UNIVERSIDADE FEDERAL FLUMINENSE

RAFAEL BARROS PEREIRA

Lazy Feature Selection for the Multi-label Classification Task

NITERÓI 2015

UNIVERSIDADE FEDERAL FLUMINENSE

RAFAEL BARROS PEREIRA

Lazy Feature Selection for the Multi-label Classification Task

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Doutor em Computação. Área de concentração: Engenharia de Sistemas e Informação

Orientador: Alexandre Plastino

Coorientadora: Bianca Zadrozny

> NITERÓI 2015

RAFAEL BARROS PEREIRA

LAZY FEATURE SELECTION FOR THE MULTI-LABEL CLASSIFICATION TASK

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense como requisito parcial para a obtenção do Grau de Doutor em Computação. Área de concentração: Engenharia de Sistemas e Informação

Aprovada em DEZEMBRO de 2015.

BANCA EXAMINADORA

Prof. Alexandre Plastino de Carvalho, UFF - Presidente

Prof.^a Bianca Zadrozny, IBM Research Brazil

Prof.ª Vanessa Braganholo Murta, UFF

Prof. José Viterbo Filho, UFF

Dr.ª Adriana Bechara Prado, EMC

Prof.^a Gisele Lobo Pappa, UFMG

Niterói

2015

Resumo

Classificação multirrótulo é um importante tópico de pesquisa na área de mineração de dados. Diferente da classificação tradicional monorrótulo, onde cada instância está sempre associada a apenas uma classe, na classificação multirrótulo cada instância pode estar associada a mais de uma classe. A popularidade crescente da classificação multirrótulo pode ser explicada em razão da sua aplicabilidade em vários problemas relevantes, tais como: categorização de texto, análise biomolecular, classificação de vídeo, diagnóstico médico, entre outros. Nos últimos anos têm crescido, por consequência, a importância e o interesse de seleção de atributos para essa tarefa. Técnicas de seleção de atributos têm por objetivo a identificação de atributos relevantes para a classificação, removendo atributos redundantes ou irrelevantes da base de treinamento. Entretanto, os métodos propostos para a tarefa de seleção de atributos específica para bases de dados multirrótulo estão espalhados na literatura de classificação multirrótulo, sem uma categorização proposta para descrevê-los e permitir uma comparação objetiva. Uma das contribuições deste trabalho é a criação de uma taxonomia visando a categorização das técnicas de seleção de atributos para esta de seleção de atributos relevantes da categorização de se forma de seleção de atributos específica para bases de dados multirrótulo estão espalhados na literatura de classificação multirrótulo, sem uma categorização proposta para descrevê-los e permitir uma comparação objetiva.

Além da seleção de atributos, outra questão relacionada à classificação multirrótulo é a avaliação do desempenho de cada estratégia. Há uma grande quantidade de medidas que foram adaptadas do paradigma monorrótulo ou desenvolvidas especificamente para o paradigma multirrótulo, e cada trabalho da literatura opta por um subconjunto distinto de medidas, dificultando a comparação dos resultados. Tendo em vista essa dificuldade, outra contribuição deste trabalho é uma extensa avaliação da correlação e a relevância das medidas de desempenho para a tarefa de classificação multirrótulo.

Por fim, o trabalho propõe a adaptação de uma técnica de seleção de atributos monorrótulo para o paradigma multirrótulo, a comparação experimental com outras técnicas conhecidas de seleção multirrótulo de atributos e, como principal contribuição, a criação de um novo método de seleção baseado na seleção de atributos do tipo lazy e específico para o contexto multirrótulo. Resultados experimentais demonstram que as técnicas propostas são competitivas em relação às técnicas de seleção multirrótulo atualmente em uso na literatura, além de serem claramente mais escaláveis em um cenário em que a quantidade de informação das bases de dados é crescente.

Palavras-chave: Seleção de Atributos, Classificação Multirrótulo, Aprendizado Lazy.

Abstract

Multi-label classification is an important topic of research in the data mining area. Unlike traditional single-label classification, where each instance is always associated with a unique class label, in multi-label classification each instance can be associated with more than one class label. The increasing popularity of multi-label classification can be explained due to its applicability in many relevant problems, such as: text categorization, biomolecular analysis, video classification, medical diagnosis, among others. In the last few years, consequently, there has been substantial research in feature selection for this task. Feature selection aims at identifying relevant features for classification, by removing redundant or irrelevant features from the training data set. However, the methods proposed for the feature selection specific to multi-label data sets are scattered in the multi-label classification literature, with no common framework to describe them and to allow an objective comparison. One of the contributions of this work is the formulation of a taxonomy for categorizing existing feature selection techniques for multi-label classification.

Besides feature selection, another problem related to multi-label classification is the performance evaluation of each strategy. There are many measures adapted from the single-label paradigm or developed specifically for the multi-label paradigm. Each different work in the area employs a distinct subset of measures, so it is difficult to compare results across them. This work presents an extensive analysis of the correlation and relevance of performance measures for the multi-label classification task.

Finally, this thesis proposes an adaptation of a single-label technique to the multi-label paradigm, which is compared experimentally with some well-known multi-label feature selection techniques; and, as our main contribution, the creation of a novel selection method, based on lazy feature selection and specific for the multi-label paradigm. Experimental results show that the proposed technique is competitive relative to multi-label feature selection techniques currently used in the literature, and is clearly more scalable, in a scenario where there is an increasing amount of data.

Keywords: Feature Selection, Multi-label Classification, Lazy Learning.

List of Figures

2.1	Multi-label classification based on select and copy transformations	7
2.2	Multi-label classification based on label powerset transformation \ldots . \ldots .	8
2.3	Multi-label classification based on binary relevance transformation	9
2.4	Multi-label classification based on algorithm adaptation	12
3.1	Multi-label evaluation measures categorization [86]	15
3.2	Graphical representation of pairwise correlation between evaluation measures	24
3.3	Graphical representation of the 12 lesser correlated measures \ldots	26
4.1	Taxonomy proposed for multi-label feature selection	29
4.2	Transformation Based/Single multi-label feature selection	31
4.3	$Transformation \ Based/Binary \ Relevance \ multi-label \ feature \ selection . \ .$	33
4.4	$Direct/Filter\ multi-label\ feature\ selection\ .\ .\ .\ .\ .\ .\ .\ .\ .\ .\ .$	37
4.5	$Direct/Wrapper \ multi-label \ feature \ selection \ \ . \ . \ . \ . \ . \ . \ . \ . \ . $	39
5.1	Critical diagram for each measure in the BRKNN classifier from the Ne- menyi post-hoc test at 0.05 significance	51
6.1	Critical diagram for each measure with the BRKNN classifier from the Nemenyi post-hoc test at 0.05 significance	62

List of Tables

2.1	Small multi-label data set example	7
3.1	Statistics for each evaluation measure, adapted from $[69]$	20
3.2	Multi-Label data sets used for analysis of measures	21
3.3	Multi-label classifiers used in the experiments	22
3.4	Multi-label measures compared with Pearson correlation	23
3.5	Multi-label measures compared with Spearman correlation	25
4.1	Summary of publications on ML feature selection based on transformation	36
4.2	Summary of publications on direct multi-label feature selection	41
5.1	Multi-label data sets used in the experiments	46
5.2	Results achieved with the BR-KNN classifier for the Hamming Loss measure	47
5.3	Multi-label classifiers used in the experiments	48
5.4	Best results achieved with the BRKNN classifier	49
5.5	Number of times that each feature selection achieved a result better than (\leq) the baseline score	50
5.6	Result of experiments on large data sets with BRKNN classifier	53
6.1	Single-label Data Set Example	55
6.2	Multi-label Data Set Example	56
6.3	Best results achieved with the BRKNN classifier, comparing feature selection techniques with the proposed LazyMLInfoGain technique	60
6.4	Result of experiments on large data sets with BRKNN classifier, comparing MLInfoGain with Lazy MLInfoGain feature selection	63
6.5	Result of experiments on large data sets with BRKNN classifier, comparing BR+InfoGain with Lazy MLInfoGain feature selection	63

6.6	Best results achieved with the ML-KNN classifier, comparing feature selection techniques with the proposed LazyMLInfoGain technique	64
6.7	Result of experiments on large data sets with Lazy MLInfoGain (10%) feature selection and the BRKNN and ML-KNN classifiers	66
7.1	Multi-label Data Set Example	78
7.2	Data Set Example after Copy transformation	79
7.3	Data Set Example after LP transformation	79
7.4	Data Set Example after BR transformation	80
7.1	Best results achieved with the HOMER + K-NN classifier $\ldots \ldots \ldots$	82
7.2	Best results achieved with the PPT + K-NN classifier	83
7.3	Best results achieved with the RK + DecisionTree classifier	84
7.4	Best results achieved with the RAKEL + K-NN classifier \ldots	85
7.5	Best results achieved with the $CC + K$ -NN classifier	86
7.6	Best results achieved with the BR $+$ DecisionTree classifier \ldots	87
7.7	Best results achieved with the BR $+$ NaiveBayes classifier	88
7.8	Best results achieved with the CC + DecisionTree classifier	89
7.9	Best results achieved with the CC + NaiveBayes classifier	90
7.10	Best results achieved with the IBLR-ML classifier	91
7.11	Best results achieved with the LP + DecisionTree classifier $\ldots \ldots \ldots$	92
7.12	Best results achieved with the LP + K-NN classifier	93
7.13	Best results achieved with the LP $+$ NaiveBayes classifier	94
7.14	Best results achieved with the ML-KNN classifier	95
7.15	Best results achieved with the RK $+$ NaiveBayes classifier $\ldots \ldots \ldots$	96
7.1	Multi-label Data Set Training Example	97
7.2	Data Set Partition where $X = 1$	99
7.3	Lazy Entropy Scores for each Value and Label in the Example Data Set	102

Contents

1	Introduction	1
2	Multi-label Classification	4
	2.1. Introduction	4
	2.2. Strategies Based on Data Set Transformation	5
	2.2.1. Copy and Select Transformations	5
	2.2.2. Label Powerset Transformation	6
	2.2.3. Binary Relevance Transformation	9
	2.3. Strategies Based on Algorithm Adaptation	11
	2.4. Chapter Summary	13
3	Correlation Analysis of Performance Measures for Multi-Label Classification	14
	3.1. Introduction	14
	3.2. Multi-label Measures	15
	3.2.1. Example-based Classification Measures	16
	3.2.2. Example-based Ranking Measures	17
	3.2.3. Label-based Classification Measures	18
	3.2.4. Label-based Ranking Measures	19
	3.2.5. Multi-label Measures in the Literature	19
	3.3. Correlation among Multi-label Measures	20
	3.3.1. Analysis of Pearson Correlation	22
	3.3.2. Analysis of Spearman Correlation	23

	3.3.3. Guidelines for Choosing Measures in a Multi-label Setting \ldots	25
	3.4. Chapter Summary	27
4	Multi-label Feature Selection	28
-	4.1 Introduction	- ° 28
	4.2 Multi label Feature Selection Based on Transformation	20
	4.2.1 Strategies Based on Single Data Transformation	30
	4.2.1. Strategies Based on Single Data Transformation	00 20
	4.2.2. Strategies Based on Binary Relevance Transformation	32
	4.2.3. Summary of Publications on Transformation Based Feature Selection	35
	4.3. Direct Multi-label Feature Selection	36
	4.3.1. Strategies Based on the Filter Strategy	36
	4.3.2. Strategies Based on the Wrapper Strategy	38
	4.3.3. Strategies Based on the Embedded Strategy	39
	4.3.4. Summary of Publications on Direct Multi-label Feature Selection $\ .$.	40
	4.4. Discussions and Conclusions	41
5	Information Gain Adaptation for Multi-label Data	43
	5.1. Introduction	43
	5.2. Information Gain Feature Selection Adaptation	43
	5.3. Experimental Evaluation	45
	5.3.1. Methodology	45
	5.3.2. Statistical Evaluation	50
	5.3.3. Experiments on Large Multi-label Data Sets	52
	5.4. Chapter Summary	53
6	Lazy Multi-label Feature Selection	54
	6.1. Introduction	54
	6.2. Lazy Feature Selection	54

	6.3.	Multi-label Adaptation	55
	6.4.	Experiments with BRKNN Classifier	59
		6.4.1. Methodology	59
		6.4.2. Statistical Evaluation	61
		6.4.3. Experiments on Large Multi-label Data Sets for BRKNN	62
	6.5.	ML-KNN Lazy Feature Selection	63
		6.5.1. Experiments on Large Multi-label Data Sets for ML-KNN	65
	6.6.	Chapter Summary	66
7	Con	clusions 7.1. Future Work	68 69
Re	ferei	ices	71
Ap	pend	lix A - Transformations on Multi-Label Data Sets	78
Ap	openo	lix B - MLInfoGain Compared with Transformation-based Techniques	81
Ap	peno	lix C - Application of the MLInfoGain and Lazy MLInfoGain Equations	97
	C.1.	. Computing the MLInfoGain measure	97
	C.2	. Computing the Lazy MLInfoGain measure	101

Chapter 1

Introduction

A large body of research in supervised learning deals with the analysis of single-label data, where instances are associated with a single label from a set of class labels [75]. More specifically, the single-label classification problem can be stated as the process of predicting the class label of new instances described by their feature values.

However, in many important data mining applications, such as text categorization, biomolecular analysis, scene classification and medical diagnosis, the instances are associated with more than one class label. This characterizes the multi-label classification problem, a recent and relevant topic of research, that has become a very common real-world task [85].

In a broad way, two groups of classification strategies have been proposed to deal with multi-label data. In the first group, the multi-label data is converted into single-label data and then the classification problem is solved using single-label classifiers. The second group is related with proposals for adapting or extending single-label classifiers to cope with multi-label data. In the former group one can find popular methods like label powerset and binary relevance transformations, and in the latter group common adaptations are: the multi-label k-nearest neighbors, the multi-label Naive Bayes classifier, multi-label AdaBoost, among others [11, 75].

In single-label classification, an instance can be classified either correctly or incorrectly. But in multi-label classification, an instance can be classified as partially correct, as the predicted subset of labels can differ not completely from the actual subset that belongs to the instance. So, the evaluation of methods that learn from multi-label data requires different measures than those used in single-label contexts [77]. There are more than twenty performance measures adapted from the single-label paradigm or developed specifically for the multi-label paradigm [74]. However, different work in the area employ distinct subsets of measures, so it is difficult to compare results across them. A contribution of this work is an analysis of multi-label measures commonly used in multi-label work and a correlation analysis between them. This analysis aims at guiding the decision of which subset of measures should be selected for reporting computational experiments in the multi-label paradigm.

The performance of a classification method is closely related to the inherent quality of the training data. Redundant and irrelevant features may not only decrease the classifier's accuracy but also make the process of building the model or running the classification algorithm slower. Feature selection is a data preprocessing step which aims at identifying relevant features for a target data mining task – specifically in this work, the classification task. Feature selection techniques are usually applied for removing from the training set features that do not contribute to, or even decrease, the classification performance [27, 41].

There is an extensive literature regarding feature selection for single-label classification, which has been summarized in surveys [10, 27, 48]. In the last few years, given the increasing popularity of multi-label classification, there has been significant research specifically in the area of feature selection for multi-label classification.

The aforementioned methods proposed for the feature selection task are scattered in the multi-label classification literature, with no common framework to describe them and to allow an objective comparison. One of the contributions of this thesis is the proposal of a taxonomy for these methods in order to review and categorize multi-label feature selection techniques.

This thesis also presents, as an important contribution to the multi-label area, a novel feature selection technique based on the adaptation of the single-label information gain measure. Information gain, which is based on the entropy concept, is a common measure of feature relevance in single-label filter strategies that evaluate features individually [80]. Previously, multi-label feature selection techniques in the literature used this measure after transforming the multi-label data set into a single-label one [3, 49, 57, 62, 68, 73, 80, 89]. The adaptation proposed in this thesis does not rely on transformation, and it is evaluated against these latter methods. The results show a significant improvement in the computational performance.

Lazy feature selection strategy was proposed for single-label classification [52]. It is based on the hypothesis that postponing the selection of features to the moment at which an instance is submitted for classification can contribute to identifying the best features for the correct classification of that particular instance. The main contribution of this thesis is a novel multi-label feature selection technique based on two characteristics: (a) the use of the information gain measure which was adapted for multi-label feature selection as a previous contribution; and (b) a multi-label adaptation of the lazy strategy to benefit the multi-label classification. This novel technique is compared experimentally with other multi-label feature selection techniques, and the results show that it is both competitive and much more scalable than current techniques used in the literature.

This work is organized as follows.

- (a) **Chapter 2** presents a bibliographic review of relevant work related to multi-label classification.
- (b) **Chapter 3** reviews the measures used in multi-label classification and presents a correlation analysis between them, guiding researchers in how to select suitable subsets of measures.
- (c) **Chapter 4** presents a multi-label feature selection taxonomy and reviews current work in the area, categorizing them according to this taxonomy.
- (d) **Chapter 5** presents the adaptation of the information gain measure and compares it experimentally with transformation-based methods currently employed in the literature.
- (e) Chapter 6 proposes the main contribution of this thesis: a novel multi-label feature selection technique based on the lazy paradigm and on the information gain multilabel adaptation. The experimental results confirm that the proposed technique is competitive when compared with other feature selection techniques used in the literature, and much more scalable for larger data sets.
- (f) Chapter 7 presents the concluding remarks of the thesis.

Chapter 2

Multi-label Classification

2.1. Introduction

The classification task can be stated as the process of predicting the class label of an instance described by a vector of feature values, given a training set where each instance is described by a vector of features and by a class label. Traditional classification is performed as a single-label task, where each data instance is associated with a single class label. Well-known single-label classification techniques include decision trees [54, 55], k-NN (k-Nearest Neighbors) [6, 9], Naive Bayes [15], neural networks [60], associative classifiers [40], SVM (Support Vector Machines) [2] and others.

The single-label classification problem can be formally defined as follows [22].

Definition 1 (Single-Label Classification). Let $X = X_1, ..., X_d$ be a set of d predictive features and $L = l_1, ..., l_q$ be a set of q class labels, where $q \ge 2$. Consider a training data set D composed of N instances of the form $(x_1, c_1), (x_2, c_2), ..., (x_N, c_N)$. In this data set, each x_i corresponds to a vector $(x_1, ..., x_d)$ that stores values for the d predictive features in X and each $c_i \in L$ corresponds to a single class label. The goal of the single-label classification task is to learn from D a function (a.k.a. classifier) y that, given an unlabeled instance t = (x, ?), is capable of effectively predicting its class label c, i.e., $y(t) \rightarrow c$. When |L| = 2 the problem is called a binary single-label classification problem.

On the other hand, in the multi-label classification task, each data instance may be associated with multiple labels. Multi-label classification is suitable for many domains such as text categorization, scene and video classification, medical diagnosis, applications in microbiology [57], and it is also a challenging problem in bioinformatics [38]. In all these The multi-label classification problem can be formally defined as follows [22, 74].

Definition 2 (Multi-Label Classification). Let $X = X_1, ..., X_d$ be a set of d predictive features and $L = l_1, ..., l_q$ be a set of q class labels, where $q \ge 2$. Consider a training data set D composed of N instances of the form $(x_1, Y_1), (x_2, Y_2), ..., (x_N, Y_N)$. In this data set, each x_i corresponds to a vector $(x_1, ..., x_d)$ that stores values for the d predictive features in X and each $Y_i \subset L$ corresponds to a subset of labels. The goal of the multi-label classification task is to learn from D a classifier h that, given an unlabeled instance t = (x, ?), is capable of effectively predicting its set of labels (a.k.a. labelset) Y, i.e., $h(t) \to Y$.

The strategies proposed to deal with multi-label classification rely mainly on problem transformation, where the multi-label problem is transformed into one or a set of single-label problems; and on algorithm adaptation, where the single-label learning algorithms are adapted to handle multi-label data directly [11, 77]. Both paradigms are presented in the next two sections.

2.2. Strategies Based on Data Set Transformation

The simplest way to apply a classification strategy to a multi-label data set is to transform it into a single-label data set. Then a traditional classification technique – like k-NN or a decision tree – can be employed to perform the classification task. This way, the transformation technique allows the usage of one or more single-label classification algorithms, which have been thoroughly studied and perfected over the last decades.

There are plenty of algorithms to transform a multi-label data set into a single-label one, mainly with the focus on classification. In the next subsections the more common transformation algorithms are described. Also, Appendix A shows a multi-label data set and the corresponding single-label data sets after applying the most common transformation methods.

