S.cerevisiae Complex Function Prediction with Modular Multi-Relational Framework

Beatriz García Jiménez, Agapito Ledezma, and Araceli Sanchis

Universidad Carlos III de Madrid
Av.Universidad 30, 28911, Leganés, Madrid, SPAIN
beatrizg@inf.uc3m.es

Abstract. Gene functions is an essential knowledge for understanding how metabolism works and designing treatments for solving malfunctions. The Modular Multi-Relational Framework (MMRF) is able to predict gene group functions. Since genes working together, it is focused on group functions rather than isolated gene functions. The approach of MMRF is flexible in several aspects, such as the kind of groups, the integration of different data sources, the organism and the knowledge representation. Besides, this framework takes advantages of the intrinsic relational structure of biological data, giving an easily biological interpretable and unique relational decision tree predicting N functions at once.
This research work presents a group function prediction of *S.cerevisiae* (i.e.Yeast) genes grouped by protein complexes using MMRF. The results show that the predictions are restricted by the shortage of examples per class. Also, they assert that the knowledge representation is very determinant to exploit the available relational information richness, and therefore, to improve both the quantitative results and their biological interpretability.

Keywords: Relational Data Mining architecture, Gene function, Biological data integration, Machine Learning, Gene networks.

1 Introduction

Functional genomic is an open problem in molecular biology. Knowing the functions of the genes it is necessary to understand how the organism tasks are distributed, and which gene/s is/are involved in each biological process. Then, if we are faced with a possible malfunction in the metabolism, we will have to locate the problem at molecular level. This knowledge is essential to design a solving treatment to the corresponding disease.

Nowadays, we still do not have this complete functional genomic knowledge. There are many distributed fragments of gene annotation, from multiple researching studies: wet or in-silico, in wide sense or specialized, in a particular area or function. But uncertainty, changeable, incomplete and unreliable annotations suggest us to develop new techniques to improve this essential gene function knowledge.

Going beyond in functional genomic, actually an isolated gene is not the responsible for a specific function, but each function is carried out by a group of genes. Every biological process can be reached due to the collaboration of all genes in the group. Without some of these genes, the biological result would be different, unfinished or non-existent. Thus, a gene annotation might differ depending on the gene groups in which it works. For example, when a gene *A* works in a group *G1* with genes *B* and *C*, the gene *A* function (*F1*) will be different from the gene *A* function (*F2*) in group *G2* consisting of genes *A*, *D*, *E* and *H*. So, functional genomic should be understood as gene *group* function instead gene function; that is, functions are related to a gene group, not to an individual gene.

Given a gene group, whether individual gene function are calculated and after the simple union of these functions are considered as the functions of this group, some drawbacks are ignored; mainly, lack of precision (i.e. a false positive increase) and lack of sensitivity (i.e. a false negative increase). First, a gene can get involved in several functions, although all these functions are not carried out with the same gene group. So, the union of all individual gene functions turns out an over-assignation of group functionality. It means that there are functions that are carried out by a gene of the current group but only working in a different group, with other genes. Second, related to lack of sensitivity, the union of individual gene functions produces also an under-assignation of functionality. In other words, there would probably have joint properties (knowledge shared by several genes) without which it is imposible to assign a specific function to a gene group, even less to an individual gene. Consequently, it is neccesary to support the functional annotation with knowledge about relations among genes.

The wide range of kinds of gene relations in molecular biology results in multiple criteria to make groups of genes: the same regulation network, protein complex, pathway, or protein interaction network [10, 17]; genes with similar patterns in expression profiles from DNA arrays [6, 13]; genes with certain level of sequence similarity, with the same cellular location, protein family, functional annotation [19] or with common phenotypical data (for instance, pathology or tissue), and so on.

There are several methods to build gene groups through biological networks [15], through Gene Ontology functional annotation [19] or from other sources [8], and some techniques to determine if a gene group is statistically significant [18, 21]. However, they are not focus on assigning function to gene groups, as our approach does.

In order to get a suitable gene group functional annotation, different kind of available data sources should be integrated, including both individual gene features (mainly came from gene sequences) and data from several relations among genes (from a group or from whichever gene relation). The huge quantity of these biological data requires the use of computational methods to manage this task. Since the experimental techniques are costly in resources and time, the function prediction methods have shown an useful alternative in the last years [14]. Some interesting approach working with gene groups and functions

have been developed, where different data sources are combined [1], but without taking into account the advantages of Multi-Relational Data Mining (MRDM).

