolaôr, J. Valverde-Rebaza, and M. C. Monard, "Lazy multi-label learning algorithms based on mutuality strategies," *Journal of Intelligent & Robotic Systems*, vol. 80, no. 1, pp. 261–276, 2015.

[4] C. Liu and L. Cao, *A Coupled k-Nearest Neighbor Algorithm for Multi-label Classification*. Cham: Springer International Publishing, 2015, pp. 176–187.

[5] X. Yan, W. Li, Q. Wu, and V. S. Sheng, *A Double Weighted Naive Bayes for Multi-label Classification*. Singapore: Springer Singapore, 2016, pp. 382–389.

[6] J. Wang, J. Feng, X. Sun, S.-S. Chen, and B. Chen, *SimplifiedConstraints Rank-SVM for Multi-label Classification*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 229–236.

[7] Z. Sun, Z. Guo, M. Jiang, X. Wang, and C. Liu, *Research and Application of Fast Multi-label SVM Classification Algorithm Using Approximate Extreme Points*. Cham: Springer International Publishing, 2016, pp. 39–52.

[8] G. Madjarov and D. Gjorgjevikj, *Hybrid Decision Tree Architecture Utilizing Local SVMs for Multi-Label Classification*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 1–12.

[9] X. Wang, S. An, H. Shi, and Q. Hu, *Fuzzy Rough Decision Trees for Multi-label Classification*. Cham: Springer International Publishing, 2015, pp. 207–217.

[10] P. M. Ciarelli, E. Oliveira, and E. O. T. Salles, "Multi-label incremental learning applied to web page categorization," *Neural Computing and Applications*, vol. 24, no. 6, pp. 1403–1419, 2014.

[11] S. Agrawal, J. Agrawal, S. Kaur, and S. Sharma, "A comparative study of fuzzy pso and fuzzy svd-based rbf neural network for multi-label classification," *Neural Computing and Applications*, pp. 1–12, 2016.

[12] G. Tsoumakas, I. Katakis, and I. Vlahavas, *Data Mining and Knowledge Discovery Handbook*. Boston, MA: Springer US, 2010, ch. Mining Multi-label Data, pp. 667–685.

[13] J. Read, "A Pruned Problem Transformation Method for Multi-label classification," in *Proc. 2008 New Zealand Computer Science Research Student Conference (NZCSRS 2008)*, 2008, pp. 143–150.

[14] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *Mach. Learn.*, vol. 85, no. 3, pp. 333–359, Dec. 2011.

[15] J. Read, L. Martino, P. M. Olmos, and D. Luengo, "Scalable multi-output label prediction: From classifier chains to classifier trellises," *Pattern Recognition*, vol. 48, no. 6, pp. 2096 – 2109, 2015.

[16] E. Hüllermeier, J. Fürnkranz, W. Cheng, and K. Brinker, "Label ranking by learning pairwise preferences," *Artificial Intelligence*, vol. 172, no. 1617, pp. 1897 – 1916, 2008.

[17] 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, 2008.

[18] L. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, no. 3, pp. 338 – 353, 1965.

[19] F. Eibe and H. Mark, "A simple approach to ordinal classification," in *Proceedings of the 12th European Conference on Machine Learning*, ser. EMCL '01. London, UK, UK: Springer-Verlag, 2001, pp. 145–156.

[20] J. S. Cardoso and J. F. P. da Costa, "Learning to classify ordinal data: the data replication method," *Journal of Machine Learning Research*, vol. 8, pp. 1393–1429, 2007.

[21] C. Brinker, E. L. Menca, and J. Frnkranz, "Graded multilabel classification by pairwise comparisons." in *ICDM*, R. Kumar, H. Toivonen, J. Pei, J. Z. Huang, and X. Wu, Eds. IEEE Computer Society, 2014, pp. 731–736.

[22] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: An update," *SIGKDD Explor. Newsl.*, vol. 11, no. 1, pp. 10–18, Nov. 2009.

[23] J. R. Quinlan, *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

[24] S. Arctander, *Perfume and Flavor Chemicals: (aroma Chemicals)*, ser. Perfume and Flavor Chemicals: Aroma Chemicals. Allured Publishing Corporation, 1969.

[25] K. Laghmari, M. Ramdani, and C. Marsala, "A distributed graph based approach for rough classifications considering dominance relations between overlapping classes," in *SITA'15, Intelligent Systems Theories and Applications, 2015 10th Inte. Conf. on*, Oct 2015, pp. 1–6.