2.2.1. Copy and Select Transformations

A simple family of transformations used to convert a multi-label data set into a singlelabel one is the select transformation. It consists of selecting for each instance one label among its label subset. The label selected can be the most frequent label in the data set (select-max), the least frequent label (select-min), a random label (select-random) or it can simply discard every multi-label example (select-ignore) [1, 3, 75].

Another family is the copy transformations, that consists in copying each multi-label instance n times, where n is the number of labels assigned to that instance. Each copied instance is then assigned one distinct single label from the original set. A variation of the copy transformation is the copy-weight, which associates a weight 1/n to each copied instance, according to the number n of labels of the original instance [77]. This variation can only be employed if the classifier is able to handle weighted instances.

Figure 2.1 illustrates the generic process of multi-label classification using a simple transformation on the original data set. A data transformation is applied to a multi-label data set, which is then used to train a traditional single-label classifier. The single-label classifier is expected to output not only a single prediction, but a probability distribution over the labels. This classifier is then applied to unlabeled new instances and a label ranking procedure can be used to select a set of labels for each new instance. Usually, a threshold on the label probability is used to select the more relevant labels. The dashed arrow indicates that the subsequent process is not always executed. For instance, for a label ranking data mining task it is not necessary to process the ranking and obtain a multi-label classification as an output, because the label ranking itself is the desired output.

2.2.2. Label Powerset Transformation

Label powerset (LP) is another kind of transformation which creates one new label for each different subset of labels that exists in the multi-label training data set. Thus, the new set of labels corresponds to the powerset of the original set of labels.

Table 2.1 gives a small example of a multi-label data set. There are four different labels for this data set: A, B, C and D. Some of the instances are associated with more than one label. When applying an LP transformation, the new set of labels becomes: A, BC, D and ABC.

After this transformation process, a single-label classification algorithm can handle the transformed data set and produce a classifier. This classifier can then be used to assign one of these new labels to new instances, which can then be mapped back to the corresponding subset of the original labels [78] – for instance, ABC becomes {A,B,C}.



Figure 2.1: Multi-label classification based on select and copy transformations

instance	features	$_{\mathrm{classes}}$
1	0.5, x	А
2	0.1, y	B,C
3	0.3, x	D
4	0.5, z	B,C
5	0.1, w	A,B,C
6	0.2, z	A

Table 2.1: Small multi-label data set example



Figure 2.2: Multi-label classification based on label powerset transformation

Label powerset is recommended only for data sets with a small number of labels, as the possible powerset combinations are 2^L , where L is the number of distinct labels in the data set. For data sets with a large number of labels, the resulting powerset data tends to become sparse and therefore making it harder for the classifier to work.

Figure 2.2 illustrates the generic process of multi-label classification using the powerset transformation on the original data set. The data transformation is applied to a multi-label data set, which is then used to train a traditional single-label classifier. Afterwards, the single-label classifier is employed to classify each new instance into one of the new labels, which is then mapped back into the corresponding multi-label set. Instead of yielding a single label class as the result, the classifier can output a probability distribution over all powerset labels. Then a single label ranking can be obtained by sorting the original labels by the probabilities from the powerset that contains them [77].

The original label powerset technique has been extended and improved in subsequent work. Two variations are the pruned problem transformation (PPT), proposed in [56], and the random k-labelsets (RAKEL), proposed in [78]. The PPT method prunes away label sets that occur fewer times than a small user-defined threshold (e.g., 2 or 3) [77], splitting them in smaller label sets that occur more frequently. This overcomes the problem of creating rarely used classes, and thus reduces overfitting via pruning [56]. The RAKEL



Figure 2.3: Multi-label classification based on binary relevance transformation

method constructs an ensemble of LP classifiers trained using different and small random subsets of the set of labels [77]. The result is then combined in a ranking by averaging the prediction of each LP classifier, per label, and a threshold is employed to sort relevant from irrelevant labels.

2.2.3. Binary Relevance Transformation

Binary relevance (BR) is an important and well-known transformation technique that produces a binary classifier for each different label of the original data set. In its simplest implementation, each resulting classifier is capable of predicting if a label is relevant for a new instance. So, each classifier handles the data as single-labeled, since it gives a relevance feedback for just one specific label. The method is called "binary relevance", because each label is considered as relevant or non-relevant.

Figure 2.3 illustrates the generic process of multi-label classification using the binary relevance transformation process. The data transformation is applied to the multi-label data set, generating L single-label data sets, where L is the number of distinct labels in the data set. Each single label data set corresponds to one label and contains all instances of the original data set, labeled positively or negatively depending on the occurrence of

the corresponding label in the original label set of the instances. Each of these data sets is used to train a traditional single-label classifier. Then, in the classification phase, each classifier outputs a prediction for one single label, either positive or negative. The multilabel classification can be achieved by combining the results of all single-label classifiers.

As binary relevance learns a single binary model for each different label, it has linear complexity with respect to the number of labels [74].

Binary relevance does not take into account label correlations. Without this information, it may fail to accurately predict a specific label combination [74]. In order to minimize this drawback, several techniques have been proposed to extend and improve the binary relevance technique [19, 21, 29, 47, 58].

In [21], the SVM algorithm was extended with a BR algorithm to consider for each instance the prediction of multiple labels individually. It also tries to handle correlation between labels by learning a second level of binary models, that follows the paradigm of stacked generalization, widely used in neural nets and adopted in subsequent multi-label work for musical titles, image and video processing [74]. In [87], multiple combinations of binary relevance with classification methods are employed in a number of multi-label data sets. BR is coupled with the following classifiers: k-NN, C4.5, random decision trees, Naive Bayes and SVM.

In [29], the algorithm called Ranking by Pairwise Comparison (RPC) was proposed. This strategy learns to predict whether a label is favored over others in order to create a label ranking. So, instead of training L classifiers for the set of L labels, RPC trains a classifier for each pair of labels, resulting in L * (L - 1)/2 predictors, i.e., a quadratic number of classifiers [47]. Different ranking algorithms allow the ensemble of pairwise classifiers to adapt to different loss functions on label rankings [29].

In [19], the Calibrated Ranking by Pairwise Comparison (CRPC) was proposed as an extension of the RPC technique. It introduces an additional artificial label used as a calibration factor to separate relevant from irrelevant labels. This label splits a ranking into a positive and a negative part, equivalent to a threshold value. Experimental results with text and gene data sets reveal that the calibrated pairwise ranking approach outperforms the original binary relevance ranking (BR) and also the pairwise ranking strategy (RPC).

HOMER [76] stands for Hierarchy Of Multilabel classifiers. It constructs multiple classifiers, each one dealing with a smaller set of labels, balancing them in a tree-shaped hierarchy built recursively.

The Classifier Chains (CC) method [58] is based on the binary relevance transformation. However, instead of processing each binary transformation independently, it uses a stacking-like process that evaluates each label according to the previous classifications. In the CC's training phase, it organizes all q labels in a randomly-ordered chain. The first classifier in the chain is trained using just the features that compose the feature set X. Then, the next classifier is trained using a different feature space, that is X augmented with the classification information of the first label in the chain. The next in the chain has X plus two new features, and so on. The feature space for each binary model is extended with the label relevances of all previous classifiers. By forming these chains of labels, the method is able to consider correlations between them.

Some variations of the Classifier Chains try to improve the performance of the classifier by changing the ordering of chains. Some of them are the One-to-One Classifier Chains (OOCC) [8], which assigns a label sequence to each new instance in the test set based on the label sequences that perform well in similar training instances; the Genetic Algorithm for Optimizing CC (GACC) method [23, 24], which makes use of an evolutionary algorithm to optimize chain classifiers; and the Ensembles of Classifier Chains (ECC) [59], which extends the original method by using a bagging scheme and ordering each binary model randomly, therefore resulting in different chains that predict different label sets.

2.3. Strategies Based on Algorithm Adaptation

The previous section described data transformations used to enable single-label classifiers to indirectly handle multi-label data. The use of simple transformations, label powersets or multiple binary classifiers generally results in ignoring the correlation between labels, as each single-label classifier cannot deal with multiple labels at once, or with more than one pair of labels (in the case of pairwise implementations). This motivated the adaptation of classification algorithms which could handle multi-label data directly, without the need of transforming it into single-label data.

Figure 2.4 represents the process of multi-label classification based on algorithm adaptation, where the multi-label data is given directly to the classifier. Any classifier learner capable of handling multi-label data without any data transformation is placed in this category.

Most traditional classifiers employed in single-label problems have been adapted to the multi-label paradigm [77]. The C4.5 decision-tree learning algorithm has been adapted



Figure 2.4: Multi-label classification based on algorithm adaptation

to handle multi-label data [5] by allowing multiple labels in the leaves of the tree. An SVM algorithm that minimizes the ranking loss metric has been proposed [16]. A multilabel adaptation of the Naive Bayes algorithm was also proposed [83]. MMAC (Multiclass, Multi-label Associative Classification) is an algorithm that follows the paradigm of associative classification which deals with the construction of multi-label classification rule sets using association rule mining [75].

Several k-NN adaptations were proposed [77], and one of them is the Multi-label k-NN [85]. For each unseen instance, it identifies the K nearest neighbors in the training set. Then, based on statistical information gained from the subset of labels of these neighboring instances, the maximum a posteriori (MAP) principle is employed to determine the label set for the unseen instance.

IBLR (Instance Based Learning by Logistic Regression) classifier [4] is an adaptation which combines the instance-based learning concept of the KNN algorithm with logistic regression. It also considers the labels of neighbored instances as features, in order to aid the classification.

The BR + KNN classifier can be adapted by using a single search instead of transforming the multi-label data set using the BR approach. It finds the k nearest neighbors and at the same time it makes independent predictions for each label [67]. While BR followed by k-NN has a computational complexity of L times the cost of computing the k nearest instances, where L is the number of labels in the data set, this adaptation runs much faster, and is more scalable than other classification algorithms based on transformation. So, this algorithm can be considered as an adaptation, as the data set is not transformed. This adaptation is used in this thesis and is referred as BRKNN classifier. When the transformation is used instead, it is referred as BR + KNN classifier.

2.4. Chapter Summary

In this chapter, we review the multi-label classification problem, and summarize the common ways to handle the task. One solution is to transform the multi-label data set into a single-label one. Another solution is adapting a classification technique to deal with multi-label data directly. Among the transformation-based methods, the following have been presented: the copy and select transformations; the label powerset transformation and its extensions, like PPT and RAKEL; and the binary relevance transformation and its extensions, like RPC and Classifier Chains. Among the classifiers based on algorithm adaptation, we presented the Multi-Label k-NN, the Multi-Label Decision Trees and others.

One of the challenges in multi-label classification is the large number of measures that can be used to evaluate the performance of a classifier. In order to settle the problem of which measures to use, in the next chapter a correlation analysis of these measures and guidelines for researchers are presented.

Chapter 3

Correlation Analysis of Performance Measures for Multi-Label Classification

3.1. Introduction

In single-label classification, an instance can be classified either correctly or incorrectly. However, in multi-label classification, an instance can be classified as partially correct, as the predicted label subset can differ, not completely, from the actual label subset that belongs to the instance. So, the evaluation of methods that learn from multi-label data requires different measures than those used in single-label context [77].

In order to evaluate the performance of multi-label classifiers, many measures were adapted from the single-label paradigm, like Precision and Recall; and some were developed specifically for the multi-label paradigm, like Hamming Loss and Subset Accuracy. However, different subsets of measures have been used in multi-label experiments arbitrarily, with the absence of proper justification.

For instance, in [4, 83, 85], the adopted measures for evaluating the proposed algorithms were: Hamming Loss, One Error, Coverage, Ranking Loss and Average Precision. In [78], the measures were Hamming Loss and Example-Based F-Measure. In [3], Micro-Averaged F-Measure and Macro-Averaged F-Measure. In [73], the measures were Hamming Loss and Example-Based Accuracy; and in [68], just the Micro-F Measure was used to report the results. In [46], a total of 16 multi-label measures were used to evaluate a large number of multi-label classifiers.

The adoption of arbitrary measures without an objective analysis of correlation or bias can lead to misleading conclusions, as an experiment evaluated with a subset of measures may appear to perform differently than when evaluated with another subset. Also, as different publications in the area currently employ distinct subsets of measures, it is difficult to compare results across publications. In this chapter, a thorough analysis of multi-label evaluation measures is provided, along with concrete suggestions for researchers to make an informed decision when choosing evaluation measures for multi-label classification.

3.2. Multi-label Measures

Many different evaluation measures specifically developed for multi-label classification have been proposed in the literature. According to [86], these measures can be grouped into example-based and label-based. Also, they can be grouped into classification (or bi-partition) measures and ranking measures. Figure 3.1 illustrates the main multi-label measures according to this categorization.



Figure 3.1: Multi-label evaluation measures categorization [86]

Example-based measures compute the classification performance for each instance, averaging the overall result after classifying all instances. On the other hand, label-based measures decompose the evaluation process into separate values for each label, averaging them subsequently over all labels [77]. These groups and the measures that belong to each one of them are going to be explained in further detail in the following subsections.

Our notation follows the same standards proposed in [74], where a multi-label data set is denoted D, with |D| = N. For each example (x_i, Y_i) , i = 1, ..., N, x_i is the set of feature values and Y_i is the set of true labels, with each element belonging to the set of q labels $L = \{\lambda_j : j = 1, ..., q\}$. Given an instance x_i , the set of predicted labels by a multi-label classifier is denoted by Z_i . The rank of labels predicted by a method is denoted as r_i , and $r_i(\lambda)$ is the rank position of a label λ . The most relevant label receives the highest rank (1), while the least relevant one, receives the lowest rank (q). Additionally, H is the model generated by the multi-label learning task, capable of predicting a subset of labels given an unseen instance.

3.2.1. Example-based Classification Measures

Hamming Loss is one of the most well-known multi-label measures. It takes into account the prediction error (when an incorrect label is predicted) and the missing error (when a relevant label is not predicted), normalized over the total number of classes and the total number of instances [67]. It is defined by Equation 3.1.

$$HammingLoss(H,D) = \frac{1}{N} \sum_{i=1}^{N} \frac{|Y_i \triangle Z_i|}{|L|},$$
(3.1)

where \triangle stands for the symmetric difference of two sets, which is equivalent to the XOR operation in Boolean logic [77].

Subset Accuracy or Exact Match [20] is defined by Equation 3.2. It considers as correct only the examples that are exactly classified, and ignores partially correct values. This is a rigid measure, particularly in the case of data sets of high label cardinality, where it is very hard to achieve an exact match.

Subset Accuracy(H, D) =
$$\frac{1}{N} \sum_{i=1}^{N} I(Y_i = Z_i),$$
 (3.2)

where I is a function that maps a true logic proposition to 1 and false to 0. The Subset 0/1 Loss is similar to Subset Accuracy, but it measures if $Y_i \neq Z_i$. This is equivalent to 1 - Subset Accuracy.

The single-label performance measures Accuracy, Precision, Recall and F-Measure were adapted for the multi-label problem, taking into account partially correct classification.

Accuracy, defined by Equation 3.3, tries to convey the overall effectiveness of a classifier [66].

$$Accuracy(H,D) = \frac{1}{N} \sum_{i=1}^{N} \frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|}$$

$$(3.3)$$

Precision, defined by Equation 3.4, measures the agreement of the labels with the

positive labels given by the classifier.

$$Precision(H,D) = \frac{1}{N} \sum_{i=1}^{N} \frac{|Y_i \cap Z_i|}{|Z_i|}$$
(3.4)

Recall (or Sensitivity), defined by Equation 3.5, measures the effectiveness of a classifier to retrieve positive labels.

$$Recall(H, D) = \frac{1}{N} \sum_{i=1}^{N} \frac{|Y_i \cap Z_i|}{|Y_i|}$$
(3.5)

F-Measure, defined by Equation 3.6, is the harmonic mean of Precision and Recall.

$$F-Measure(H,D) = \frac{1}{N} \sum_{i=1}^{N} \frac{2|Y_i \cap Z_i|}{|Z_i| + |Y_i|}$$
(3.6)

3.2.2. Example-based Ranking Measures

Example-based Ranking measures take into account the label ranking generated by the classifier, averaging the results over all the examples.

One Error evaluates how frequently the top-ranked label is not in the set of the relevant labels of the instance [77], defined by Equation 3.7.

One
$$Error(H, D) = \frac{1}{N} \sum_{i=1}^{N} \delta(argmin \ r_i(\lambda)),$$
 (3.7)

where

$$\lambda \in L, \delta(\lambda) = \begin{cases} 1 & \text{if } \lambda \notin Y_i, \\ 0 & \text{otherwise} \end{cases}$$

Coverage, defined by Equation 3.8, evaluates how far it is needed, on average, to go down the ranked list of labels in order to cover all the relevant labels of the example [77].

$$Coverage(H, D) = \frac{1}{N} \sum_{i=1}^{N} max \ r_i(\lambda) - 1, \qquad (3.8)$$

where $\lambda \in Y_i$.

Ranking loss expresses the number of times that irrelevant labels are ranked higher

than relevant labels [77], given by Equation 3.9.

$$Ranking \ Loss(H,D) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{|Y_i| |\overline{Y_i}|} \{ (\lambda_a, \lambda_b) : r_i(\lambda_a) > r_i(\lambda_b), (\lambda_a, \lambda_b) \in Y_i \times \overline{Y_i} \}, \ (3.9)$$

where $\overline{Y_i}$ is the complementary set of Y_i with respect to L.

Average Precision, defined by Equation 3.10, computes for each relevant label the proportion of relevant labels that are ranked before it, and finally averages over all relevant labels [67].

Average
$$Precision(H, D) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{|Y_i|} \sum_{\lambda \in Y_i} \frac{|\lambda' \in Y_i : r_i(\lambda') \le r_i(\lambda)|}{r_i(\lambda)}$$
 (3.10)

3.2.3. Label-based Classification Measures

Any measure for single-label classification can be adapted as a label-based measure for multi-label classification. The calculation of these measures for all labels can be achieved using two averaging operations, called macro-averaging and micro-averaging [77].

Let T_p , T_n , F_p and F_n denote the true positives, true negatives, false positives and false negatives evaluated by a single-label classifier, respectively. The traditional measures Accuracy, Precision, Recall and F-Measure are given by the Equations 3.11, 3.12, 3.13 and 3.14, respectively.

$$Accuracy(H,D) = \frac{T_p + T_n}{N}$$
(3.11)

$$Precision(H,D) = \frac{T_p}{T_p + F_p}$$
(3.12)

$$Recall(H,D) = \frac{T_p}{T_p + F_n}$$
(3.13)

$$F-Measure(H,D) = \frac{2 T_p}{2 T_p + F_p + F_n}$$
(3.14)

Let B(Tp, Tn, Fp, Fn) be any of these four evaluation measures. The macro-averaged

and micro-averaged versions of B are calculated as defined by Equations 3.15 and 3.16, respectively.

$$B_{macro}(H,D) = \frac{1}{q} \sum_{i=1}^{q} B(Tp_i, Fp_i, Tn_i, Fn_i)$$
(3.15)

$$B_{micro}(H,D) = B(\sum_{i=1}^{q} Tp_i, \sum_{i=1}^{q} Fp_i, \sum_{i=1}^{q} Tn_i, \sum_{i=1}^{q} Fn_i)$$
(3.16)

It is worth mentioning that micro-averaging has the same result as macro-averaging for some measures, such as accuracy; and that Hamming Loss represents the average error, which is equal to 1 minus the value of (macro/micro) accuracy [74].

3.2.4. Label-based Ranking Measures

Area Under the Curve (AUC) or Balanced Accuracy is a statistical measure that corresponds to the total area under the Receiver Operating Characteristic curve. This curve represents the fraction of true positives out of the total actual positives (i.e., Recall) versus the fraction of false positives out of the total actual negatives. It was introduced only recently in multi-label classification, to measure the ability of a classifier to avoid false classification [66].

As the AUC is a measure for the single-label classification domain, it can also be adapted to multi-label by macro-averaging (macroAUC) or micro-averaging (microAUC) its results.

3.2.5. Multi-label Measures in the Literature

In [69], a systematic review on experimental multi-label learning was presented. A total of 64 papers were selected, all of them consisting of publications which report experimental results for multi-label learning research. This work also counted the number of times each measure was used to report a multi-label result. Even though example-based ranking measures were not assessed in this count, a significant number of the 64 papers used them in their experimental reports, including [4, 7, 73, 83, 85, 88], which used One Error, Coverage, Ranking Loss and Average Precision. So we have also counted these specific measures in order to evaluate their use in the literature.