We think that MRDM is more suitable for solving the gene group function prediction problem than traditional propositional Data Mining (DM). In biological domain, there are many relational information, due to the intrinsic structure of the molecules, the importance of the similarity among different species (i.e. homology associations), and even more the relations among genes in groups, which are essential in this domain. Additional advantages of MRDM over the propositional DM approach are: (a)a decrease in the number of redundant features and missing values (very common facts in biological domains); (b)a better representation of real world problems, without losing the semantic after a proposionalization process; (c)an improved storage and management of the data, organised in modules or tables, according to the relations; (d)an easier representation of structured information, such as networks or graphs in interaction networks, pathways or semi-structured data from text mining results.

MRDM have been successfully applied to individual gene function prediction [4, 23]. Other similar biological domains have been faced with relational techniques also: protein-protein interaction prediction [22] and a work with gene groups although only related with microarrays [20].

Uncertainty and unknown information in biology always make difficult the bioinformatics problems. Gene group function annotation is even a harder domain, mainly due to the high variability in its context. First, it varies because there is a frequently changeable environment, caused by the improvement in the high-throughput experimental technologies that produces a huge quantity of data that is constantly renewed, and not neccesary compatible with the old ones. Second, it varies according the bio-expert interest, who alter the selection of the grouping criterion and the kind of input/output data. So, particular and very specific systems are not good solutions for this problem.

This paper proposes a modular framework which is adaptable to be applied to any different gene group function prediction problems. Functional annotation of *S.cerevisiae* genes grouped by complexes is a real open problem dealt with this new multi-relational and flexible approach.

This paper is organised as follows: Section 2 explains the application of the Modular Multi-Relational Framework in yeast genome. Section 3 presents and analyses the application results. Finally, in Section 4, conclusions and future work are summarized.

2 Modular Multi-Relational Framework applied to Yeast Complex Function Prediction

Modular Multi-Relational Framework (MMRF) [7] is a new approach for solving the domain of *Gene Group Function Prediction*, facing the problem from a relational and flexible point of view.

MMRF is designed by modules for managing the high variability which this biological domain entails; changing independiently data, criteria and methodol-

ogy. MMRF uses a multi-relational approach (in representation and learning) for fitting the intrinsic relational structure of gene grouping data, and for integrating different data sources.

MMRF is splitted into six modules (see Figure 1) covering all involved domain tasks (grouping, representing and learning). Each module consists of one or several abstract tasks [7] which must be individually instantiated in a particular application of the framework.

The MMRF application described in this work predicts the function of gene groups in *S.cerevisiae* (i.e.Yeast), where genes are grouped by protein complexes. The instantiation of all the framework modules in this application are described below, that is, the definition of the particular value for the tasks in each MMRF module. The specific results and analysis appear in the Section 3.

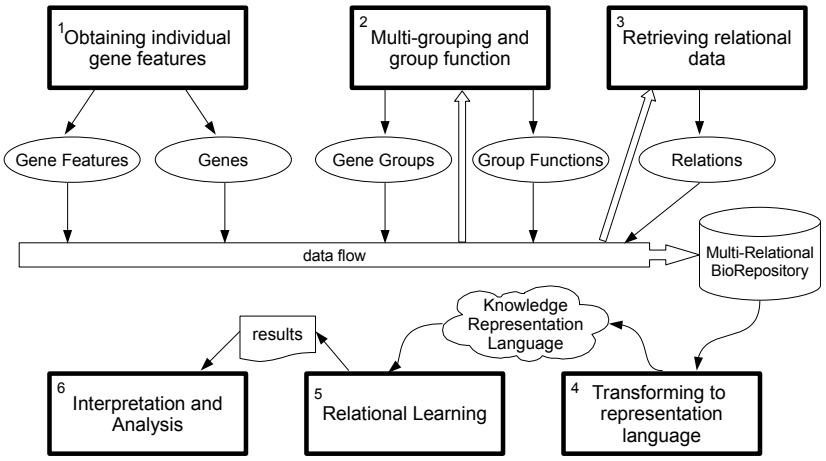

Fig. 1. The schema of the proposed Modular Multi-Relational Framework (MMRF). The rectangles represent modules and the ellipses represent data.