Table 3.1 shows individual results collected using each multi-label measure (in the

Evaluation Measures	#Collected	Number of Papers
Hamming-Loss	1474	55
Accuracy	1016	26
F-Measure	859	18
Precision	623	18
Recall	623	18
Subset-Accuracy	612	10
Ranking Loss	440	8
Coverage	432	8
Micro F-Measure	492	15
One Error	366	7
Macro F-Measure	331	12
Average Precision	262	10
Subset 0/1 loss	240	3
Macro Precision	133	5
Micro Precision	131	4
Micro Recall	129	3
Macro Recall	127	2
Micro AUC	7	1
Macro AUC	7	1

Table 3.1: Statistics for each evaluation measure, adapted from [69]

"#Collected" column) and the number of times each multi-label measure was employed by a selected work (in the "Number of Papers" column).

The Hamming Loss measure was the most used to evaluate the multi-label experiments, in 55 out of the 64 publications. The other example-based measures were adopted in a range varying between 10 and 26 publications, like the micro F-Measure in 12 publications and the macro F-Measure was used in 15 publications. The most used ranking-based measure was Average Precision, in 10 publications. Micro and Macro AUC were used only in 1 publication assessed in the survey.

3.3. Correlation among Multi-label Measures

In the previous section it was showed that many multi-label measures exist and are employed together in a large number of publications. However, each work decides on a different small subset of these measures to report their experiments, or elects a large number of measures, like the survey conducted in [46], which employed sixteen measures. If the subset of chosen measures contains measures that are strongly correlated, while leaving out other important measures that are not correlated, the experimental evaluation can be misleading.

To the best of our knowledge, this correlation analysis has not been considered so far in the literature. This analysis is provided below, by evaluating the correlation among measures in a large number of multi-label experiments. The analysis compared the results from multi-label classifiers in order to find out which multi-label measures are correlated

			feat	ures	labels					
name	domain	instances	nominal	numeric	distinct	$\operatorname{cardinality}$				
bibtex	text	7,395	1,836	0	159	2.402				
birds	audio	645	2	258	19	1.014				
CAL500	music	502	0	68	174	26.044				
corel5k	images	5,000	499	0	208	2.028				
emotions	music	593	0	72	6	1.869				
enron	text	1,702	1,001	0	53	3.378				
flags-ml	images	194	9	10	7	3.392				
$_{genbase}$	biology	662	1186	0	27	1.252				
medical	text	978	1449	0	45	1.245				
scene	image	2,407	0	294	6	1.074				
yeast	biology	2417	0	103	14	4.237				

Table 3.2: Multi-Label data sets used for analysis of measures

to each other. Commonly used multi-label data sets and classification algorithms were elected, and the experiments were executed using the Mulan framework [77]. Mulan is an open-source Java library for learning from multi-label data sets, built on top of the Weka tool [79]. The library includes a variety of state-of-the-art algorithms for performing: multi-label classification, ranking and a few simple feature selection techniques.

Currently, there is a limited number of publicly available multi-label data sets. Most of the initiatives that compare multi-label learning algorithms experimentally adopt a subset of these available data sets. In Table 3.2, the data sets which are used in this work to evaluate multi-label classification algorithms are provided. The first two columns show the name and the domain associated to the data set. The "instances" column presents the number of instances of the data set. The "features" columns show the number of nominal and numeric features. The "distinct" subcolumn in the "labels" column shows the number of distinct labels over all instances in the data set, and the "cardinality" subcolumn represents the average number of labels of each instance, which in the multilabel setting is always greater than 1.

Mulan contains an evaluation framework that calculates a rich variety of performance measures [77]. Sixteen measures were chosen, among the most commonly adopted in articles related to multi-label classification and feature selection, and which were described in the previous sections, i.e.: Hamming Loss, Subset Accuracy (equivalent to Subset 0/1 loss), Accuracy, F-Measure, Precision, Recall, Micro F-Measure, Macro F-Measure, Micro Precision, Macro Precision, Micro Recall, Macro Recall, Coverage, One Error, Average Precision and Ranking Loss. The Micro and Macro AUC were not selected because they do not have a widespread use in the multi-label literature. The selected measures are also the same used in [46] for the evaluation of multi-label classification techniques.

A large number of classification techniques have been employed, from both transfor-

mation and algorithm adaptation paradigms. Table 3.3 summarizes and categorizes the multi-label classifiers used for the evaluation. The transformation techniques used were: Label Powerset, Binary Relevance, Classifier Chains, PPT, RaKEL and HOMER, coupled with the k-NN, decision trees (J48) and Naive Bayes single-label classifiers. The algorithm adaptations employed in this experiment were the ML-kNN, the IBLR classifier and the BRKNN adaptation. These are common multi-label classifiers used in the literature [46].

Paradigm	Classifier	Base Classifier				
		k-NN				
	Binary Relevance	Decision Tree				
		Naive Bayes				
		k-NN				
	Label Powerset	Decision Tree				
		Naive Bayes				
Data Transformation		k-NN				
	Classifier Chains	Decision Tree				
		Naive Bayes				
		k-NN				
	RaKEL	Decision Tree				
		Naive Bayes				
	HOMER	k-NN				
	PPT	k-NN				
	ML-KNN	not applicable				
Algorithm Adaptation	IBLR	not applicable				
	BRKNN adaptation	not applicable				

Table 3.3: Multi-label classifiers used in the experiments

3.3.1. Analysis of Pearson Correlation

For the first correlation analysis, an execution for all classifiers (16) was performed, using all data sets (11) and using a total of 16 different measures for each case. This leads to more than 2,500 results (16*11*16). After consolidating these results by averaging them, each pair of measures were compared using Pearson Correlation, considering the group of results achieved for every classifier and data set. Pearson's correlation coefficient is given by:

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_x \sigma_y},\tag{3.17}$$

where cov is the covariance and σ_x is the standard deviation of X.

Table 3.4 summarizes the results. Pearson's correlation coefficient varies from -1 to 1. The absolute correlation value is reported, since a negative correlation is as important as a positive correlation, especially considering that some measures are the complement of others. The results that achieved a value equal or greater than 0.8 are marked in bold. The matrix is mirrored around its main diagonal.

Multi-label measure	HL	SA	ΕbΑ	EbF	EbP	EbR	$_{maF}$	MaF	$^{\mathrm{maP}}$	MaP	$_{maR}$	MaR	AP	C_{0}	OE	RL
Hamming Loss (HL)	1	0.41	0.07	0.07	0.01	0.21	0.10	0.01	0.21	0.08	0.32	0.15	0.10	0.01	0.01	0.25
Subset Accuracy (SA)	0.41	1	0.88	0.79	0.77	0.67	0.73	0.78	0.71	0.80	0.56	0.65	0.67	0.23	0.54	0.52
Ex-Based Accuracy (EbA)	0.07	0.88	1	0.98	0.95	0.88	0.94	0.88	0.79	0.89	0.79	0.77	0.86	0.23	0.71	0.48
Ex-Based F Measure (EbF)	0.07	0.79	0.98	1	0.97	0.91	0.96	0.86	0.76	0.86	0.83	0.76	0.87	0.22	0.73	0.44
Ex-Based Precision (EbP)	0.01	0.77	0.95	0.97	1	0.81	0.94	0.79	0.82	0.83	0.72	0.64	0.85	0.22	0.77	0.43
Ex-Based Recall (EbR)	0.21	0.67	0.88	0.91	0.81	1	0.87	0.79	0.54	0.74	0.95	0.85	0.81	0.25	0.63	0.45
Micro-avg F-Measure (maF)	0.10	0.73	0.94	0.96	0.94	0.87	1	0.90	0.80	0.90	0.86	0.79	0.90	0.33	0.81	0.41
Macro-avg F-Measure (MaF)	0.01	0.78	0.88	0.86	0.79	0.79	0.90	1	0.71	0.98	0.79	0.91	0.83	0.26	0.71	0.40
Micro-avg Precision (maP)	0.21	0.71	0.79	0.76	0.82	0.54	0.80	0.71	1	0.79	0.48	0.49	0.77	0.25	0.75	0.44
Macro-avg Precision (MaP)	0.08	0.80	0.89	0.86	0.83	0.74	0.90	0.98	0.79	1	0.71	0.82	0.84	0.25	0.72	0.41
Micro-avg Recall (maR)	0.32	0.56	0.79	0.83	0.72	0.95	0.86	0.79	0.48	0.71	1	0.90	0.80	0.36	0.66	0.41
Macro-avg Recall (MaR)	0.15	0.65	0.77	0.76	0.64	0.85	0.79	0.91	0.49	0.82	0.90	1	0.76	0.29	0.60	0.42
Average Precision (AP)	0.10	0.67	0.86	0.87	0.85	0.81	0.90	0.83	0.77	0.84	0.80	0.76	1	0.40	0.89	0.68
Coverage (Co)	0.01	0.23	0.23	0.22	0.22	0.25	0.33	0.26	0.25	0.25	0.36	0.29	0.40	1	0.40	0.36
OneError (OE)	0.01	0.54	0.71	0.73	0.77	0.63	0.81	0.71	0.75	0.72	0.66	0.60	0.89	0.40	1	0.64
Ranking Loss (RL)	0.25	0.52	0.48	0.44	0.43	0.45	0.41	0.40	0.44	0.41	0.41	0.42	0.68	0.36	0.64	1

Table 3.4: Multi-label measures compared with Pearson correlation

Depending on the domain, a value greater than 0.5, 0.6 or 0.7 is considered a high correlation between two variables. In this work a conservative value of 0.8 was adopted to represent highly correlated measures, and above 0.9 for very high correlation. In Figure 3.2, the correlation results are presented in a graphical representation. Each dashed line connecting two measures means that the measures presented a correlation equal or greater than 0.8. Bold lines represent a correlation greater or equal than 0.9.

According to the criteria adopted, most of multi-label measures are highly correlated (Pearson correlation ≥ 0.8) with at least another one, a few are not correlated with any other (Hamming Loss, Coverage and Ranking Loss) and some of them are correlated with only a few others (Micro-averaged Precision, Micro-averaged Recall, Subset Accuracy and One Error). The following measures are strongly correlated (Pearson correlation ≥ 0.8) with eight other measures: Example-Based F-Measure, Average Precision, Example-Based Precision and Example-Based Accuracy.

Additionally, further analysis showed that by varying the subset of classifiers used in the experiments, the end result varied by less than 5% on average, and the graphical representation of correlation between measures still holds.

3.3.2. Analysis of Spearman Correlation

Spearman's rank correlation coefficient is also used to measure the strength of association between two variables. It is used in this work as an alternative analysis of correlation between multi-label measures. While Pearson correlation is computed on the true values of the variables and depicts linear relationships, the Spearman correlation is computed on ranks and depicts monotonic relationships. It also does not make any assumptions about the frequency distribution of the variables [90].

Consider that n is the sample size and x_i and y_i depict the rank of the variable scores



Figure 3.2: Graphical representation of pairwise correlation between evaluation measures X_i and Y_i , respectively. The Spearman correlation coefficient is given by:

$$\rho = \frac{6\sum d_i^2}{n(n^2 - 1)},\tag{3.18}$$

where $d_i = x_i - y_i$ is the difference between ranks.

Analogously to the previous table, Table 3.5 summarizes the results of the Spearman correlation coefficient, which also varies from -1 to 1. The absolute values are reported, and the results that achieved a value greater or equal than 0.8 are marked in bold.

As can be seen in the results, the Spearman correlation coefficient achieves values directly comparable to the previous Pearson correlation analysis. In fact, considering the 0.8 threshold, with the exception of a few pairs, the coefficient for the same measures from both tables are marked in bold. This reinforces the notion that there are measures

Multi-label measure	HL	SA	ЕbА	EbF	EbP	EbR	$_{maF}$	MaF	maP	MaP	$_{\rm maR}$	MaR	ΑP	C_{0}	OE	RL
Hamming Loss (HL)	1	0.31	0.05	0.06	0.00	0.15	0.08	0.05	0.23	0.10	0.24	0.08	0.03	0.04	0.05	0.45
Subset Accuracy (SA)	0.31	1	0.87	0.77	0.75	0.64	0.72	0.78	0.69	0.82	0.53	0.64	0.71	0.64	0.49	0.55
Ex-Based Accuracy (EbA)	0.05	0.87	1	0.97	0.93	0.84	0.91	0.87	0.74	0.88	0.74	0.74	0.84	0.55	0.67	0.47
Ex-Based F Measure (EbF)	0.06	0.77	0.97	1	0.95	0.89	0.95	0.88	0.72	0.87	0.82	0.77	0.86	0.55	0.72	0.42
Ex-Based Precision (EbP)	0.00	0.75	0.93	0.95	1	0.77	0.91	0.80	0.79	0.84	0.69	0.64	0.83	0.48	0.74	0.41
Ex-Based Recall (EbR)	0.15	0.64	0.84	0.89	0.77	1	0.84	0.78	0.51	0.72	0.95	0.88	0.79	0.56	0.62	0.43
Micro-avg F-Measure (maF)	0.08	0.72	0.91	0.95	0.91	0.84	1	0.91	0.78	0.91	0.84	0.79	0.91	0.62	0.80	0.39
Macro-avg F-Measure (MaF)	0.05	0.78	0.87	0.88	0.80	0.78	0.91	1	0.69	0.97	0.76	0.88	0.85	0.67	0.69	0.43
Micro-avg Precision (maP)	0.23	0.69	0.74	0.72	0.79	0.51	0.78	0.69	1	0.78	0.48	0.49	0.77	0.48	0.73	0.47
Macro-avg Precision (MaP)	0.10	0.82	0.88	0.87	0.84	0.72	0.91	0.97	0.78	1	0.70	0.81	0.87	0.66	0.71	0.44
Micro-avg Recall (maR)	0.24	0.53	0.74	0.82	0.69	0.95	0.84	0.76	0.48	0.70	1	0.91	0.79	0.62	0.65	0.38
Macro-avg Recall (MaR)	0.08	0.64	0.74	0.77	0.64	0.88	0.79	0.88	0.49	0.81	0.91	1	0.77	0.68	0.58	0.44
Average Precision (AP)	0.03	0.71	0.84	0.86	0.83	0.79	0.91	0.85	0.77	0.87	0.79	0.77	1	0.69	0.88	0.62
Coverage (Co)	0.04	0.64	0.55	0.55	0.48	0.56	0.62	0.67	0.48	0.66	0.62	0.68	0.69	1	0.53	0.48
OneError (OE)	0.05	0.49	0.67	0.72	0.74	0.62	0.80	0.69	0.73	0.71	0.65	0.58	0.88	0.53	1	0.55
Ranking Loss (RL)	0.45	0.55	0.47	0.42	0.41	0.43	0.39	0.43	0.47	0.44	0.38	0.44	0.62	0.48	0.55	1

Table 3.5: Multi-label measures compared with Spearman correlation

which are strongly correlated, and others that are more independent.

3.3.3. Guidelines for Choosing Measures in a Multi-label Setting

To create guidelines for choosing measures, we carefully analyze these correlation results, assuming that two strongly correlated measures should not appear together in an overall report of the performance of a multi-label technique. We also take into account three other criteria: popularity in the literature, the choice of including or not label ranking measures and the number of desired measures, suggesting the adoption of the following guidelines:

- (a) Small set of measures: the smaller choice of measures should consider employing the measures that are not correlated with others (Hamming Loss, Coverage and Ranking Loss), and one from the main cluster of correlated measures (either Example-Based F-Measure or Example-Based Accuracy, because they are the ones with most correlations to other measures). This is equivalent to choosing one representative measure for each connected component of the graph in Figure 3.2.
- (b) Small set of measures discarding ranking-based measures: alternatively, one could discard one or both ranking-based measures (Coverage and Ranking Loss) if the focus is not on producing rankings of labels even though these measures could still be useful in a pure multi-label classification task, by evaluating how far incorrectly predicted labels are to the actual subset of labels.
- (c) Exact match evaluation: Subset Accuracy is a measure commonly used in the literature and evaluates the capacity of one algorithm to yield an exact set of correctly predicted labels. The alternative measure Subset 1/0 Loss is almost equivalent to it (yielding 1 - SubsetAccuracy score), and can be used instead.
(d) Focus on ranking measures: Ranking Loss and Coverage are not correlated with any other measure, and should be adopted for researchers dealing specifically with the multi-label ranking task. One Error and Average Precision are also rankingbased measures, but are correlated with each other (not strongly). As they are both commonly used in the literature, we suggest the following set of measures: Ranking Loss, Coverage, One Error, Average Precision, Hamming Loss and, optionally, Subset Accuracy.

The same intuition can be used for selecting other subset of measures. For instance, if one needs to evaluate the Recall or the Precision of an algorithm, the use of F-Measure should be avoided, because it is strongly correlated with them. The only exception should be made for measures used extensively in the related literature, for comparison purposes.

Throughout this work, we elected the following measures to evaluate our algorithms: Hamming Loss, Subset 1/0 Loss, Example-Based Accuracy and Ranking Loss.



Figure 3.3: Graphical representation of the 12 lesser correlated measures

3.4. Chapter Summary

In this chapter we have presented a correlation analysis of multi-label measures used in the literature. We argue that the adoption of these measures in multi-label experimental comparisons without an objective criteria can lead to biased conclusions due to correlation between measures.

The main contribution of this chapter is to provide a comprehensive and detailed analysis of the correlation that exists between multi-label measures by experimenting with multiple classification techniques and data sets from various domains.

Based on this analysis, we are able to take an informed decision when choosing performance measures for evaluating multi-label classification.

In the next chapter, we begin our feature selection review and contribution in the multi-label setting. We propose a taxonomy for categorizing multi-label feature selection techniques and review the current literature on the topic.

Chapter 4

Multi-label Feature Selection

4.1. Introduction

According to [26], feature selection techniques are employed to identify relevant and informative features, primarily to improve the classifier predictive accuracy. In general, besides this main goal, there are other important motivations: the reduction and simplification of the data set, the acceleration of the classification task, the simplification of the generated classification model, and others.

Traditional feature selection techniques can generally be categorized into three approaches: embedded, wrapper or filter [42]. Embedded strategies are incorporated into the algorithm responsible for the induction of the classification model. Decision tree induction algorithms can be viewed as having an embedded feature selection technique, since they internally select the features that will be tested at each node of the generated tree.

Wrapper and filter strategies are performed in a preprocessing phase and they search for the most suitable feature set to be used by the classification algorithm or by the classification model inducer. In wrapper feature selection, the adopted classification algorithm itself is used to evaluate the quality of candidate feature subsets, while in filter feature selection, feature quality is evaluated independently from the classification algorithm using a measure which generally takes into account the feature and class label distributions. Among common filter measures, there are those that evaluate each feature individually, as exemplified by Information Gain Ranking [54] and Relief [31, 35]; and measures that evaluate subsets of features, combined with a heuristic search for finding the best subset, like the Correlation-based Feature Selection [28] and Consistency-based Feature Selection [43]. There are also hybrid strategies which try to combine the wrapper and the filter approaches [44].



Figure 4.1: Taxonomy proposed for multi-label feature selection

Feature selection techniques intended specifically for multi-label classification have been developed in recent years. Even though there are many publications on this topic, it is still considered an active research area [14, 68] and, to the best of our knowledge, there is no work that surveys or categorizes the current multi-label feature selection techniques.

In the next subsections, we propose a novel taxonomy for multi-label feature selection, and based on this taxonomy, we review the feature selection techniques for multi-label classification that have been proposed in the literature.

One of the contributions of the thesis is a comprehensive survey and a taxonomy of multi-label feature selection techniques. Figure 4.1 shows our proposed taxonomy for categorizing multi-label feature selection. It aims at categorizing the feature selection techniques according to characteristics inherent to the multi-label paradigm.

This taxonomy is composed of two main categories based on the multi-label classification paradigms already explained in our work: transformation-based methods and direct methods. The transformation-based and direct categories are described in the next sections.

4.2. Multi-label Feature Selection Based on Transformation

The simplest way to employ feature selection to a multi-label data set is to change it into a single-label data set and apply a traditional feature selection technique. There are plenty of algorithms to transform a multi-label data set into a single-label one.

Methods to transform a multi-label data set into single-label data were described in Chapter 2, in the multi-label classification context. In the next sections, we present the copy, select, label-powerset and binary relevance transformations in the feature selection context, and review previous work that employed these methods.

4.2.1. Strategies Based on Single Data Transformation

Single data transformation for multi-label feature selection consists of changing the multilabel data into one single-label data set and applying a traditional feature selection technique. Single data transformation encompasses both simple and label powerset transformations.

The following common simple transformation techniques: select-max, select-min, select-random, select-ignore, copy and copy-weight; and the label powerset transformation, used to convert a multi-label data set into a single-label one were described in Chapter 2. These transformations have also been employed to perform feature selection over multi-label data.

Figure 4.2 presents a feature selection model to represent this category of transformations applied to the multi-label data. It initially converts the original multi-label data into a single-label data set using one of the transformations. Then a traditional singlelabel feature selection is employed to the data. The output of this process is a list of the selected features. Optionally, a subsequent process – indicated with dashed lines – can be employed to deliver the original multi-label data containing only the corresponding selected features. This way a multi-label classifier can be used to perform its predictions over the multi-label data.

In [3], these data set transformations were used to allow the application of traditional feature selection techniques to the text categorization problem. According to the model in Figure 4.2, the multi-label data were transformed into a single-label data set after executing the following simple transformations: copy, select-ignore, select-max and select-



Figure 4.2: Transformation Based/Single multi-label feature selection

min. They also proposed a new transformation – from multi-label into single-label data – based on the entropy measure, which reweights each instance using this metric as a variation of the copy-weight transformation described before in subsection 2.2.1.

Note that after employing a feature selection technique, it is possible to deliver to the classifier either the transformed single-label data set, or the original multi-label data set maintaining only the corresponding selected features. In the latter case this is required if it is necessary to run a multi-label classifier after the feature selection. Nonetheless, as these feature selection techniques based on a simple data transformation disregard the correlation between labels or subsets of labels, they might fail to identify a suitable feature set for correct classification of some specific instances.

The label powerset transformation is also directly applied to the task of multi-label feature selection based on transformation, as it is capable of delivering a single-label data set with each subset of labels converted into a new class label.

In [73], several multi-label classification strategies were evaluated and compared for the task of automated decision of emotion in a music data set. For the empirical evaluation of feature selection, the use of a label powerset transformation was proposed to produce a single-label data set, and then a common feature selection measure was employed (χ^2 statistic) to select the best features. They verified that, for the evaluated data set, using the ML-KNN algorithm [85] as the classifier and the label powerset to apply the feature selection achieved a better Hamming Loss result than without feature selection.

The label powerset transformation is also used for feature selection in [68], in conjunction with the relief and information gain measures. With this feature selection, it was possible to reduce the dimensionality of the data sets without compromising the classification performance.

The label powerset transformation tends to create too many classes, causing overfitting and imbalance problems [37]. In [14], the pruned problem transformation (PPT) [56], an improvement over the label powerset, was used for multi-label feature selection on three real-world data sets from different domains: gene, semantic scene and emotion (in music) classification. Then a multi-label k-NN algorithm was employed over the original multilabel data containing only the selected features. When compared with the χ^2 statistic adopted in [73], and also with a non feature selection scenario, the mutual information measure allowed the classification phase to achieve a better result in terms of the Hamming Loss and the accuracy of the classifier.

The feature selection techniques can be categorized as wrapper, embedded or filter. An algorithm from any of these categories can be applied after a single data transformation. However, all publications reviewed in this subsection are categorized as Transformation Based/Single/Filter. This means that there is a lack of work evaluating single-label embedded and wrapper feature selection techniques for multi-label classification.

4.2.2. Strategies Based on Binary Relevance Transformation

A different line of attacking the multi-label feature selection problem is to transform the multi-label data set into several single-label data sets and use existing feature selection methods on each data set, particularly those that follow the filter paradigm [77].

The process of transforming a multi-label data set into several single-label ones was explained in Chapter 2. The same technique can be employed for feature selection. For each different label in the original data set, a binary single-label data set is created, and then feature selection is executed.

Figure 4.3 represents a feature selection model based on the binary relevance (BR) transformation. Each label from the data set is considered individually in order to perform the feature selection. Then the single-label feature selection is applied once for each single-label data set.

There are two ways to handle the feature selection result on a BR approach. The first one, is to apply the classification method directly to each single-label data set obtained after the feature selection step. We call this the *Internal* approach. As the multi-label data set is transformed into a single-label data set, both the classifier and the feature selection techniques are able to handle the data. After the feature selection, each reduced



Figure 4.3: Transformation Based/Binary Relevance multi-label feature selection

single-label data set will serve as input for a single-label classifier. After the classification step, the results are combined analogously to Figure 2.3 in Chapter 2.

Another way to handle the feature selection result of each binary model, which we call the *External* approach, is to combine the results from each feature selection into a single output, and then output the reduced data set to a multi-label classifier. In this case there is the need to aggregate the feature selection results before classification.

A typical way to output a list of selected features is ensuring a score threshold or a fixed number of features across the rankings (e.g., the top 500 features). Other ways to combine the multiple feature rankings produced by the binary classifiers is to consider the overall maximum score or the average score of each feature across the binary models [73]. The feature selection used in this External strategy can be a filter or a wrapper technique.

In [18], a round robin aggregation method was proposed, which considers the best features of each binary model in sequence, and a variation named rand-robin, that selects the best features in a roulette fashion inversely to the frequency of each label in the original data set. The process of combining the lists of features is also known as aggregation. This is the approach shown in Figure 4.3. After the aggregation process, indicated by a dashed line, there is an optional step of removing the features from the original multi-label data set to produce a corresponding data set with the chosen features only. In [80], common feature selection measures (document frequency, information gain, mutual information, χ^2 statistic and term strength) were evaluated in a text categorization multi-label problem. Each label was evaluated separately, which is equivalent to an external binary relevance transformation. After applying the feature selection to this data set, the k-NN classification technique was employed. Up to 98% of the features were removed without losing categorization accuracy, when using the information gain and χ^2 techniques; the same result occurred when 90% of features were removed with the document frequency metric; and 50-60% with term strength. Mutual information achieved an inferior performance compared to the other methods.

Some text classification work [49, 89] employed the binary relevance technique to apply single-label feature selection measures, like information gain and χ^2 statistic.

In [62], several filter feature selection techniques were applied in text categorization data sets. Again, each label was considered individually, which is equivalent to a BR transformation. Then the following feature selection measures were applied to the data sets: document frequency, information gain, a binary version of information gain and the χ^2 statistics. From the resulting feature ranking of each measure, both the average and the maximum value were considered as an aggregated score. The empirical results showed in [62] suggested that combining the use of multiple feature selection was advantageous for eliminating rare words in a consistent way across different classifiers. In the experimental evaluation on [14], the max and avg aggregation strategies were also used for the BR.

The BR transformation is also used for feature selection in [68], in conjunction with the relief and information gain measures. This feature selection strategy is compared with the LP transformation using the same measures, with the conclusion that both methods achieved a similar performance in the experiments with data sets from various domains commonly used in multi-label work.

In [71], BR was used to apply feature selection in conjunction with several aggregation techniques to data sets from the text categorization domain. The best results were achieved by using the maximum score across all labels with the χ^2 measure.

In [78], the RAKEL method was proposed and evaluated on three data sets from different domains (semantic scene, gene and textual classification). In the data transformation step, RAKEL constructs an ensemble of label powersets. Then feature selection was applied to the textual data set to reduce the computational cost of training. The χ^2 statistic was used separately for each label in order to obtain different rankings of all features, and in the aggregation step the top 500 features were selected (i.e., the features with the highest score over all labels). This same label-based approach was applied in [57] for a text-categorization data set (Reuters) in conjunction with the information gain measure.

As it occurs with single transformation-based feature selection, there is a lack of work evaluating embedded and wrapper feature selection techniques after a BR transformation. All techniques described in this subsection are categorized as Transformation Based/BR/Filter/External, except for [13], which is categorized as Transformation Based/BR/Filter/Internal.

As it occurs with classification, the use of binary relevance transformation can cause a loss of information from the multi-label data, like label dependence, an important issue in multi-label learning [69].

4.2.3. Summary of Publications on Transformation Based Feature Selection

Table 4 shows publications related to multi-label feature selection that rely on data transformation. The "Data Transformation" column specifies which transformation technique described in our taxonomy was used. In the case of the binary relevance transformation, we also specify how the multiple lists of features were combined (indicated by the '+' sign): either using the average or maximum score, in the case of the *Internal* approach, or selecting a specific number of top features, in the case of the *External* approach.

The "Feature Selection" column indicates which feature selection technique was used – all of them single-label techniques, relying on the data transformation executed before. The "Classifier" column shows which classification strategy was employed, in some cases combined with some data transformation technique, indicated by the '+' sign (e.g., RA-KEL + SVM). Finally, the "Data Sets (domain)" column lists the data sets used and to which domains they belong, in parentheses.

We observe that most publications employed a data transformation technique from just one paradigm (simple transformations, label powerset-based or binary relevancebased). The BR approach is usually External. The feature selection strategies used are simple filters that evaluate one feature at a time. Furthermore, most of them aim to evaluate one or two classifiers, using data sets from just one or a few multi-label domains.

Publication	Data Transformation	Feature Selection	Classifier	Data Sets (domain)
[3]	Single/Filter/copy Single/Filter/select-ignore Single/Filter/select-min Single/Filter/select-max Single/Filter/entropy-based	information gain χ^2 statistic OCFS	SVM	Reuters-21578 (text) Reuters RCV1-v2 (text)
[80]	BR/External/Filter +top features	document frequency information gain mutual information χ^2 statistic term strength	k-NN Regression (LLSF)	OSHUMED (text) Reuters-22173 (text)
[73]	Single(LP)/Filter BR/External/Filter + avg and max	χ^2 statistic	Multi-Label k-NN	Emotions (music)
[89]	BR/External/Filter	χ^2 statistic information gain correlation coefficient odds ratio	Naive Bayes Logistic Regression	Reuters-21578 (text)
[14]	Single(PPT)/Filter Single(LP)/Filter	$rac{1}{\chi^2}$ statistic	Multi-Label k-NN SVM	Yeast (gene) Scene (image) Emotions (music)
[49]	BR/External/Filter +max, avg and min	document frequency χ^2 statistic information gain	k-NN	RCV1-v2 (text)
[62]	BR/External/Filter + avg and max	document frequency information gain binary information gain χ^2 statistic	k-NN Naive Bayes SVM Rocchio	Reuters-21578 (text) Reuters RCV1 (text)
[78]	BR/External/Filter + top 500 features	χ^2 statistic	RAKEL+SVM BR+SVM	tmc2007 (text)
[13]	BR/Internal/Filter	information gain	${}^{ m BR+kNN}_{ m BR+SVM}$	Reuters RCV1 (text) EUROVOC (text)
[57]	BR/External/Filter + top 500 features	information gain	BR+Naive Bayes LP+Naive Bayes	Reuters RCV1 (text)
[68]	Single(LP)/Filter BR/External/Filter	information gain relief	BRKNN	Various Domains
[71]	BR/External/Filter +avg,max,round/rand-robin	χ^2 statistic bi-normal separation	BR+SVM	Various Domains
[6.4]	Single(LP)/Filter	mutual information maximization	Malli Tabal I. NIN	Scene (image)
[04]	${ m BR/External/Filter}\ + \max$	joint mutual information max.	wiulti-Label K-NN	Yeast (gene)

Table 4.1: Summary of publications on ML feature selection based on transformation

4.3. Direct Multi-label Feature Selection

Several feature selection techniques were proposed to deal directly with the multi-label data. They consist mostly of algorithm adaptations of well-known feature selection techniques. Unlike the previous categories, in this case there is no transformation of the multi-label data. In [38], a feature selection algorithm based on this model is claimed to perform better than common techniques that transform the multi-label data into single-label.

We will categorize these multi-label feature selection techniques in three sub-categories: *Filter*, *Wrapper* and *Embedded*, in the same way they are categorized for their single-label counterparts [42], and according to our proposed taxonomy.

4.3.1. Strategies Based on the Filter Strategy

Filter strategies generally use an evaluation function which depends only on the properties of the data set, so they are independent of any particular learning algorithm.

Figure 4.4 illustrates this approach, which typically employs a heuristic search strategy

and a metric able to evaluate subsets of features. The heuristic search can be also a ranker which evaluates each feature individually by a specific metric. Afterwards, the ranking is processed to output the selected features, either by establishing a metric value threshold or selecting the top n features from the ranking. Then this result can be combined with the original multi-label data to produce a new data set with only the selected features.



Figure 4.4: Direct/Filter multi-label feature selection

In [36], the well-known technique FCBF (Fast Correlation-Based Filter), introduced in [82], is extended to handle multi-label data. The technique consists of transforming the data set into a directed graph and applying the symmetrical uncertainty measure to evaluate the features of the data set. This feature selection is applied in conjunction with the IBLR-ML [4] and ECC [59] classifiers, and data sets from multiple domains are evaluated.

Some feature extraction techniques were adapted from single-label counterparts, like PCA (Principal Component Analysis) and LSI (Latent Semantic Indexing). They produce a ranking of features after applying a technique to reduce the number of features, either by removing irrelevant features, or by creating a projection of the feature space. For instance, in [81] it is proposed the Multi-label Latent Semantic Indexing (MLSI). It is a feature extraction technique based on dimensionality reduction, as an extension of the LSI technique to make use of multi-label information. Feature extraction is a task different from feature selection [41], so it is not the focus on this thesis.

In [50], a multi-label filter adaptation based on the information gain measure was proposed. The technique was evaluated on various multi-label data sets and coupled with ML-KNN, BR-KNN, CC, and other classifiers. It achieved an overall better result than the LP and copy transformations, and competitive results against the BR transformation. In terms of scalability, the multi-label information gain filter outperformed the other transformation-based techniques when coupled with the BR-KNN classifier and assessed with data sets from the Yahoo directory (more than 30,000 features). In [39], an adaptation based on the information gain measure was proposed, and the experimental results confirmed it as an effective approach compared with other feature selection techniques.

Similarly, in [88], the MDDM method – Multi-label Dimensionality reduction via Dependence Maximization – is proposed. It consists of a dimensionality reduction method, like PCA, aimed to the multi-label domain. It creates a ranking of features by maximizing the dependence between the features and the associated class labels using a well-known dependence measure. It is compared with other similar methods like PCA and MLSI coupled with the multi-label k-NN classifier and eleven Yahoo web-pages data sets.

Filter strategies can also consider subsets of features instead of single features. After a number of iterations, the feature subset with the best metric value is selected. Like in other multi-label feature selection techniques, a subsequent process can be employed to deliver the original multi-label data with the corresponding selected features.

In [34], a new multi-label feature selection technique designed for graph classification is proposed, called gMLC. It is based on an efficient search for optimal subgraph features for graph objects with multiple labels, and evaluates each subset with a particular criteria. Graph data sets are evaluated with this method and compared with a BR transformation coupled with the information gain measure, and also with a technique that selects the top k-frequent subgraph features.

Common single-label feature selection techniques were adapted to the multi-label paradigm recently. The reliefF measure was adapted in [53] and [69]. The mutual information measure was adapted in [37]. The correlation-based feature selection technique, capable of handling subset of features, instead of individual features, was adapted to the multi-label setting in [30].

4.3.2. Strategies Based on the Wrapper Strategy

The wrapper approach for feature selection [33] consists in a method that searches for a relevant subset of features and employs a classification technique to evaluate it. In other words, given a multi-label learning algorithm, the method searches for the subset of features that optimizes a multi-label measure function on the training data set [77].

Figure 4.5 shows a suitable model for the wrapper paradigm, that is capable of handling multi-label data directly. It works as follows: the data set is submitted to a heuristic search algorithm, and for each selected subset of features, the classification algorithm is used to evaluate it. The best subset of features according to the classification performance is then selected.

Note that in a wrapper approach the adopted classifier can belong to any one of the multi-label categories described in the multi-label classification sections.



Figure 4.5: Direct/Wrapper multi-label feature selection

In [83], a wrapper technique is used over the data set to identify the best feature set. The wrapper feature selection implements a genetic algorithm as the search component. To evaluate this method, the Multi-label Naive Bayes classifier – proposed in the same work – is employed to select the best features. The classification achieved a better result when coupled with the feature selection. Also, it achieved a better performance when compared with the following classifiers: ADTBoost.MH, Rank-SVM, BR+Naive Bayes and CNMF (Constrained Non-negative Matrix Factorization) [45].

In [65], the HOML – Hybrid Optimization based Multi-Label feature selection – is proposed. It consists of a hybrid wrapper feature selection technique, combining simulated annealing, genetic algorithm and hill-climbing to optimize the search for an optimal subset of features. HOML is compared with other wrapper algorithms that employ the following heuristic search algorithms: simulated annealing, forward selection, backward selection and genetic algorithms; all of them coupled with the following base classifiers: ML-KNN [85], BP-MLL [84], Rank-SVM [16] and MLNB [83]. Experimental results on two multi-label data sets favor the HOML technique.

4.3.3. Strategies Based on the Embedded Strategy

There is also the case of embedded feature selection algorithms, where the classification process itself performs the feature selection naturally as part of learning. Techniques like decision trees [54, 55] are examples of classification algorithms that employ an embedded feature selection strategy. In order to build a decision tree model, the learning algorithm selects the features to produce the branches, and the leaves of the tree represent the class labels.

Other examples of classifier learning algorithms with embedded feature selection are neural networks, random forests and feature selection using the weight vector of SVM classifiers [27]. There is not a general model for the embedded strategy, as the selection of an optimal subset of features is built into the classifier construction [63], so it is highly dependent on the classifier.

In [5], a multi-label decision tree was proposed as an extension of the C4.5 algorithm, by allowing multiple labels in the leaves of the tree. In [12], a multi-label boosting algorithm was combined with decision trees to produce a novel method – ADTBoost.MH – capable of handling multi-label data.

In [38], the PRECOMN technique is proposed, based on a previous technique named MEFS (Multi-label Embedded Feature Selection). It consists in an embedded technique coupled with the ML-KNN algorithm. It combines the sequential backward search algorithm with an evaluation measure, named prediction risk criterion, to evaluate the subset of features. The technique is evaluated on one data set, and results show that it achieves a better performance than the ML-KNN classification without feature selection, and another classification technique called COMN, that was also proposed in the work.

The Correlated LaRank SVM method proposed in [25] is a dimensionality reduction technique incorporated into the SVM classifier with a ranking system of labels, an extension of LaRank SVM (Label Ranking SVM). The feature selection is incorporated into the classification algorithm; hence it is categorized as an embedded technique.

4.3.4. Summary of Publications on Direct Multi-label Feature Selection

Table 4 describes the publications that employ multi-label feature selection techniques that are capable of handling the task without transformation. They are correspondent to the Direct category in our proposed taxonomy. The "Feature Selection Category" column specifies which of the three categories – *Filter*, *Wrapper* or *Embedded* – the work is focused on.

The "Feature Selection Technique" column indicates which multi-label techniques were used – all of them capable of handling the data directly. If the feature selection method is embedded, it is specified in which classifier the feature selection is inserted. The "Classifier" and "Data Sets (domain)" columns are analogous to the ones in Table 4.

In this table we see that direct multi-label feature selection methods have been applied

Publication	Feature Selection	Feature Selection	Classifier	Data Sets (domain)
	Category	Technique		
[5]	Embedded	C4	.5H	Phenotypic data (gene)
[19]	Embedded	ADTB	Reuters-21450 (textual)	
[12]	Dilibedded	ADIBO	Newsgroups (textual)	
				Scene (image)
[25]	Embedded	Correlated I	Yeast (gene)	
				Yahoo webpages (web)
[38]	Embedded	PRECOMN (bas	Yeast (gene)	
				Bibtex (text)
[3:0]	Embedded	Predictive Clustering	Trees Feature Banking	Emotions (music)
[02]	Embedded	i iculture of astering	rices readure realizing	Enron (text)
				Medical (text)
[36]	Filter	FCBF extension	IBRL-ML	Various domains
[50]	1 moet	TODI extension	ECC	various domanis
[34]	Filter	gMLC	BoosTexter SVM	Graph
				Scene (image)
[83]	Wrapper	Genetic Algorithm (wrapper)	Multi-label Naive Bayes	Yeast (gene)
				Synthetic data sets
			ML-KNN	
[65]	Wrapper	HOML	BP-MLL	Yeast (gene)
11		(Hybrid optimization ML FS)	Rank-SVM	TCM CHD (medical)
[Multi-label Naive Bayes	
[53]	Filter	ReliefF-ML	ML Lazy Ranking Algorithms	Various domains
[69]	Filter	Reheff'-ML	BR-KNN	Synthetic data sets
				Enron (text)
[37]	Filter	Mutual Information ML	Multi-Label Naive Bayes	Scene (image)
				Yeast (gene)
[30]	Filter	ML-Correlation-based FS	ML-KNN	Bioinformatics gene data
[Fol	D .1.6		ML-RBF (neural network)	
[50]	Filter	ML Information Gain	Various classifiers	Various domains
[39]	Filter	ML Information Gain	ML-KNN	Various domains
1 ' '	1		Kand-SVM	1

Table 4.2: Summary of publications on direct multi-label feature selection

to data set from few domains. It would be interesting to develop work which evaluates this category of feature selection over a higher number of data sets from various domains.