1. **Obtaining individual gene features.** In this application, the individual features for yeast organism are retrieved from Ensembl project [9], through the BioMart tool [16]. Most of them are extracted from gene and protein sequences. Some examples are the gene length, the chromosome name, the gene biotype, the protein family domain, if it is or not a coiled-coil domain, etc.
2. **Multi-Grouping and group function.** This module includes two main activities. (a) Making gene groups, according to a specific criterion. Several of these criteria are listed above (Section 1). In this application, the genes are grouped by protein complexes. They are extracted from high-throughput experimental data from the detailed Krogan et.al. study [10]. These 547 protein complexes are inferred from protein-protein interaction data, after applying a Markov clustering algorithm. The 2,375 interaction pairs are identified by two different mass spectrometry methods, in order to increase the data reliability. (b) Assigning functions to each group. In this initial application,

a Gene Ontology (GO) function shared by a 60% of the genes in a group belongs to the assigned group functions. Gene Ontology Slim:Biological Process is the functional catalogue used. These assigned functions will be the classes in the further supervised learning process.

3. **Retrieving relational data.** The second source of background knowledge, after individual gene features, are common features of a subset of genes. In yeast complex function prediction application, the relational data consists of homology data [9, 16] and regulatory data [11]. Homologs are paralogs from relations between genes from yeast, and orthologs from relations between yeast and mouse, cow or human genes. The homology data represents only binary relations between a pair of genes, while regulatory data means gene group relations (N genes related among themselves, with N>=2). Experimental regulatory data described in [11] includes different kind of networks of regulator-regulated genes. Only one, the most frequent kind of network, has been chosen: multi-input motifs (i.e. a gene set regulates other group of genes). Thus, we get the highest possible number of instances for the learning task, but avoiding to mix different sources. It means 257 gene group relations, and besides 826 regulator-regulated group binary relations.
4. **Transforming to representation language.** The knowledge representation language is defined as first-order logic predicates [12], in a prolog syntax, involving all the collected data described above (see Figure 2).

```
gene_in_group(groupID,geneID).
group_function(groupID,goID).
gene(geneID,chrom,length,strand).
pfam_domain(geneID,pfamID).
transmembrane_domain(geneID).
ncoils_domain(geneID).
...
cds_length(geneID,length).
paralog(geneID,geneID,identity,coverage).
ortholog(geneID,geneID,identity,id2,type).
ortholog_signal_domain(geneID).
gene_in_coregulated(coregGroupID,geneID).
regulator(coregGroupID,geneID).
```

Fig. 2. Fragment of the knowledge representation language in gene group function prediction domain.

5. **Relational Learning.** Using the previously defined logic predicates, we apply the machine learning algorithm TILDE, Top-down induction of logical decision trees [2], implemented in the ACE tool. It is a classical Quinlan decision tree adapted to a relational approach, through first-order logic. This tree has logic clauses in nodes instead of attribute-value comparisons. Before applying TILDE, we carried out a data pre-process, inspired by other works [3, 23], in order to get a multi-class and multi-label learning in an unique classifier. Briefly, each gene group is represented by a boolean vector of functions, and a regression prediction is applied to each possible class, at the same time. It is neccesary because a group of genes (example) can have more than one function assigned (class). This fact is very important in this domain, since it increases its complexity and makes difficult its resolution.

6. **Interpretation and Analysis.** A biological interpretation aproximation of the relational tree appears in the next section. Since the positives predictions are the most interesting in this domain and the highly-skewed data set, Precision-Recall curves (PRC) are the measure more suitable for leading the computational analysis of the learning results [5].

3 Results and Discussion

The goal of this section is to show how MMRF is applied to a real gene function context, and what results it brings out.

3.1 Experimental Design

The relational system learns from 208,098 logic predicates, where different kind of them are included. In particular, 7,124 `gene`, 20,989 `ortholog`, 11,412 `interpro_domain` or 2,246 `gene_in_coregulated` predicates. The initial number of groups retrieved from [10] is 547. In that set, on average, there are 4.9 genes per group, ranged from 2 to 54 genes, being most of the groups small. Besides, the set has on average 2.7 functions per group, ranged from 1 to 10 functional annotations in the same group.

The number of groups in the dataset is reduced after filtering those gene groups without any GOSlim function assignation and following the 60% of shared functions criterion (task 2.b). The final number of groups depends on the required minimum **number of examples per class**. According to that parameter, three different datasets (a, b and c) are defined. These datasets mainly differ in the number of classes (functions) and the number of examples (groups):

a) At least 1 groups per function (≥1): This dataset has 40 different classes in 360 examples (on average, 24 positives and 336 negatives per class).
b) At least 10 groups per function (≥10): This dataset has 24 different classes in 357 examples (on average, 37 positives and 320 negatives per class).
c) At least 50 groups per function (≥50): This dataset has 8 different classes in 280 examples (on average, 57 positives and 223 negatives per class).

These 3 datasets have been defined because the original one is very skewed through the number of examples available for predicting each class. The values goes from 1 to 68 examples per class, underlining a 40% with less than 10 examples per class and a 62.5% with less than 25 examples per class. The effect of this fact is evaluated in the experiments in the next section.

Moreover, there is one more difference in the experiments done, related to the **relational knowledge** used in the learning process: ***(1)binary relations*** (i.e. homology data) or ***(2)binary and group relations*** (that means to add `regulator` and `gene_in_coregulated` predicates). This separation let us to analyse the influence of group relations in the learning process.

The next section shows the results for the 6 configurations we have considered (2 relations subsets x 3 datasets). The remainder background knowledge (i.e. logic predicates) and learning parameters are the same for all the configurations.

3.2 Results and Analysis

The results shown in Table 1 and Figure 3 come from 10 folds cross validation experiments. Table 1 shows several quantitative measures which evaluate different aspects of the relational learning process, from the solution size (two first rows) to the prediction goodness (two last rows). In addition, Precision-Recall curves appear in Figure 3 for some configurations. All of them are the average results about overall classes in a specific configuration. Although several individual classes in the same configuration have its Precision-Recall curve higher than this average curve (see Figure 3, "individual class 2.c" line).

Table 1. Quantitative results from yeast complex function prediction with MMRF. AU(PRC): Area Under Precision-Recall Curve.

	Relational knowledge					
	1) only binary			**2) binary+group**		
	No. examples per class					
	≥1	**≥10**	**≥50**	**≥1**	**≥10**	**≥50**
Tree nodes:	32.6	33.9	28.5	34.8	32.1	29.9
Tree literals:	89.3	93.3	72.1	94.3	66.3	75.3
Correlation:	0.009	0.015	0.056	0.008	0.031	0.049
AU(PRC):	0.120	0.134	0.230	0.116	0.144	0.226
	1.a	**1.b**	**1.c**	**2.a**	**2.b**	**2.c**

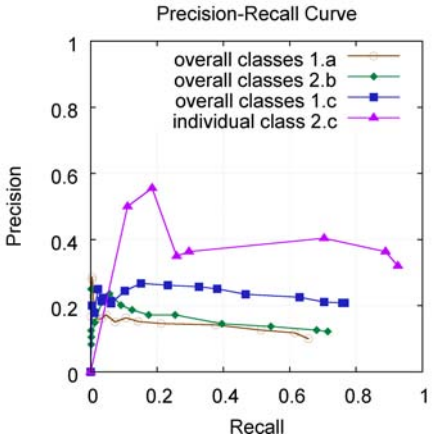

Fig. 3. Precision-Recall curves from yeast complex function prediction with MMRF, in different configurations.

Analysing the obtained results, we can conclude that the quantitative measures are not very high, although they are closed to the results in similar biological domain. For example, our preliminary AU(PRC)s are not very distant from the maximum of 0.3 reached in individual gene function prediction with relational data mining [23].

Furthermore, Table 1 and Figure 3 point out that the predictions improve from "a" to "b", and from "b" to "c" configurations. It means that the results are better when the number of examples per class is higher. Although it also implies that the set of predicted classes is smaller. Therefore, improving the results implies to search alternative ways in order to increase the number of groups. For example, looking for experimental data producing a higher number of gene groups, mixing data from several experimental studies, modifying the grouping criterion to another which gives more groups, or even joining groups coming from different grouping criteria.