4.4. Discussions and Conclusions

In this chapter, we have surveyed previous work and proposed an original categorization for the current feature selection techniques that deal with multi-label data. Up to this time, they were scattered in the literature with no common framework to describe them. With this new categorization, we expect to make it more straightforward to describe, classify, evaluate, compare and combine multi-label feature selection algorithms.

Among simple transformation methods, entropy-based transformation achieved better results for text data sets and coupled with SVM classifier in [3]. LP transformation for feature selection is more popular and achieves competitive results in various domains. However, it is generally outperformed by the BR transformation for most measures [50, 73, 68, 64], with the exception of the Subset Accuracy measure that is more sensitive on label dependency.

Our categorization showed that there is no work based on transformation methods which employ a wrapper or embedded strategy. All of them, both single and BR transformations employ a filter strategy. The analysis of how well these other strategies scale and perform on transformed multi-label data sets is therefore an open problem. Direct methods achieve better results than transformation methods in terms of performance in the case of relief and mutual information methods [70] and in terms of computational scalability [50]. For a more thorough analysis, filter techniques like in [36] should be evaluated with other multi-label classifiers. Wrapper and embedded techniques should be assessed on more domains.

Other unexplored subjects in the multi-label feature selection domain are: how well the current algorithms scale with respect to labels; how they handle class imbalance; the ability of methods to consider label correlations; an empirical comparison of representative methods from each category, in order to better visualize the pros and cons of the each one; evaluating and comparing the performance of direct multi-label feature-selection methods in the filter, wrapper and embedded categories; and evaluating and comparing methods which apply binary relevance feature selection externally with the ones which apply it internally.

Another relevant question is to determine whether a specific category is able to achieve better experimental results when combined with a specific classification method or a specific multi-label domain. So, in the next chapter, we propose a direct filter technique to handle multi-label data, adapted from the information gain metric. Then we compare it with common transformation-based techniques which employ the same information gain metric. We also employ many classification techniques and data sets from multiple domains, and evaluate the results.

Chapter 5

Information Gain Adaptation for Multilabel Data

5.1. Introduction

This chapter proposes a novel feature selection technique based on the information gain metric and capable of handling multi-label data directly. The adaptation of the information gain is based on the multi-label decision-tree classifier [5], and is constructed as a general filter feature selection technique, which could be coupled with any classifier.

The proposed adaptation is then compared with well-known transformation-based feature selection techniques. The compared techniques are coupled with various multilabel classifiers and data sets from various domains, in a comprehensive evaluation.

The techniques are also compared in terms of scalability with large data sets, evaluating which algorithms are more computationally efficient.

5.2. Information Gain Feature Selection Adaptation

This work employs the information gain measure to evaluate multi-label feature selection techniques currently used in the literature. It also adapts this measure to create a direct multi-label feature selection filter technique.

The information gain measure is based on the entropy concept. It is commonly used as a measure of feature relevance in filter strategies that evaluate features individually [80], and this method has the advantage of being fast. Let D be a data set composed by N instances of the form $(x_1, c_1), (x_2, c_2), ..., (x_N, c_N)$. In this data set, each x_i corresponds to a vector $(x_1, ..., x_d)$ that stores values for the *d* predictive features in *X* and each $c_i \in L$ corresponds to a single class label. Let *m* be the number of distinct class values $\{c_1, c_2, ..., c_m\}$, in a single-label context. The entropy of the class distribution in *D*, represented by Entropy(D), is defined by Equation 5.1.

$$Entropy(D) = -\sum_{i=1}^{m} p_i * log_2(p_i), \qquad (5.1)$$

where p_i is the probability that an arbitrary instance in D belongs to class c_i .

The concept defined in Equation 5.1 is used by the single-label feature selection strategy known as Information Gain Attribute Ranking [80] to measure the ability of a feature to discriminate between class values.

In [5], the C4.5 algorithm was adapted for handling multi-label data. This proposed decision tree algorithm allowed multiple labels at the leaves of the tree, by using an adaptation of entropy calculation. Let $(x_1, Y_1), (x_2, Y_2), ..., (x_N, Y_N), N \ge 1$, be a multi-label data set composed by N instances. Each feature X_i corresponds to a vector $(x_{i1}, ..., x_{id})$ that stores values for the d predictive features in X and each $Y_i \subset L$ corresponds to a subset of labels. The entropy of the label set distribution in D, represented by Ent.ML(D), is defined by Equation 5.2

$$Ent.ML(D) = -\sum_{i=1}^{l} p(\lambda_i) * \log_2 p(\lambda_i) + q(\lambda_i) * \log_2 q(\lambda_i),$$
(5.2)

where $p(\lambda_i)$ is the probability that an arbitrary instance in D belongs to class label λ_i , $q(\lambda_i) = 1 - p(\lambda_i)$, and l is the number of labels in the data set.

An intuition into the reason for this multi-class entropy formula is to compute the number of bits needed to describe all the labels an instance belongs to [5]. One bit per label should suffice to represent any subset of labels, but this is usually more bits than actually needed. Suppose one label Y_1 occurs in 80% of the instances. Then it is expected that one instance is more likely to belong to label Y_1 then not to belong.

We have adopted this formula to create an information gain feature selection capable of handling multi-label data. By using this as a filter approach, the feature selection can be employed with any multi-label classifier.

Let D_{ji} , $1 \leq j \leq d$ and $1 \leq i \leq N$, be the partition of D composed of all instances whose value of feature X_j is equal to x_{ji} . The entropy of the label distribution in D, restricted to the values of feature X_j , $1 \leq j \leq d_j$, represented by $Ent.ML(D, X_j)$, is defined by Equation 5.3.

$$Ent.ML(D, X_j) = \sum_{i=1}^{d_j} \left[\left(\frac{|D_{ji}|}{|D|} \right) * Ent.ML(D_{ji}) \right]$$
(5.3)

The Multi-Label Information Gain measure for each feature is computed by subtracting from the entropy of the label distribution in D the value of the entropy restricted to the values of feature X_j , $1 \le j \le d_j$. This is given by Equation 5.4.

$$MLInfoGain(D, X_i) = Ent.ML(D) - Ent.ML(D, X_i)$$
(5.4)

The proposed feature selection is a filter which work as follows: it computes the MLInfoGain values for each feature X_j in D. Next, all the scores are sorted in a ranking. In order to list the selected features as an output, it is necessary to inform the number of selected features. This can be either a percentage of the total number of features or a score threshold to split the ranking. In this work the percentage of features is used in order to compare each technique with equal conditions. This proposed feature selection is named MLInfoGain from now on.

In Appendix C these equations are revisited and used to compute the MLInfoGain scores for a multi-label data set example.

5.3. Experimental Evaluation

5.3.1. Methodology

The proposed information gain adaptation (MLInfoGain) was compared with other multilabel feature selection techniques by executing a large number of experiments. For this purpose we have elected commonly used multi-label data sets and classification algorithms. The experiments were executed using the Mulan framework [77]. Mulan is an opensource Java library for learning from multi-label data sets with a variety of state-of-the-art algorithms. We used in our experiments data sets from various domains available in the Mulan site¹. Most of the initiatives that compare multi-label learning algorithms adopt a subset of these available data sets.

In Table 5.1, we show the data sets used to evaluate multi-label classification and

¹http://mulan.sourceforge.net

			fea	atures	la	abels
name	domain	instances	nominal	numeric	distinct	$\operatorname{cardinality}$
bibtex	text	7,395	1,836	0	159	2.402
birds	audio	645	2	258	19	1.014
CAL500	music	502	0	68	174	26.044
corel5k	images	5,000	499	0	208	2.028
emotions	music	593	0	72	6	1.869
enron	text	1,702	1,001	0	53	3.378
flags-ml	images	194	9	10	7	3.392
genbase	biology	662	1,186	0	27	1.252
medical	text	978	1,449	0	45	1.245
scene	image	2,407	0	294	6	1.074
yahoo	text	$5,423{\pm}1,259$	0	$32,786{\pm}7,990$	31 ± 6	$1.481 {\pm} 0.154$
yeast	biology	2,417	0	103	14	4.237

Table 5.1: Multi-label data sets used in the experiments

feature selection algorithms.

The first two columns show the name and the domain associated to the data set. The "instances" column presents the number of instances of the data set. The "features" column shows the number of nominal and numeric features. The "distinct" subcolumn in the "labels" column shows the number of distinct labels over all instances in the data set, and the "cardinality" subcolumn represents the average number of labels of each instance, which in the multi-label setting is always superior to 1.

The feature selection techniques compared were: Binary Relevance (Transformationbased/BR/External), Copy Transformation (Transformation-based/Single), Label Powerset (Transformation-based/Single) and our proposed Multi-label Information Gain technique (Direct/Filter). The categorization in parenthesis refers to the taxonomy presented in Chapter 4. All transformation methods are coupled with the single-label information gain ranking method, in order to achieve an unbiased comparison, and so they are all categorized as a Filter feature selection.

The information gain measure requires discrete feature values. Therefore, the transformationbased techniques adopted the recursive entropy minimization heuristic [17] to discretize continuous features. This heuristic is a supervised technique, which uses the class information to select the best cut points for discretizing numeric features. It was coupled with a minimum description length criterion [61] to control the number of intervals produced over the continuous space. This procedure is commonly used in the single-label context.

The proposed direct multi-label feature selection adaptation (MLInfoGain) also requires discrete feature values. However, there is no supervised technique currently in use for discretizing multi-label data sets, to the best of our knowledge. Hence, in order to discretize features without transforming the data set, a simple unsupervised technique with 10 bins was adopted for the multi-label information gain technique.

Each feature selection technique was experimented with nine executions in which we varied the percentage of selected features between 10% and 90%, in increments of 10%. We evaluated the classifiers using 10-fold cross-validation. As an example, Table 5.2 shows the results obtained with the BR-KNN classifier, for the Hamming Loss measure and the proposed Multi-label Information Gain technique, compared with the results without feature selection (100%) as a baseline. In bold we mark the results that achieved a value equal or better than the baseline. It is possible to see that most of the feature selection options improve the predictive performance of the classification algorithm, reducing the number of features and achieving a better Hamming Loss score.

Data			Ν	/Iulti-labe	el Informa	ation Gai	n			No Sel.
\mathbf{Set}	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
bibtex	0.0132	0.0134	0.0135	0.0137	0.0138	0.0139	0.0141	0.0142	0.0143	0.0143
birds	0.0438	0.0454	0.0468	0.0459	0.0458	0.0457	0.0459	0.0461	0.0461	0.0454
CAL500	0.1435	0.1423	0.1417	0.1412	0.1420	0.1423	0.1422	0.1419	0.1422	0.1425
Corel5k	0.0094	0.0094	0.0094	0.0094	0.0094	0.0094	0.0094	0.0094	0.0094	0.0094
emotions	0.2139	0.2128	0.2081	0.2022	0.1949	0.1918	0.1929	0.1890	0.1901	0.1934
enron	0.0580	0.0596	0.0604	0.0604	0.0581	0.0576	0.0565	0.0571	0.0568	0.0580
flags-ml	0.2655	0.2540	0.2474	0.2595	0.2637	0.2630	0.2681	0.2712	0.2661	0.2749
genbase	0.0038	0.0038	0.0038	0.0038	0.0038	0.0038	0.0038	0.0038	0.0038	0.0038
medical	0.0160	0.0169	0.0175	0.0175	0.0176	0.0177	0.0182	0.0184	0.0182	0.0180
scene	0.1559	0.1351	0.1152	0.1084	0.0999	0.0957	0.0935	0.0931	0.0928	0.0920
yeast	0.2137	0.2086	0.1971	0.1963	0.1969	0.1959	0.1953	0.1942	0.1964	0.1952

Table 5.2: Results achieved with the BR-KNN classifier for the Hamming Loss measure

We have employed a large number of classification techniques, from both the transformation paradigm as well as the algorithm adaptation paradigm. Table 5.3 summarizes and categorizes the multi-label classifiers used for the evaluation. The transformation techniques used were: Label Powerset, Binary Relevance, Classifier Chains, RaKEL and HOMER, coupled with the k-NN, Decision trees (J48) and Naive Bayes single-label classifiers. The algorithm adaptations employed in this experiment were the ML-kNN and the IBLR classifier.

Mulan contains an evaluation framework that calculates a rich variety of performance measures [77]. The following multi-label measures were chosen to evaluate the results: Hamming Loss, Subset 0/1 Loss (counterpart of Subset Accuracy), Example-based Accuracy and Ranking Loss. They were chosen based on our conclusions in Chapter 3, i.e., current use in the literature and their diversity, since measures with similar equations are more likely to yield results correlated with each other. Their formulas were presented in Chapter 3. Example-based Accuracy values were inverted, so that all measures have the same pattern: the lower the value, the better.

Table 5.4 shows the overall result of each feature selection technique coupled with the

Paradigm	Classifier	Base Classifier
		k-NN
	Binary Relevance	Decision Tree
		Naive Bayes
		k-NN
	Label Powerset	Decision Tree
		Naive Bayes
Data Transformation		k-NN
	Classifier Chains	Decision Tree
		Naive Bayes
		k-NN
	RaKEL	Decision Tree
		Naive Bayes
	HOMER	k-NN
	PPT	k-NN
	ML-KNN	not applicable
Algorithm Adaptation	IBLR	not applicable
	BRKNN adaptation	not applicable

Table 5.3: Multi-label classifiers used in the experiments

BRKNN classifier, which is the adaptation derived from the BR + KNN classifier and described in Chapter 2. Each table section presents the result for a specific performance measure. The first column indicates the data set used. "BR+InfoGain", "Copy+InfoGain" and "LP+InfoGain" stand for a transformation followed by the single-label information gain measure to rank and select features. "MLInfoGain" corresponds to the multi-label information gain technique proposed in this work. "No Sel." is the result without feature selection, and also our baseline. Each cell shows the result of the multi-label measure achieved in the best case among the different percentages used in the experiment. The evaluation measures vary between 0 and 1, and the lower the value, the better. In parenthesis it is indicated the percentage of selected features that achieved the best value for each technique, and in case of ties the smaller percentage is reported. Bold values show the results that achieved a result equal or better than the baseline. Underlined values show the best result achieved in each row, for the given data set. At the end of the table the results are summarized. The "Best values (underlined)" shows the number of times that the technique achieved the best value in the experiment. The " \leq baseline score (bold)" shows the number of times that the technique achieved a value equal or better than the classification without feature selection.

With the BR-KNN classifier, the proposed multi-label information gain technique (MLInfoGain) achieved a competitive result, holding the best performance in 22 cases, out of the 44 experiments. The BR+InfoGain also achieved the best result in 22 cases. Only in 8 cases the result without feature selection achieved the best result, indicating that in most cases feature selection is helpful. In 41 cases, the proposed multi-label information gain technique was able to yield a value equal or better than the baseline (without feature selection).

HAMMING LOSS					
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.0128~(10%)	0.0132~(10%)	0.0137~(20%)	0.0132~(10%)	0.0143
birds	$\overline{0.0447}$ (30%)	0.0458~(90%)	$0.0456\ (80\%)$	0.0438~(10%)	0.0454
CAL500	0.1411 (80%)	0.1416~(40%)	0.1410~(30%)	$\overline{0.1412}$ (40%)	0.1425
Corel5k	0.0094 (10%)	0.0094~(10%)	$\overline{0.0094}$ (10%)	0.0094~(10%)	0.0094
emotions	0.1917 (90%)	0.1910(80%)	0.1951 (90%)	0.1890(80%)	0.1934
enron	0.0525(10%)	0.0579(10%)	0.0523(10%)	$\overline{0.0565}$ (70%)	0.0580
flagsml	0.2510(20%)	0.2570(20%)	0.2540(20%)	0.2474(30%)	0.2749
genbase	0.0038 (10%)	0.0038 (10%)	0.0038 (10%)	$\overline{0.0038(10\%)}$	0.0038
medical	$\frac{1}{0.0139}$ (10%)	$\frac{0.0160}{0.0160}$ (10%)	$\frac{0.0162}{0.0162}$ (10%)	$\frac{0.0160}{0.0160}$ (10%)	0.0180
scene	$\frac{0.0958}{0.0958}$ (90%)	0.0932(90%)	0.0947 (90%)	0.0928 (90%)	0.0920
veast	0.1924 (70%)	0.1971(50%)	0.1945 (90%)	0.1942 (80%)	$\frac{0.0020}{0.1952}$
SUBSET 0/1 LOSS					
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.8817 (10%)	0.9120 (10%)	0.9516 (30%)	0.9118(10%)	0.9754
birds	0.4945(50%)	0.5084(70%)	0.5069(70%)	0.4852(20%)	0.5039
CAL500	1.0000 (10%)	1.0000(10%)	1.0000(10%)	1.0000(10%)	1.0000
Corel5k	$\overline{0.9992}$ (50%)	$\overline{0.9994}$ (70%)	$\overline{0.9992}$ (90%)	$\overline{0.9994}$ (30%)	1.0000
emotions	$\frac{0.6985}{0.6985}$ (30%)	0.6883 (70%)	$\frac{0.7035}{0.7035}$ (90%)	0.6732 (80%)	0 7085
enron	0.8908(10%)	0.8837 (40%)	0.8996 (40%)	$\frac{0.0002}{0.8866}$ (40%)	0.9195
flagsml	0.8084 (20%)	$\frac{0.8450}{0.8450}$ (20%)	0.8087 (20%)	0.8034(30%)	0.8547
genbase	0.0785 (10%)	0.0785 (10%)	0.0785(10%)	$\frac{0.0001}{0.0785}$ (10%)	0.0785
medical	$\frac{0.0100(1070)}{0.4530(10\%)}$	$\frac{0.0700}{0.5471}$ (10%)	$\frac{0.0100(10\%)}{0.5471(10\%)}$	$\frac{0.0100}{0.5359}$ (10%)	0.5982
scono	$\frac{0.4000(1070)}{0.4130(00\%)}$	0.0411 (1070)	0.0411 (1070)	0.0005 (10%)	0.0002
Nonst	0.4130(9070)	0.4088 (9078)	0.4033(90%) 0.8056(90\%)	$\frac{0.4003}{0.7064}$ (80%)	0.4038
EXAMPLE-BASED A	$\frac{0.1965(3070)}{CCUBACY(IN)}$	VERTED)	0.0000 (0070)	0.7304 (8070)	0.0010
Data Set	BB+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.7894 (10%)	0.8369 (10%)	0.8848(30%)	0.8369 (10%)	0.9289
birds	$\frac{0.1001}{0.4443}$ (30%)	0.4560 (90%)	0.4535(80%)	0.4282(10%)	0 4482
CAL500	0.8094 (80%)	0.8107(70%)	0.8099 (60%)	$\frac{0.8106}{0.8106}$ (40%)	0.8144
Corel5k	$\frac{0.00011}{0.9915}$ (80%)	0.9928 (70%)	0.9941 (80%)	0.9925 (70%)	0.9975
emotions	$\frac{0.0010}{0.4702}$ (70%)	0.4686 (80%)	0.4871(50%)	0.4643 (80%)	0 4851
enron	0.4102(10%)	0.7314 (20%)	0.7000 (10%)	$\frac{0.1010}{0.7162}$ (70%)	0.7973
flagsml	$\frac{0.0000(1070)}{0.3953(20\%)}$	0.3045(20%)	0.1000 (10%) 0.3903 (20%)	0.3824 (30%)	0.1313
gonbaso	0.0000(2070)	0.0463 (10%)	0.0300 (2070)	$\frac{0.0324}{0.0463}$ (10%)	0.4504
modical	$\frac{0.0403 (1070)}{0.2815 (10\%)}$	$\frac{0.0403(1070)}{0.4700(10\%)}$	$\frac{0.0403(1070)}{0.4838(1070)}$	$\frac{0.0403(1070)}{0.4718(10\%)}$	0.0403
	$\frac{0.3813(1070)}{0.3881(0007)}$	0.4799 (1070)	0.4828 (1070)	0.4710(1070)	0.0407
scene	0.3001(9070)	0.3031(9070)	0.3037 (9070)	$\frac{0.3730}{0.4065}$ (80%)	0.3602
PANKING LOSS	0.4975 (90%)	0.3037 (9070)	0.3002(90%)	0.4905 (80%)	0.4998
Data Set	BB +InfoCain	Conv+InfoGain	LP+InfoCain	MLInfoCain	No Sel
bibtev	0.1342 (10%)	$\frac{0.1807}{10\%}$	$\frac{11}{-0.2296}$ (30%)	0.1805(10%)	0.2830
birds	$\frac{0.1042}{0.0861}$ (1070)		0.0878 (40%)	0.0872 (60%)	0.2000
	$\frac{0.0001}{0.0201}$ (70%)	0.0003 (3070)	0.0078 (4070)	0.0012 (0070)	0.0004
Corol5k	$0.2301 (10%) \\ 0.1887 (10%)$	0.2301 (3070) 0.1007 (1007)	$\frac{0.2290}{0.9954} (4070)$	0.2010 (80%)	0.2010
coreisk	$\frac{0.1607}{0.1604}$	0.1997 (1070)	0.2234 (1070)	0.1903 (1070)	0.3243
eniotions	0.1024(7070)	0.1023 (3070)	0.1060 (1007)	$\frac{0.1004}{0.1007}$	0.1010
	0.1100 (10%)	0.1090 (10%)	0.1200 (10%)	$\frac{0.1087 (10\%)}{0.1001 (10\%)}$	0.1055
nagsmi	$\frac{0.1815}{0.0059} (50\%)$	0.1855 (20%)	0.1810 (50%)	0.1891 (40%)	0.1978
genbase	$\frac{0.0052}{0.0052}$ (10%)	$\frac{0.0052}{0.0490}$ (10%)	$\frac{0.0052}{0.0445}$ (10%)	$\frac{0.0052}{0.0497}$ (10%)	<u>0.0052</u>
medical	$\frac{0.0350(10\%)}{0.0005(10\%)}$	0.0438 (10%)	0.0445 (10%)	0.0437 (10%)	0.0475
scene		0.0902 (90%)	0.0927 (90%)	0.0905 (90%)	$\frac{0.0889}{0.1550}$
yeast	0.1757 (90%)	0.1766 (90%)	0.1797 (90%)	0.1755 (80%)	0.1778
Best values (underlined)	22	7	10	22	8
\leq baseline score (bold)	39	33	31	41	

Table 5.4: Best results achieved with the BRKNN classifier

It is worth noting the behaviour for some data sets: the genbase data set is not affected by feature selection, which indicates that it can be drastically reduced without compromising its performance; on the other hand, the scene data set achieves a better performance with most of its features, indicating that it is less suitable for feature selection.

Table 5.5 corresponds to a summarized result of the other classifiers performance when coupled with feature selection, similar to the last row of the previous table. It shows the number of times that each feature selection achieved a result better than (\leq) the baseline score, considering the evaluated data sets and the four performance measures adopted in this work. For instance, the third row in the table refers to the "BRKNN" classifier, which corresponds to the summarized last row of Table 5.4. The full tables for all classifiers are presented in the Appendix B.

The results indicate that most of the time the feature selection was beneficial for the overall classification. For instance, when using the RAKEL + K-NN classifier, the BR + InfoGain feature selection achieved a performance equivalent or better than the result without feature selection 37 times out of 44 results (i.e., 4 measures * 11 data sets). For the Copy + InfoGain feature selection this result was achieved 32 times; for the LP + InfoGain 31 times; and for the proposed MLInfoGain this result occurred 39 times.

Classifier	BR+InfoGain	$\mathbf{Copy} + \mathbf{InfoGain}$	LP+InfoGain	MLInfoGain
BR + DecisionTree	42	40	42	42
BR + NaiveBayes	37	30	31	37
BRKNN	39	33	31	41
CC + DecisionTree	43	38	40	40
CC + K-NN	38	37	35	43
CC + NaiveBayes	32	29	30	36
HOMER + K-NN	37	36	38	39
IBLR ML	38	34	31	37
LP + DecisionTree	42	38	38	41
LP + K-NN	39	39	36	43
LP + NaiveBayes	32	27	30	30
ML-KNN	39	28	27	37
PPT + K-NN	30	25	25	26
RAKEL + K-NN	37	32	31	39
RAKEL + DecisionTree	37	32	33	35
RAKEL + NaiveBayes	37	28	29	34

Table 5.5: Number of times that each feature selection achieved a result better than (\leq) the baseline score

5.3.2. Statistical Evaluation

The same procedure described in [46] was followed. The Friedman test was employed in order to evaluate if the differences in performance of the multi-label feature selection techniques are statistically significant. A non-parametric test makes no assumption about the data distribution, unlike, for instance, a paired t-test which assumes data normality. The feature selection techniques were ranked according to their performance for each classification algorithm and data set. The best performing technique was ranked first, the second best was ranked second, and so on. In case of ties, the ranks were averaged. From the average ranks of the techniques, the Friedman statistic was calculated, and then at a significance level of 5%, the hypothesis that techniques performed equally in mean ranking was rejected.

Then a post-hoc Nemenyi test was used to compare each feature selection techniques to each other. The performance of two techniques is considered significantly different if their average ranks differ by more than a critical distance value. Figure 5.1 shows the results from the Nemenyi post-hoc test for the four different measures used in the experiments for the BRKNN classifier. Each diagram presents an enumerated axis with the average ranks of each technique. The best ranked are at the right-most side of the diagram. The lines for the average ranks of the algorithms that do not differ significantly (at the significance level of 0.05) are connected with a line.



Figure 5.1: Critical diagram for each measure in the BRKNN classifier from the Nemenyi post-hoc test at 0.05 significance

The diagrams show that for most measures the MLInfoGain feature selection technique is significantly better than the Copy+InfoGain and LP+InfoGain techniques. However, when comparing MLInfoGain and BR+InfoGain techniques, the diagrams reveal no significant difference.

5.3.3. Experiments on Large Multi-label Data Sets

Most multi-label classification methods either do not scale or have unsatisfactory performance [72]. So, feature selection becomes an important task for large data sets. In order to evaluate which feature selection techniques scales better, we have also conducted experiments on larger multi-label data sets.

The number of features is the criteria we used to define a large data set, given that we are assessing feature selection techniques. Among the data sets used before, the one with the most number of features was the bibtex data set, with 1,836 features.

So, we have chosen 11 independently compiled data sets from the Yahoo! directory [72], each one with more than 5,000 instances and 30,000 features, being suitable for the scalability experiments.

For these experiments, the BRKNN classifier was employed. It is implemented in Mulan using a single search for k nearest neighbors but at the same time making independent predictions for each label [67]. This adaptation was described in Chapter 2. This BRKNN adaptation runs much faster than the transformation-based BR technique followed by the k-NN algorithm. This makes BRKNN the most scalable classification algorithm used in this work, and the reason to employ this algorithm for the experiments with larger data sets.

Table 5.6 shows the result of the experiment with larger data sets executed in a similar fashion as the previous one. We used BR+InfoGain and the proposed MLInfoGain techniques with the 10% parameter of selected features. Other percentages were evaluated, but there is no significant difference in the results, except for the larger classification time. Each row shows the result on a Yahoo data set. Columns "HLoss", "SLoss", "EbAcc" and "RLoss" show the result of the Hamming Loss, Subset 0/1 Loss, Example-based Accuracy (inverted) and Ranking Loss, respectively. Column "Time(s)" shows the total execution CPU time of the experiment (feature selection time + classification time), in seconds. The computer used in the experiments was an AMD FX 8210 8-Core 3.1 Ghz with 8 Gb of RAM and a 64 bit OS.

The same non-parametric statistical test used before shows no significant difference between both techniques for the performance measures. However, the computational time of MLInfoGain is much smaller than the BR+InfoGain. The BR approach takes roughly 100 times more to execute the same experiment when compared with the direct MLInfoGain approach. Higher computational time also occurs on experiments with the

Data Sat		\mathbf{BR}	-InfoGai	n 10%			\mathbf{ML}	InfoGain	10%	
Data Set	HLoss	SLoss	EbAcc	RLoss	Time(s)	HLoss	SLoss	EbAcc	RLoss	Time(s)
Arts	0.0595	0.8991	0.8770	0.1941	$53,\!692$	0.0617	0.9280	0.9128	0.2093	686
Business	0.0267	0.4464	0.3000	0.0745	$93,\!634$	0.0270	0.4497	0.3026	0.0767	1,015
Computers	0.0360	0.6497	0.5900	0.1509	186,670	0.0368	0.6439	0.5812	0.1604	1,869
Education	0.0413	0.8771	0.8578	0.1658	142,035	0.0427	0.9192	0.9035	0.1854	1,487
Entertainment	0.0578	0.7621	0.7390	0.1778	125,560	0.0578	0.8113	0.7944	0.1933	1,726
Health	0.0430	0.6890	0.6141	0.1292	110,008	0.0456	0.7299	0.6295	0.1342	1,174
Recreation	0.0559	0.8262	0.8117	0.1990	122,812	0.0584	0.8757	0.8624	0.2328	$1,\!647$
Reference	0.0317	0.6342	0.6002	0.2009	133,902	0.0326	0.6839	0.6546	0.2107	1,344
Science	0.0343	0.9054	0.8940	0.2100	120,105	0.0350	0.9456	0.9379	0.2264	1,069
Social	0.0254	0.6204	0.5937	0.1277	334,846	0.0276	0.6849	0.6579	0.1341	3,080
Society	0.0537	0.7762	0.7207	0.1898	215,605	0.0547	0.8075	0.7620	0.1998	2,442

Table 5.6: Result of experiments on large data sets with BRKNN classifier

Copy and LP transformations, and the results are not reported in this work due to their low performance.

It is worth noting that running this experiment with large data sets and without feature selection is faster than using the BR approach, because the transformation process takes a large amount of time. This is the main advantage of the MLInfoGain approach, which is faster than any feature selection based on transformation and it also accelerates the classification by reducing the number of features.

5.4. Chapter Summary

In this chapter, an adaptation of the information gain feature selection technique to handle multi-label data directly was proposed. This filter technique, named MLInfoGain, was experimentally compared with various multi-label feature selection methods based on transformation. All compared techniques were coupled with different classification techniques and data sets from different domains.

Experimental results indicated that the proposed multi-label information gain feature selection strategy (MLInfoGain) achieved a competitive performance against the other techniques and outperformed the baseline on most cases. For larger data sets, the proposed technique scaled much better than the other feature selection methods, significantly outperforming the transformation-based techniques in terms of computational efficiency.

In the next chapter, we advance our research on direct multi-label feature selection techniques, by proposing a further adaptation based on the lazy paradigm.

Chapter 6

Lazy Multi-label Feature Selection

6.1. Introduction

This chapter presents one of the main contributions of this thesis: a new method for multi-label feature selection based on the lazy paradigm. This method has two main characteristics: (a) the use of the information gain measure that was adapted for multi-label feature selection in Chapter 5 and in [50]; and (b) a multi-label adaptation of the single-label lazy strategy proposed in [52].

The goal of this novel technique is to benefit the multi-label classification and be more scalable than the current techniques used in the literature. The lazy adaptation is compared experimentally with other multi-label feature selection techniques and the results are reported.

6.2. Lazy Feature Selection

In conventional feature selection strategies, features are selected in a preprocessing phase. The features which are not selected are discarded from the data set and no longer participate in the classification process.

In [52], a lazy feature selection strategy was proposed based on the hypothesis that postponing the selection of features to the moment at which an instance is submitted for classification can contribute to identifying the best features for the correct classification of that particular instance. For each different instance to be classified, it is possible to select a distinct and more appropriate subset of features to classify it.

Below we give a single-label example from [52] to illustrate the fact that the classi-

fication of certain instances could take advantage of features discarded by conventional feature selection strategies.

In Table 6.1, the same data set, composed of two features -X, Y – and the class C, is represented twice. The left occurrence is ordered by the values of X and the right one is ordered by the values of Y. It can be observed in the left occurrence that the values of X are strongly correlated with the class values making it a useful feature. Only value 4 is not indicative of a unique class value.

Furthermore, as shown in the right occurrence, feature Y would be a strong candidate to be eliminated since its values do not properly discriminate between the classes. However, there is a strong correlation between the value 4 of feature Y and the class value B, which would be lost if this feature were discarded. The classification of an element with value 4 in the Y feature would clearly take advantage of the presence of this feature.

Data S	Set Sorte	d by X	Data Set Sorted by Y		
– X –	– Y –	– C –	– X –	– Y –	– C –
1	2	В	2	1	A
1	3	В	3	1	В
1	4	В	4	1	A
2	1	A	1	2	В
2	2	A	2	2	A
2	3	A	3	2	В
3	1	В	1	3	В
3	2	В	2	3	A
3	4	В	4	3	В
4	1	A	1	4	В
4	3	В	3	4	В
4	4	B	4	4	В

Table 6.1: Single-label Data Set Example

A conventional feature selection strategy (an "eager" selection strategy) – is likely to select feature X in detriment of Y, regardless of the instances that are submitted for classification. Hence, the main motivation behind the proposed lazy feature selection is the ability to assess the feature values of the instance to be classified, and use this information to select features that discriminate the classes well for those particular values.

6.3. Multi-label Adaptation

In the previous section, a single-label data set that motivates the lazy feature selection was presented, indicating a situation where it could be worthwhile to avoid discarding features from the data set.

Table 6.2 presents an analogous multi-label example. Again, the data set, composed

of two features -X, Y - and their labels, is represented twice. The left occurrence is ordered by the values of X and the right one is ordered by the values of Y.

Dat	Data Set Sorted by X			a Set Sor	ted by Y
- X -	– Y –	– Labels –	– X –	– Y –	– Labels –
1	1	A	1	1	A
1	2	В	2	1	A
1	3	В	3	1	B,C
1	4	A,B	4	1	В
2	1	A	1	2	В
2	2	A	2	2	A
2	3	A	3	2	B,C
2	4	A,B	4	2	A,C
3	1	B,C	1	3	В
3	2	B,C	2	3	A
3	3	B,C	3	3	B,C
3	4	A,B	4	3	A
4	1	В	1	4	A,B
4	2	A,C	2	4	A,B
4	3	A	3	4	A,B
4	4	A,B	4	4	A,B

Table 6.2: Multi-label Data Set Example

It can be observed in the left occurrence that the values of X are strongly correlated with at least one label value. The instances with X = 1 do not have the label C among their labels. When X = 2, the label A is present, and the label C is not. For the instances with X = 3, the label B is always present. The value X = 4 is the only one that is not strongly correlated with any labels. Nonetheless, this makes X a useful feature, because most of its values are correlated to at least one label (or its absence).

On the other hand, as shown in the right occurrence, the values Y = 1, Y = 2 and Y = 3 do not have a strong correlation with any label. Y would be a strong candidate to be eliminated, since most of its values do not properly discriminate a label. However, there is a very strong correlation between the value 4 of feature Y and the three labels values: A, B and the absence of C (or $\neg C$). This correlation would be lost if this feature were discarded. The multi-label classification of an element with value 4 in the Y feature would clearly take advantage of the presence of this feature.

So we present in this small multi-label example a motivation for postponing the selection of features to the moment at which an instance is submitted for classification. This way the selection can take a more informed decision on which features to keep on the data set.

Lazy feature selection is a general strategy, as it can employ different evaluation measures to evaluate the quality of the features. This work proposes to instantiate the strategy analogous to the original work [52], using an entropy-based criterion to rank features [80]. This entropy measure was extended to the multi-label setting in [5], where the C4.5 algorithm was adapted for handling multi-label data. This decision tree algorithm allowed multiple labels at the leaves of the tree, by using an adaptation of entropy calculation, described in the previous chapter and revisited in Equation 5.2.

$$Ent.ML(D) = -\sum_{i=1}^{l} p(\lambda_i) * \log_2 p(\lambda_i) + q(\lambda_i) * \log_2 q(\lambda_i), \qquad (5.2 \text{ revisited})$$

where $p(\lambda_i)$ is the probability that an arbitrary instance in D belongs to class label λ_i , $q(\lambda_i) = 1 - p(\lambda_i)$, and l is the number of labels in the data set.

Chapter 5 also presented the formula for computing the entropy of the label distribution in D, restricted to the values of feature X_j , $1 \leq j \leq d_j$, represented by $Ent.ML(D, X_j)$ and defined by Equation 5.3.

$$Ent.ML(D, X_j) = \sum_{i=1}^{d_j} [(\frac{|D_{ji}|}{|D|}) * Ent.ML(D_{ji})],$$
(5.3 revisited)

where D_{ji} , $1 \leq i \leq d_j$, is the partition of D composed of all instances whose value of feature X_j is equal to x_{ji} .

These equations were used in the MLInfoGain technique in Chapter 5, as an adaptation of the Information Gain Ranking [54] for the multi-label context. This technique is considered as "eager", which is the opposite of "lazy". It is able to select the features as a data preprocessing step.

In order to adapt this concept to the lazy paradigm, each individual feature value needs to be measured separately from the others. The entropy of the label distribution in D, restricted to the value x_{ji} , $1 \le i \le d_j$ and to the label l_k , $1 \le k \le q$, of feature X_j , $1 \le j \le d$, represented by $Ent.ML(D, X_j, x_{ji}, l_k)$ is defined by Equation 6.1.

$$Ent.ML(D, X_j, x_{ji}, l_k) = Ent.ML(D_{jik}).$$
(6.1)

For each label l_k this equation gives a different entropy value. The closer the entropy $Ent.ML(D, X_j, x_{ji}, l_k)$ is to zero, the greater the chance that the value x_{ji} of feature X_j is a good discriminator for label l_k . Equation 6.2 aggregates the result for all q labels in D using the *min* function, in order to identify feature values which best discriminate at least one label.

$$LazyEnt.ML(D, X_j, x_{ji}) = min_{k=1}^{k=q}Ent(D_{jik}).$$
(6.2)

For computing the lazy multi-label information gain, for each feature X_j , if the discrimination ability of the specific value x_{ji} of X_j (*Ent.ML*(D, X_j, x_{ji})) is better than (less than) the overall discrimination ability of feature X_j (*Ent.ML*(D, X_j)) then the former will be considered for ranking X_j . This is given by Equation 6.3.

$$LazyML.IG(D, X_j, x_{ji}) = Ent.ML(D) - min[Ent.ML(D, X_j), LazyEnt.ML(D, X_j, x_{ji})]$$
(6.3)

The choice of considering the minimum value from both the entropy of the specific value and the overall entropy of the feature was motivated in [52] by the fact that some instances may not have any relevant features considering their particular values. In this case, features with the best overall discrimination ability will be selected. In other words, if the values of an instance do not help the feature selection (lazy), then select the best features in the data set regardless, considering the multi-label information gain measure (eager).

As it was done with the equations from Chapter 5, in Appendix C the equations from this chapter are revisited and used to compute the Lazy MLInfoGain scores for a multi-label data set example.

The proposed lazy adaptation of the multi-label information gain works as follows: for each instance I to be classified, the value $LazyML.IG(D, X_j, x_{ji})$ for each feature X_j and value x_{ji} is computed, where i is the index of the feature value for this specific instance I. The scores are sorted in a ranking. The filter strategy implemented in this work selects a percentage r of the best features. After the feature selection phase, the multi-label classification will only use the best features according to the percentage r to classify instance I. For the next instance, this process is repeated. This feature selection technique is categorized as Direct/Filter according to our taxonomy.

Any feature selection lazy adaptation should be coupled with a lazy multi-label classifier, because the 'lazy module' is called at classification time for every new instance. This restriction is motivated by the target classifier not requiring to construct a model as a preprocessing step. For instance, a decision tree classifier would not benefit from the lazy adaptation, because the tree model would need to be reconstructed for every new instance. On the other hand, the k-NN classifier, being a lazy classifier, does not construct a model in the preprocessing phase. So after receiving a new instance to be classified, it could call the 'lazy module', select the suitable features for that instance, and then compute the neighbors distances and proceed with the classification.

6.4. Experiments with BRKNN Classifier

6.4.1. Methodology

The multi-label lazy feature selection (Lazy MLInfoGain) was implemented in Mulan [77], the same open-source framework used in the previous experiments of this work.

The lazy feature selection was incorpored into the BRKNN classifier. The BRKNN classifier was implemented using a single search for k nearest neighbors [67], the same used in Chapter 5 and described in Chapter 2. The lazy feature selection was executed within the algorithm just before the actual classification takes place. The classifier considered only the r features selected in a lazy manner to compute its neighbors distances, for each test instance. This implies that for different instances distinct subsets of features were used. The experiments were executed with the default parameter settings in the Mulan tool, using the original BRKNN classifier.