The different relational background knowledge used is the remaining analysis proposed according to the experimental design (see Section 4.1). Looking at Table 1 and the decision trees in different configurations (very similar to the

tree shown in Figure 4), we realise that the numeric results barely change from configurations 1 to 2. Moreover, the new regulatory predicates hardly ever appear in the decision tree nodes in configurations 2. Hence, it does not matter if we add more relational information, even more enriching one (group relations is better than only binary relations), because the learning process not take advantage of it. At least, with the current defined knowledge representation.

Additional experiments have been carried out, in order to try to solve the low quantitative results, mainly modifying the grouping criterion in module 2, from protein complexes to coregulated genes or both criteria together. In these additional experiments the flexibility of the framework is checked, because only swapping `gene_in_group` and `gene_in_coregulated` predicates a new framework application is defined: *co-regulated gene function prediction*. So, now the grouping criterion (module 2.a) is co-regulated genes and relational background knowledge (module 3) includes protein complex relations. In this dataset, only 56 examples appear for predicting 30 differents classes, versus 360 examples for 40 classes in the previous application. All classes have less than 10 examples for learning. Consequently, the output predictor will be not generic, but very dependent on these few examples. Since the results depend very much on the specific MMRF application, we must take care of its design.

Other experiments consist of increasing the number of groups and examples per class by joining groups coming from different grouping criteria, particularly protein complexes and co-regulated genes criteria. Thus, we define easily other new application. Here, the number of groups increases from 360 to 415, with at least one assigned GO function. However, the results are practically the same as Table 1 shows, since groups from complexes are much more abundant than from co-regulation, so the latter are covered by the former. This is the bad consequence of mixing data for increasing the number of examples. Therefore, with this two modifications, it is checked as a tricky domain.

```
1: class(-A,-B,-C,-D,-E,-F,-G,-H,-I)
2: [0.186507936507937] 252.0
3: gene_in_group(A,-J),gene(J,-K,-L,-M),ncoils_domain(J) ?
4: +--yes: [0.12258064516129] 155.0
5: |       paralog(J,-N,-O,-P),O>=23 ?
6: |       +--yes: [0.15714285,0.37142857,0.17142857,0.07142857,
7: |                 0.25714285,0.18571428,0.1,       0.37142857]
8: |               70.0
9: |               [0.04381267,0.05816884,0.04537138,0.03100409,
10:|                0.05261569,0.04681516,0.03611575,0.058168843]
11:|       +--no:  [0.141176470588235] 85.0
12:|               ortholog(J,-Q,-R,-S,-T),R>=22 ?
13:|               ...
```

Fig. 4. Fragment of the relation decision tree in configuration '1.c'.

On the other hand, the Figure 4 shows a relational decision tree fragment. The biological interpretation of this tree means that given a gene J belonging to the group A, if gene J had a coiled-coil domain and a paralog relationship with gene N with an identity $\geq$23%, then the eigth regression values in lines 6 and 7 determine the functions assigned (among B to I) to group A. For example, with a threshold of 0.35, the functions *C=GO:0006350* and *I=GO:00050709*

would be predicted to the group A. The line 8 in Figure 4 tells how many examples satisfy these conditions, and line 9 and 10 show the error measured in the regression process. The tree goes on checking if other logic predicates are true in the knowledge base.

4 Conclusions and Further Work

In this work, a variation of gene function prediction domain is tackled: gene *group* function prediction. The Modular Multi-Relational Framework (modular for flexibility and Multi-Relational due to the structured data) is proposed to solve this new domain. It is applied to a specific real problem, the yeast complex function prediction. Preliminary quantitative results are around low levels, although not very distant from typical ones in related domains. However, it seems clear that the prediction is very restricted by the high number of functions (i.e. classes) and mainly by the few groups per function. It means that when the latter number increases (and consequently the former decreases), the predictions improve. In addition, it is concluded that the group relational data is not exploited with the current knowledge representation.

Thus, an important improvement in this research would be to modify the knowledge representation (module 4) so that the learning process can take advantage of the fundamental information existing in bio-relational data. As further work related to the skewed classes, we could explore different alternatives in order to increase the number of groups in yeast, analysing their influence in the prediction results. Another proposal is to include new grouping criteria, with several goals; such as, to apply the system to multi-grouping scenarios, to define new framework applications changing the instantiation of module 2 (for example, pathways function prediction or co-expressed gene function prediction), and to increase the number of groups too. Other near idea is to add new relational information (module 3), integrating more data sources. Also, the functional catalogue could be changed to other less generic than Gene Ontology.