Table 6.3 shows the overall result of each feature selection technique coupled with the BRKNN classifier. Each table section presents the result for a specific performance measure. The first column indicates the data set used, and the other columns indicates which feature selection technique was applied before the classification. "BR+InfoGain", "Copy+InfoGain" and "LP+InfoGain" stand for a transformation followed by the singlelabel information gain measure to rank and select features. "MLInfoGain" corresponds to the multi-label information gain technique proposed in Chapter 5. "Lazy MLInfoGain" is the lazy adaptation proposed in this chapter.

"No Sel." is the result without feature selection, and also the baseline. Each cell shows the result of the multi-label measure achieved in each case, varying between 0 and 1, and the lower the value, the better. In parenthesis we show the percentage of selected features that achieved the best value for each technique, and in case of ties we report the smaller percentage. Bold values show the results that achieved a score equal or better than the baseline, and underlined values show the best result achieved in each row. At the end of the table we summarize the results.

With the BRKNN classifier, the proposed lazy multi-label information gain technique

HAMMING LO	SS	<u>a</u>	IDLL COL			
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLIntoGain	Lazy MLIntoGain	No Sel.
DIDIEX	$\frac{0.0128(10\%)}{0.0447(80\%)}$	0.0132(10%)	0.0137 (20%)	0.0132(10%)	$\frac{0.0128}{0.0445}$ (10%)	0.0143
birds	0.0447 (30%)	0.0458 (90%)	0.0456(80%)	$\frac{0.0438(10\%)}{0.1418(40\%)}$	0.0445(30%)	0.0454
CALSUU	0.1411(80%)	0.1416(40%)	$\frac{0.1410(30\%)}{0.000}$	0.1412(40%)	0.1415(60%)	0,1425
Corel5k	$\frac{0.0094}{0.0094}$	$\frac{0.0094}{0.0094}$	$\frac{0.0094}{0.0094}$	$\frac{0.0094}{0.0094}$	$\frac{0.0094}{0.0094}$ (10%)	0.0094
emotions	0.1917 (90%)	0.1910 (80%)	0.1951 (90%)	0.1890(80%)	0.1912 (70%)	0.1934
enron	0.0525(10%)	0.0579(10%)	0.0523(10%)	0.0565 (70%)	0.0508(30%)	0.0580
flagsml	0.2510(20%)	0.2570(20%)	0.2540(20%)	0.2474(30%)	0.2521 (30%)	0.2749
genbase	0.0038(10%)	0.0038(10%)	0.0038(10%)	0.0038(10%)	0.0038(10%)	<u>0.0038</u>
medical	0.0139(10%)	0.0160 (10%)	0.0162(10%)	0.0160(10%)	0.0163(10%)	0.0180
scene	0.0958(90%)	0.0932 (90%)	0.0947 (90%)	0.0928 (90%)	0.0918 (60%)	0.0920
yeast	0.1924 (70%)	0.1971~(50%)	0.1945 (90%)	0.1942(80%)	0.1949 (40%)	0.1952
SUBSET 0/1 LC	DSS					
Data Set	BR+In foG ain	Copy+InfoGain	LP+InfoGain	MLInfoGain	Lazy MLIn foGain	No Sel.
bibtex	0.8817 (10%)	0.9120(10%)	0.9516 (30%)	0.9118 (10%)	0.8772 (10%)	0.9754
birds	0.4945 (50%)	0.5084~(70%)	0.5069(70%)	0.4852 (20%)	0.4914 ($30%$)	0.5039
CAL500	1.0000 (10%)	1.0000 (10%)	1.0000 (10%)	1.0000 (10%)	1.0000 (10%)	1.0000
Corel5k	0.9992 (50%)	0.9994(70%)	0.9992 (90%)	0.9994(30%)	0.9976 (10%)	1.0000
emotions	0.6985 (30%)	0.6883 (70%)	0.7035 (90%)	0.6732 ($80%$)	0.6968 (90%)	0.7085
enron	0.8908(10%)	0.8837 (40%)	0.8996~(40%)	0.8866 (40%)	0.8720 $(30%)$	0.9195
flagsml	0.8084(20%)	0.8450(20%)	0.8087 (20%)	0.8034(30%)	0.8192(20%)	0.8547
genbase	0.0785 (10%)	$0.0785\ (10\%)$	0.0785(10%)	0.0785(10%)	0.0785 (10%)	<u>0.0785</u>
medical	0.4530 (10%)	0.5471(10%)	0.5471(10%)	0.5359(10%)	0.5440 (10%)	0.5982
scene	0.4130 (90%)	0.4088(90%)	0.4088(90%)	0.4005(80%)	0.3930 (70%)	0.4038
yeast	0.7985 (90%)	0.8014(90%)	0.8056 (90%)	0.7964(80%)	0.8014 (90%)	0.8018
EXAMPLE-BAS	SED ACCURACY	(Inverted)				
Data Set	BR+In foGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	Lazy MLIn foGain	No Sel.
bibtex	0.7894 (10%)	0.8369 (10%)	0.8848(30%)	0.8369 (10%)	0.7887 (10%)	0.9289
birds	0.4443(30%)	0.4560(90%)	0.4535 $(80%)$	0.4282(10%)	0.4349(30%)	0.4482
CAL500	0.8094 (80%)	0.8107 (70%)	0.8099(60%)	0.8106(40%)	0.8120 (70%)	0.8144
Corel5k	0.9915(80%)	0.9928 (70%)	0.9941 (80%)	0.9925(70%)	0.9876 (20%)	0,9975
emotions	0.4702 (70%)	0.4686(80%)	0.4871(50%)	0.4643(80%)	0.4754(60%)	0.4851
enron	0.6530 (10%)	0.7314(20%)	0.7000 (10%)	0.7162(70%)	0.6202 (20%)	0.7973
flagsml	0.3953 (20%)	0.3945(20%)	0.3903 (20%)	0.3824 (30%)	$\overline{0.3955}$ (20%)	0,4364
genbase	0.0463 (10%)	0.0463 (10%)	0.0463 (10%)	0.0463(10%)	0.0463 (10%)	0.0463
medical	$\frac{1}{0.3815(10\%)}$	$\frac{1}{0.4799(10\%)}$	$\frac{0.4828}{0.4828}$ (10%)	0.4718(10%)	$\frac{1}{0.4867}$ (10%)	0.5437
scene	$\frac{0.3881}{0.3881}$	0.3831 (90%)	0.3837 (90%)	0.3750(80%)	0.3669 (70%)	0.3802
veast	0.4975(90%)	0.5037 (90%)	0.5002(90%)	0.4965(80%)	0.5004 (90%)	0.4998
BANKING LOS	S (S S	(0070)	(0070)		(0070)	
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	Lazy MLIn foGain	No Sel-
hibtex	0.1342 (10%)	0.1807 (10%)	0.2296 (30%)	0.1805(10%)	0.1412 (10%)	0.2830
birds	$\frac{0.0861}{0.0861}$ (70%)	0.0889 (90%)	0.0878 (40%)	0.0872 (60%)	0.0868(70%)	0.0864
CAL500	$\frac{0.0001}{0.2301}$ (70%)	0.2301 (30%)	0.2295(40%)	0.2310 (90%)	0.2285 (70%)	0.2310
Corel5k	0.1887 (10%)	0.1997 (10%)	0.2254(10%)	0.1983(10%)	$\frac{0.2025}{0.2025}$ (10%)	0.3243
emotions	$\frac{0.1624}{0.1624}$ (70%)	0.1623 (80%)	0.1599 (90%)	0.1584 (60%)	0.1574(50%)	0 1610
enron	0 1165 (10%)	0 1096 (10%)	0 1 260 (10%)	0.1087 (10%)	$\frac{0.1039}{0.1039}$	0.1655
flageml	0 1815 (50%)	0 1855 (20%)	0 1816 (50%)	0 1801 (40%)	$\frac{0.1832}{0.1832}$ (50%)	0 1978
ganbasa	$\frac{0.1010}{0.0052}$ (10%)	0.1000 (2070)	0 0059 (10%)	0 0059 (40%)	0.0052 (00%)	0.1970
Benuase	$\frac{3.0032}{0.0350}$ (10%)	$\frac{0.0002}{0.0438}$ (10%)	$\frac{0.0002}{0.0445}$ (10%)	$\frac{0.0002}{0.0427}$ (10%)	$\frac{0.0032}{0.0481}$ (10%)	0.0002
medical	$\frac{0.0350(10\%)}{0.0025(0.0\%)}$	0.0438 (10%)	0.0027 (0.0%)	0.0437 (10%)	0.0431 (1070)	0.0475
scene	0.0925 (90%)	0.0902 (90%)	0.0927 (90%)	0.0908 (90%)	$\frac{0.0851}{0.1803}$ (60%)	0.0889
Peast Peast	0.1757 (90%)	0.1700 (90%)	0.1797 (90%)	0.1755 (80%)	0.1803 (60%)	0,1778
Dest values	17	6	7	18	21	6
(unuernnea)						<u> </u>
_ Dasenne score	39	33	31	41	41	

Table 6.3: Best results achieved with the BRKNN classifier, comparing feature selection techniques with the proposed LazyMLInfoGain technique

(Lazy MLInfoGain) achieved a competitive result, holding the best performance in 21 cases, out of the 44 experiments. The non-lazy MLInfoGain technique achieved the best result in 18 cases, and the BR+InfoGain transformation technique achieved the best result in 17 cases. Only in 6 cases the result without feature selection achieved the best result. In 41 cases, both the proposed multi-label information gain technique and the lazy adaptation were able to yield a value equal or better than the baseline (without feature selection).

These preliminary results indicate an improvement over the non-lazy MLInfoGain technique proposed in the previous chapter. To confirm this, in the next section a statistical analysis is used to evaluate these results.

6.4.2. Statistical Evaluation

The same statistical analysis conducted in Chapter 5 was used to evaluate if the differences in performance of the multi-label feature selection techniques are statistically significant.

The five feature selection techniques were ranked according to their performance for each data set and percentage of selected features. The best performing technique was ranked first, the second best was ranked second, and so on. In case of ties, the ranks were averaged. From the average ranks of the techniques, the Friedman statistic was calculated, and then at a significance level of 5%, the hypothesis that the techniques performed equally well in average ranking was rejected.

Then a post-hoc Nemenyi test was used to compare the feature selection techniques to each other. The performance of two techniques is considered significantly different if their average ranks differ by more than a critical distance value. Figure 6.1 shows the results from the Nemenyi post-hoc test for the four different measures used in the experiments for the BRKNN classifier. Each diagram presents an enumerated axis with the average ranks of each technique. The best rankings are at the right-most side of the diagram. The lines for the average ranks of the algorithms that do not differ significantly (at the significance level of 0.05) are connected with a line.

The diagrams show that the proposed Lazy MLInfoGain generally outperforms the original MLInfoGain (eager) with a significant difference, except for the Example-based Accuracy. It also outperforms the Copy+InfoGain and LP+InfoGain techniques for all measures. The second best feature selection algorithm is the BR+InfoGain, which is not significantly worse than Lazy MLInfoGain, and not significantly better than the original


Figure 6.1: Critical diagram for each measure with the BRKNN classifier from the Nemenyi post-hoc test at 0.05 significance

MLInfoGain. The other results are similar to the ones obtained in the previous chapter.

6.4.3. Experiments on Large Multi-label Data Sets for BRKNN

This section reports the experiments on larger multi-label data sets, analogous to the ones reported in Chapter 5. The same 11 independently compiled data sets from the Yahoo! directory [72] were chosen, each one with more than 5,000 instances and 30,000 features.

Table 6.4 shows the result of the experiment with larger data sets executed in a similar fashion as the previous one. The feature selection techniques compared are the MLInfoGain proposed in Chapter 5 and the proposed Lazy MLInfoGain proposed in this chapter. Both techniques selects 10% of features from the data sets. Each row shows the result on a Yahoo data set. Columns "HLoss", "SLoss", "EbAcc" and "RLoss" show the result of the Hamming Loss, Subset 0/1 Loss, Example-based Accuracy (inverted) and Ranking Loss, respectively. Column "Time(s)" shows the total execution time of the experiment (feature selection time + classification time), in seconds. The computer used in the experiments was an AMD FX 8210 8-Core 3.1 Ghz with 8 Gb of RAM and a 64 bit OS.

Data Sat		MLInfoGain 10%					Lazy MLInfoGain 10%			
Data Set	HLoss	SLoss	EbAcc	RLoss	Time(s)	HLoss	SLoss	EbAcc	RLoss	Time(s)
Arts	0.0617	0.9280	0.9128	0.2093	686	0.0615	0.9213	0.9071	0.2027	1,111
Business	0.0270	0.4497	0.3026	0.0767	1,015	0.0273	0.4505	0.3062	0.0773	1,590
Computers	0.0368	0.6439	0.5812	0.1604	1,869	0.0362	0.6515	0.5887	0.1518	2,832
Education	0.0427	0.9192	0.9035	0.1854	1,487	0.0429	0.9109	0.8954	0.1693	2,188
Entertainment	0.0578	0.8113	0.7944	0.1933	1,726	0.0566	0.7993	0.7811	0.1708	2,750
Health	0.0456	0.7299	0.6295	0.1342	1,174	0.0442	0.6927	0.6147	0.1367	1,738
Recreation	0.0584	0.8757	0.8624	0.2328	$1,\!647$	0.0586	0.8828	0.8699	0.2128	2,584
Reference	0.0326	0.6839	0.6546	0.2107	1,344	0.0315	0.6523	0.6217	0.1818	2,141
Science	0.0350	0.9456	0.9379	0.2264	1,069	0.0346	0.9316	0.9240	0.2064	1,762
Social	0.0276	0.6849	0.6579	0.1341	3,080	0.0266	0.6500	0.6239	0.1249	4,795
Society	0.0547	0.8075	0.7620	0.1998	2,442	0.0549	0.7803	0.7264	0.1967	$3,\!694$

Table 6.4: Result of experiments on large data sets with BRKNN classifier, comparing MLInfoGain with Lazy MLInfoGain feature selection

For these Yahoo! directory data sets, the Lazy MLInfoGain generally outperformed the original non-lazy MLInfoGain technique, specially for the Example-Based Accuracy and Ranking Loss measures. In terms of computational time, the Lazy technique was slower due to the overhead associated with the postponing of feature selection to the classification time. This difference is not significant when compared with transformationbased techniques, which takes more time for the classification. For instance, Table 6.5 compares the performance of BR + InfoGain and the Lazy MLInfoGain.

				1001					1 1001	
Data Sat		BR_{+}	-InfoGai	n 10%			Lazy N	ALInfoG	ain 10%	
Data Set	HLoss	SLoss	EbAcc	\mathbf{RLoss}	Time(s)	HLoss	SLoss	EbAcc	RLoss	Time(s)
Arts	0.0595	0.8991	0.8770	0.1941	$53,\!692$	0.0615	0.9213	0.9071	0.2027	1,111
Business	0.0267	0.4464	0.3000	0.0745	$93,\!634$	0.0273	0.4505	0.3062	0.0773	$1,\!590$
Computers	0.0360	0.6497	0.5900	0.1509	186,670	0.0362	0.6515	0.5887	0.1518	2,832
Education	0.0413	0.8771	0.8578	0.1658	142,035	0.0429	0.9109	0.8954	0.1693	2,188
Entertainment	0.0578	0.7621	0.7390	0.1778	125,560	0.0566	0.7993	0.7811	0.1708	2,750
Health	0.0430	0.6890	0.6141	0.1292	110,008	0.0442	0.6927	0.6147	0.1367	1,738
Recreation	0.0559	0.8262	0.8117	0.1990	122,812	0.0586	0.8828	0.8699	0.2128	2,584
Reference	0.0317	0.6342	0.6002	0.2009	133,902	0.0315	0.6523	0.6217	0.1818	2,141
Science	0.0343	0.9054	0.8940	0.2100	120,105	0.0346	0.9316	0.9240	0.2064	1,762
Social	0.0254	0.6204	0.5937	0.1277	334,846	0.0266	0.6500	0.6239	0.1249	4,795
Society	0.0537	0.7762	0.7207	0.1898	$215,\!605$	0.0549	0.7803	0.7264	0.1967	$3,\!694$

Table 6.5: Result of experiments on large data sets with BRKNN classifier, comparing BR+InfoGain with Lazy MLInfoGain feature selection

Even though the results from most measures favor slightly the BR + InfoGain technique for these Yahoo! directory data sets, the computational time of the Lazy MLInfoGain is significantly better, as it occurred with the non-lazy MLInfoGain technique.

6.5. ML-KNN Lazy Feature Selection

We have also incorporated the lazy attribute selection into the ML-KNN, which is another classifier capable of handling multi-label data directly. The lazy attribute selection was executed within the algorithm just before the actual classification takes place. Analogous to the BRKNN implementation, the classifier considered only the r features selected in a lazy manner to compute its neighbors distances, for each test instance. Again, this implies that for different instances distinct subsets of features were used. The experiments were executed with the default parameter settings in the Mulan tool, using the original ML-KNN classifier.

Table 6.6 shows the overall result of each feature selection technique coupled with the ML-KNN classifier. The results are reported similarly to the experiments with the BRKNN classifier in Table 6.3.