However, it might occur that the collected data and similar one not contains enough knowledge to face the problem of predicting group functions directly. Maybe the next step should be to slightly change the approach, splitting the prediction process in two phases. First, predicting individual gene function with Relational Data Mining (whose viability has been checked in closed domain [4, 23]), although increasing the relevance of gene group membership, as background knowledge. Second, predicting group functions, through the inferred individual function from the first phase.

Finally, probably the most relevant improvement is to change the objective organism from yeast to human. It implies more interesting gene groups and more useful annotations (for instance, related to some disease). Thus, this framework could be applied to predict results so relevant as the unknown function of human gene groups.

Acknoledgments. This research work has been supported by CICYT, TRA 2007-67374-C02-02 project and by the expert biological knowledge of the Struc-

tural Computational Biology Group in Spanish National Cancer Research Centre (CNIO). The authors would like to thank members of TILDE tool developer group in K.U.Leuven for providing their help and many useful suggestions.

References

1. F. Al-Shahrour et al. Fatigo+: a functional profiling tool for genomic data. *Nucl.AcidsRes.*, 35:91–96, 2007.
2. H. Blockeel and L. De Raedt. Top-down induction of logical decision trees. *Artificial Intelligence*, 101 (1-2):285–297, 1998.
3. H. Blockeel et al. Decision trees for hierarchical multilabel classification: A case study in functional genomics. In *PKDD*, 2006.
4. A. Clare. *Machine learning and data mining for yeast functional genomics.* PhD thesis, University of Wales, Aberystwyth, Feb. 2003.
5. J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In *ICML*, pages 233–240, New York, USA, 2006. ACM.
6. R. Edgar et al. Gene Expression Omnibus: NCBI gene expression and hybridization array data repository. *Nucl. Acids Res.*, 30(1):207–210, 2002.
7. B. Garcia et al. Modular multi-relational framework for gene group function prediction. In *ILP*, 2009.
8. D. Glez-Pena et al. WhichGenes: a web-based tool for gathering, building, storing and exporting gene sets with application in gene set enrichment analysis. *Nucl. Acids Res.*, 37:329–334, 2009.
9. T. Hubbard et al. Ensembl 2009. *Nucl. Acids Res.*, 37:690–697, 2009.
10. N. Krogan et al. Global landscape of protein complexes in the yeast saccharomyces cerevisiae. *Nature*, 440(7084):637–643, 2006.
11. T. Lee et al. Transcriptional regulatory networks in saccharomyces cerevisiae. *Science*, 298:799–804, 2002.
12. J.W. Lloyd. *Foundations of logic programming.* Springer-Verlag New York, 1987.
13. H. Parkinson et al. Arrayexpress update–from an archive of functional genomics experiments to the atlas of gene expression. *Nucl. Acids Res.*, 37:868–872, 2009.
14. P. Pavlidis et al. Learning gene functional classifications from multiple data types. *Journal of computational biology*, 9(2):401–411, 2002.
15. R Sharan etal. Network-based prediction of protein function. *MolSystBiol*, 3, 2007.
16. D. Smedley et al. Biomart-biological queries made easy. *BMC Genomics*, 10, 2009.
17. C. Stark et al. Biogrid: a general repository for interaction datasets. *Nucl. Acids Res.*, 34:535–539, 2006.
18. A. Subramanian et al. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *PNAS*, 102:15545–15550, 2005.
19. Z. Tang et al. Prediction of co-regulated gene groups through gene ontology. In *IEEE CIBCB'07*, pages 178–184, 2007.
20. I. Trajkovski et al. Learning relational descriptions of differentially expressed gene groups. *IEEE Transactions on Systems, Man, and Cybernetics*, 38(1):16–25, 2008.
21. I. Trajkovski et al. Segs: Search for enriched gene sets in microarray data. *Journal of Biomedical Informatics*, 41(4):588–601, 2008.
22. T. Tran et al. Using inductive logic programming for predicting protein-protein interactions from multiple genomic data. In *PKDD*, pages 321–330, 2005.
23. C. Vens et al. Decision trees for hierarchical multi-label classification. *Machine Learning*, 73(2):185–214, 2008.