HAMMING LO	SS					
Data Set	BR+In foG ain	Copy+InfoG ain	LP+InfoGain	MLInfoGain	Lazy MLIn foGain	No Sel.
bibtex	$0.0126\ (20\%)$	0.0129 ($20%$)	0.0133 $(30%)$	$0.0130 \ (10\%)$	0.0135(70%)	0.0136
birds	0.0479(30%)	0.0477 (80%)	0.0472 ($90%$)	$0.0463 \ (10\%)$	0.0472 (90%)	0.0473
CAL500	0.1381 (40%)	0.1380(50%)	0.1381 (20%)	0.1380(70%)	0.1379 (70%)	0.1388
Corel5k	0.0094(10%)	0.0094(10%)	0.0094(10%)	0.0094(10%)	0.0094(70%)	0.0094
emotions	$\overline{0.1903(60\%)}$	0.1921(80%)	0.1966 (70%)	0.1898(90%)	0.1929(70%)	0.1951
enron	0.0502 (10%)	0.0531 (90%)	0.0502(10%)	0.0520(70%)	0.0517(50%)	0.0524
flagsml	$\frac{0.2489}{0.2489}$ (40%)	0.2622 (90%)	$\frac{0.2622}{0.2622}$ (90%)	0.2570 (90%)	0.2447 (80%)	0.2536
ganhasa	0.0048 (10%)	0.0048 (10%)	0.0048 (10%)	0.0048 (10%)	$\frac{0.2111}{0.0048}$ (10%)	0.0048
medical	$\frac{0.0040(1070)}{0.0126(10\%)}$	$\frac{0.0040(10\%)}{0.0140(10\%)}$	$\frac{0.0040(10\%)}{0.0148(10\%)}$	$\frac{0.0040(10\%)}{0.0147(10\%)}$	$\frac{0.0040}{0.0150}$ (10%)	0.0151
meurcar	$\frac{0.0120(1070)}{0.0800(0007)}$	0.0149 (1078)	0.0146(10%)	0.0147 (1076)	0.0130(2076)	0.0151
scene	0.0899 (90%)	0.0879(90%)	0.0911 (90%)	0.0807 (90%)	$\frac{0.0860}{0.1034}$ (80%)	0.0862
yeast	0.1915 (90%)	0.1935 (60%)	0,1945 (90%)	0.1925 (80%)	0,1934 (60%)	0,1933
SUBSET 0/1 LC	DSS					
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	Lazy MLIn foGain	No Sel.
bibtex	0.8619(20%)	0.8807(10%)	0.9154(30%)	0.8818(10%)	0.8955(10%)	0.9396
birds	0.5085 (40%)	0.5240 ($80%$)	$0.5210\ (90\%)$	$0.5100 \ (10\%)$	0.5116~(60%)	<u>0.5085</u>
CAL500	1.0000 (10%)	1.0000 (10%)	$1.0000 \ (10\%)$	$1.0000 \ (10\%)$	1.0000 (10%)	1.0000
Corel5k	0.9972 (90%)	0.9980 (90%)	0.9988 (90%)	$0.9980 \ (90\%)$	0.9958 (30%)	0.9982
emotions	0.6816 $(80%)$	0.6866 (70%)	0.7019 ($60%$)	0.6832 $(70%)$	0.7087 (60%)	0.7169
enron	0.9013 (10%)	0.9424~(90%)	0.9125(10%)	0.9172~(60%)	0.8996 $(50%)$	0.9260
flagsml	0.8087 (10%)	0.8500(40%)	0.8297 (60%)	0.8603 (80%)	0.8034 (70%)	0.8453
genbase	0.0890 (10%)	0.0890(10%)	0.0890 (10%)	0.0890(10%)	0.0890(10%)	0.0890
medical	$\overline{0.3967(10\%)}$	0.4816(10%)	0.4776(30%)	0.4633(10%)	0.4745(20%)	0.4940
scene	$\frac{0.3743}{0.3743}$ (90%)	0.3760 (90%)	0.3797 (80%)	0.3685(70%)	0.3722 (60%)	0.3752
veast	0.8097 (90%)	0.8101 (60%)	0.8192 (70%)	$\frac{0.8113}{0.8113}$ (80%)	0.8035(50%)	0.8126
EXAMPLE DAG	SED ACCURACY	(invented)	010102 (1070)	010110 (0070)	0.0000 (0070)	010120
Data Sot	BB + InfoGain	$\frac{1}{1}$	LP_InfoGain	MLInfoGain	Lagy ML Info Cain	No Sel
bibtor	0.7538 (20%)	0.7840 (10%)	0.8207 (20%)	0.7840 (10%)	0.7018 (10%)	0.8640
binda	$\frac{0.7538}{0.4510}$ (2070)	0 4622 (20%)	0.4617 (0.0%)	0.7849(1070) 0.4511(10%)	0.7518(1070)	0.3040
CIALEDO	0.4319 (8070)	0.4022 (8078)	0.4017 (9070)	0.4311(1076)	$\frac{0.4303}{0.7000}$ (4078)	0,4515
CALSOO				0.8007 (40%)	$\frac{0.7999}{0.000}$	0.8028
Corelak	0.9849 (90%)	0.9829 (90%)	0.9897 (90%)	0.9834(90%)	$\frac{0.9666}{20\%}$	0.9853
emotions	0.4427 (80%)	0.4507 (80%)	0,4601 (60%)	$\frac{0.4423}{0.00}$	0.4578 (70%)	0.4674
enron	0.6074 (10%)	0.6898 (90%)	0.6254 (10%)	0.6586 (60%)	0.5906(40%)	0.6684
flagsml	0.3679 (40%)	0.3982 (90%)	0.3917~(40%)	0.3863(40%)	0.3733 (80%)	0.3896
genbase	0.0584 (10%)	$0.0584 \ (10\%)$	$0.0584 \ (10\%)$	$0.0584 \ (10\%)$	0.0584 (10%)	0.0584
medical	0.3245 (10%)	0.4100 (10%)	0.4045~(30%)	$0.3902 \ (10\%)$	0.3997 ($20%$)	0.4187
scene	0.3280 (90%)	0.3322 (90%)	$0.3356\ (80\%)$	$0.3256\ (70\%)$	0.3270 (60%)	0.3330
yeast	0.4804 (90%)	0.4875~(60%)	0.4889 (90%)	0.4848 (80%)	0.4843~(90%)	0.4838
RANKING LOS	s					
Data Set	BR+In foG ain	Copy+InfoGain	LP+InfoGain	MLInfoGain	Lazy MLIn foGain	No Sel.
bibtex	0.1351 (20%)	0.1577 (10%)	0.1845(30%)	0.1563(10%)	0.1432(10%)	0.2083
birds	0.0742(10%)	0.0759(90%)	0.0753 (40%)	0.0724(80%)	0.0745 (70%)	0.0746
CAL500	0,1830 (30%)	0.1820(40%)	0.1823(40%)	0.1825(80%)	0.1823 (70%)	0.1828
Corel5k	0.1325(80%)	0.1340(80%)	0.1346 (90%)	0.1338 (60%)	0.1281(10%)	0.1340
emotions	0.1624 (50%)	0.1591 (80%)	0.1601 (90%)	0.1546 (60%)	$\frac{0.1201(10,0)}{0.1587(50\%)}$	0 1633
enron	0.0883(10%)	0.0919(70%)	0.0898(10%)	$\frac{0.0924}{0.0924}$	0.0892 (30%)	0.0920
flagsml	$\frac{0.0000}{0.1844}$ (40%)	0 1906 (90%)	0 1906 (90%)	0 1952 (30%)	0.1909 (40%)	0.2012
ganhaga	$\frac{0.1044}{0.0062}$ (10%)	0.0069 (10%)	0.0069(10%)	0.0062(100%)	0.0062 (10%)	0.0062
Bennase	0.0002 (1070)	$\frac{0.0002}{0.0384}$ (1070)	$\frac{0.0002}{0.0280}$ (1070)	$\frac{0.0002}{0.0202}$ (1070)	$\frac{0.0002}{0.0972}$ (10%)	0.0002
meurcar	0 0 0 0 0 0 1 0 0 7 1		11.11.389 1.311701	0.0393 [90%]	1.1.3 / 2 111 201	0.0390
	$\frac{0.0329 (10\%)}{0.0700 (00\%)}$			0.0707 (0.017)		0.0574
scene	$\frac{0.0329 (10\%)}{0.0799 (90\%)}$	0.0791 (90%)	0.0792 (90%)	0.0787 (90%)	0.0762 (60%)	0.0774
scene yeast	0.0329 (10%) 0.0799 (90%) 0.1636 (80%)	0.0384 (30%) 0.0791 (90%) 0.1644 (90%)	0.0792 (90%) 0.1660 (90%)	0.0787 (90%) 0.1649 (80%)	0.0762 (60%) 0.1663 (60%)	0.0774 0.1652
scene yeast Best values	0.0329 (10%) 0.0799 (90%) 0.1636 (80%) 23	0.0791 (90%) 0.1644 (90%) 7	0.0792 (90%) 0.1660 (90%) 7	0.0787 (90%) 0.1649 (80%) 13	$\frac{0.0762 \ (60\%)}{0.1663 \ (60\%)}$	0.0774 0.1652 7
scene yeast Best values (underlined)	0.0329 (10%) 0.0799 (90%) 0.1636 (80%) 23	0.0791 (90%) 0.1644 (90%) 7	0.0792 (90%) 0.1660 (90%) 7	0.0787 (90%) 0.1649 (80%) 13	$\frac{0.0762 (60\%)}{0.1663 (60\%)}$ 19	0.0774 0.1652 7
scene yeast Best values (underlined) seaseline score	0.0329 (10%) 0.0799 (90%) 0.1636 (80%) 23 39	0.0334 (30%) 0.0791 (90%) 0.1644 (90%) 7 28	0.0792 (90%) 0.1660 (90%) 7 27	0.0787 (90%) 0.1649 (80%) 13 37		0.0774 0.1652 7

Table 6.6: Best results achieved with the ML-KNN classifier, comparing feature selection techniques with the proposed LazyMLInfoGain technique

With the ML-KNN classifier, the proposed lazy multi-label information gain technique (Lazy MLInfoGain) also achieved a competitive result, holding the best performance in 19 cases, out of the 44 experiments. The non-lazy MLInfoGain technique achieved the

best result in 13 cases, and the BR+InfoGain transformation technique achieved the best result in 23 cases. Only in 7 cases the result without feature selection achieved the best result. In 40 cases, the proposed Lazy MLInfoGain was able to yield a value equal or better than the baseline (without feature selection).

The corresponding statistical evaluation yields a similar result when compared with the BRKNN classifier. These results indicate that the lazy MLInfoGain technique is also a competitive feature selection technique when coupled with other multi-label classifiers.

6.5.1. Experiments on Large Multi-label Data Sets for ML-KNN

This section reports the experiments on larger multi-label data sets, when the feature selection strategies are coupled with the ML-KNN classifier. The same 11 independently compiled data sets from the Yahoo! directory were chosen, each one with more than 5,000 instances and 30,000 features.

Feature selection strategies based on transformation, i.e., Copy+InfoGain, LP+InfoGain and BR+InfoGain were unable to yield a classification result in the proposed configuration: AMD FX 8210 8-Core 3.1 Ghz with 8 Gb of RAM and a 64 bit OS. An "Out of Memory" error occurred for all data sets. Besides a higher number of instances and features, each data set has also a varying number of labels between 20–40. The experiments showed that the ML-KNN is not as scalable as the BRKNN adaptation.

Table 6.7 shows the result of the experiment with larger data sets executed with the proposed Lazy MLInfoGain feature selection technique. Each row shows the result on a Yahoo data set. Columns "HLoss", "SLoss", "EbAcc" and "RLoss" show the result of the Hamming Loss, Subset 0/1 Loss, Example-based Accuracy (inverted) and Ranking Loss, respectively. Column "Time(s)" shows the total execution time of the experiment (feature selection time + classification time), in seconds. As the Lazy MLInfoGain does not rely on data transformation, it was able to run without problems. It is compared with the results of the BRKNN classifier presented before. Both classifiers achieve similar results in terms of performance, except for the Ranking Loss measure, which is better (lower) with the ML-KNN classifier. For instance, the ML-KNN execution of the Social data set and 10% of feature selection with Lazy MLInfoGain takes 15,298 seconds, or 4.24 hours.

The experiments with larger data sets and the ML-KNN classifier reinforced the conclusion that the proposed direct feature selection techniques – MLInfoGain and Lazy

Data Sat	BR-KNN + Lazy MLInfoGain 10%					ML-KNN + Lazy MLInfoGain 10%				
Data Set	HLoss	SLoss	EbAcc	RLoss	Time(s)	HLoss	SLoss	EbAcc	\mathbf{RLoss}	Time(s)
Arts	0.0615	0.9213	0.9071	0.2027	1,111	0.0625	0.9103	0.8903	0.1570	3,536
Business	0.0273	0.4505	0.3062	0.0773	1,590	0.0275	0.4581	0.3070	0.0376	7,043
Computers	0.0362	0.6515	0.5887	0.1518	2,832	0.0369	0.6367	0.5627	0.0782	$11,\!113$
Education	0.0429	0.9109	0.8954	0.1693	2,188	0.0427	0.9343	0.9226	0.0926	9,460
Entertainment	0.0566	0.7993	0.7811	0.1708	2,750	0.0571	0.7572	0.7280	0.1101	12,911
Health	0.0442	0.6927	0.6147	0.1367	1,738	0.0449	0.7113	0.6300	0.0633	$5,\!657$
Recreation	0.0586	0.8828	0.8699	0.2128	2,584	0.0595	0.8670	0.8468	0.1592	11,512
Reference	0.0315	0.6523	0.6217	0.1818	2,141	0.0322	0.8141	0.7978	0.0813	6,353
Science	0.0346	0.9316	0.9240	0.2064	1,762	0.0357	0.9175	0.9048	0.1317	4,545
Social	0.0266	0.6500	0.6239	0.1249	4,795	0.0272	0.6383	0.6114	0.0994	$15,\!298$
Society	0.0549	0.7803	0.7264	0.1967	$3,\!694$	0.0562	0.7663	0.7037	0.0739	11,309

Table 6.7: Result of experiments on large data sets with Lazy MLInfoGain (10%) feature selection and the BRKNN and ML-KNN classifiers

MLInfoGain – are more scalable than the well-known techniques used in the literature and which relies on transformation. The former techniques were able to yield a classification result for these larger data sets, while the latter techniques ran out of memory.

6.6. Chapter Summary

In this chapter, a new method for multi-label feature selection was proposed, based on the lazy paradigm. The lazy strategy for single-label feature selection was reviewed, and then a corresponding example of a multi-label data set was presented. This example indicated how a lazy strategy could also benefit the multi-label feature selection by postponing the selection to the classification moment.

The proposed lazy strategy for the multi-label context was implemented as a new direct feature selection method based on the information gain measure. An experimental evaluation was conducted with various multi-label feature selection methods and data sets from different domains. Two multi-label classifiers were used to assess the lazy adaptation: BRKNN and ML-KNN.

Experimental results and a statistical analysis confirmed that the Lazy MLInfoGain outperformed the non-lazy (eager) MLInfoGain proposed in the previous chapter. Since the MLInfoGain method was already competitive compared with other techniques, the lazy adaptation can be also considered as a competitive feature selection technique for multi-label classification. In terms of scalability, the proposed technique was able to run at faster times than the transformation-based techniques. Moreover, for the experiment with larger data sets coupled with the ML-KNN classifier, the transformation-based techniques were unable to yield a classification result due to memory restrictions. This reinforces the importance of employing direct feature selection methods for larger data sets. In the next chapter, we review the contributions of this work and present our concluding remarks.

Chapter 7

Conclusions

Multi-label classification is currently an important topic of research in the data mining area. It is a popular research topic and it has applicability in many relevant problems, such as: text categorization, biomolecular analysis, video classification, medical diagnosis, among others. In the last few years there has been substantial research in feature selection specifically for multi-label classification.

The main goal of this thesis is to contribute to the development of multi-label classification, and more specifically to the feature selection aimed at this task. The contributions are listed and reviewed below:

- (a) Correlation Analysis of Performance Measures for Multi-Label Classification (Chapter 3): There are many performance measures adapted from the single-label paradigm or developed specifically for the multi-label paradigm. Each different work in the area employs a distinct subset of measures, so it is difficult to compare results across them. The thesis presented an analysis of the correlation and relevance of performance measures for the multi-label classification task. Many measures were pointed as highly correlated with others, when using the Pearson Correlation and the Spearman Correlation. Next, guidelines for researchers were provided in order to select a suitable subset of measures when working with multi-label classification.
- (b) Multi-label Feature Selection (Chapter 4): The methods proposed for the feature selection specific to multi-label data sets are scattered in the multi-label classification literature, without a proposed categorization to describe them and to allow an objective comparison. This chapter reviews the feature selection problem for multi-label classification and formulates a taxonomy for categorizing the existing

feature selection techniques.

- (c) Information Gain Adaptation for Multi-label Data (Chapter 5): This chapter proposed the MLInfoGain feature selection technique. It consists of an adaptation of a single-label technique to the multi-label paradigm, using the information gain measure. It was compared experimentally with some well-known multi-label feature selection techniques. Experimental results indicated that the proposed strategy (MLInfoGain) achieved a competitive performance against the other techniques and outperformed the baseline on most cases. For larger data sets, the proposed technique scaled much better than the other feature selection methods, significantly outperforming the transformation-based techniques in terms of computational efficiency.
- (d) Lazy Multi-label Feature Selection (Chapter 6): The lazy strategy for singlelabel feature selection is based on the hypothesis that postponing the selection of features to the classification moment can contribute to identifying the best features for the correct classification of a particular instance. This chapter proposed a novel selection method for multi-label classification, based on the lazy feature selection paradigm and on the previous direct adaptation of the information gain measure. An experimental evaluation was conducted with various multi-label feature selection methods and data sets from different domains. Two multi-label classifiers were used to assess the lazy adaptation: BRKNN and ML-KNN. Experimental results and a statistical analysis confirmed that the Lazy MLInfoGain outperformed the nonlazy (eager) MLInfoGain proposed in the previous chapter. It was also a scalable method for feature selection, running faster than the techniques which relies on data transformation. Therefore, the lazy strategy was able to yield competitive results for the multi-label scenario.

7.1. Future Work

An unexplored subject in the multi-label feature selection scenario is to determine whether a specific category is able to achieve better experimental results when combined with a specific classification method or a specific multi-label domain. The evaluation in Chapters 5 and 6 indicated that direct filter techniques are much more efficient in terms of CPU time and scalable, when compared to transformation-based techniques. Further evaluations of specific combinations of classification and feature selection techniques are still an open topic for multi-label research. Other unexplored subjects in the multi-label feature selection domain which remain are: how well the current algorithms scale with respect to labels; how they handle class imbalance; the ability of methods to consider label correlations; an empirical comparison of representative methods from each category in order to better visualize the pros and cons of each one; evaluating and comparing the performance of direct multi-label featureselection methods in the filter, wrapper and embedded categories; and evaluating and comparing methods which apply binary relevance feature selection externally with the ones which apply it internally.

Experimental results in this thesis compared filter feature selection techniques, either based on data transformation or as a direct approach. But the performance of the classification depended on the number of features selected, which is a user-defined parameter. In practice, it may be difficult to select a proper value for this parameter, that is, the value that produces the best performance for the classification task. In [51], two approaches to overcome this drawback were proposed: the use of a wrapper-based strategy and the combination of multiple number of features using a voting approach. As future work, these strategies can be adapted to multi-label feature selection techniques.

The proposed direct multi-label feature selection adaptations (MLInfoGain and Lazy MLInfoGain) require discrete feature values. However, there is no supervised technique currently in use in the literature for discretizing multi-label data sets. It is an open problem for future work that could benefit multiple techniques in the multi-label area.

Furthermore, regarding extensions of the Lazy MLInfoGain technique, it is possible to adapt other measures for the multi-label context besides information gain. A topic of investigation is the adaptation of measures that evaluate subsets of features, like Correlation-based Feature Selection [28] and Consistency-based Feature Selection [43].

References

- [1] BOUTELL, M. R.; LUO, J.; SHEN, X.; BROWN, C. M. Learning multi-label scene classification. *Pattern recognition* 37, 9 (2004), 1757–1771.
- [2] BURGES, C. J. C. A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery 2 (1998), 121–168.
- [3] CHEN, W.; YAN, J.; ZHANG, B.; CHEN, Z.; YANG, Q. Document transformation for multi-label feature selection in text categorization. In *Proceedings of the 7th IEEE International Conference on Data Mining* (2007), pp. 451–456.
- [4] CHENG, W.; HÜLLERMEIER, E. Combining instance-based learning and logistic regression for multilabel classification. *Machine Learning* 76, 2-3 (2009), 211–225.
- [5] CLARE, A.; KING, R. D. Knowledge discovery in multi-label phenotype data. In Proceedings of the 5th European Conference on Principles of Data Mining and Knowledge Discovery (2001), pp. 42–53.
- [6] COVER, T.; HART, P. Nearest neighbor pattern classification. *IEEE Transactions* on Information Theory 13 (1967), 21–27.
- [7] CRAMMER, K.; SINGER, Y.; K, J.; HOFMANN, T.; POGGIO, T.; SHAWE-TAYLOR, J. A family of additive online algorithms for category ranking. *Journal of Machine Learning Research* 3 (2003), 1025–1058.
- [8] DA SILVA, P. N.; GONÇALVES, E. C.; PLASTINO, A.; FREITAS, A. A. Distinct chains for different instances: An effective strategy for multi-label classifier chains. In Proceedings of 7th tEuropean Conference on Machine Learning and Knowledge Discovery in Databases (2014), pp. 453-468.
- [9] DASARATHY, B. V. Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, 1991.
- [10] DASH, M.; LIU, H. Feature selection for classification. Intelligent Data Analysis 1 (1997), 131–156.
- [11] DE CARVALHO, A. C.; FREITAS, A. A. A tutorial on multi-label classification techniques. In Foundations of Computational Intelligence Volume 5. Springer, 2009, pp. 177–195.
- [12] DE COMITÉ, F.; GILLERON, R.; TOMMASI, M. Learning multi-label alternating decision trees from texts and data. In Proceedings of the 3rd International Conference on Machine Learning and Data Mining in Pattern Recognition (2003), Springer-Verlag, pp. 35–49.

- [13] DENDAMRONGVIT, S.; VATEEKUL, P.; KUBAT, M. Irrelevant attributes and imbalanced classes in multi-label text-categorization domains. *Intelligent Data Analysis* 15, 6 (2011), 843–859.
- [14] DOQUIRE, G.; VERLEYSEN, M. Feature selection for multi-label classification problems. In Proceedings of the 11th Conference on Artificial Neural Networks on Advances in Computational Intelligence (2011), Springer-Verlag, pp. 9–16.
- [15] DUDA, R. O.; HART, P. E.; STORK, D. G. Pattern Classification, 2nd ed. John Wiley & Sons, 2001.
- [16] ELISSEEFF, A.; WESTON, J. A kernel method for multi-labelled classification. In Advances in Neural Information Processing Systems (2001), vol. 14, pp. 681–687.
- [17] FAYYAD, U. M.; IRANI, K. B. Multi-interval discretization of continuous-valued attributes for classification learning. In *Proceedings of the 13th International Joint Conference on Artificial Intelligence* (1993), pp. 1022–1029.
- [18] FORMAN, G. A pitfall and solution in multi-class feature selection for text classification. In Proceedings of the 21st International Conference on Machine Learning (2004), ACM, pp. 1–38.
- [19] FÜRNKRANZ, J.; HÜLLERMEIER, E.; LOZA MENCÍA, E.; BRINKER, K. Multilabel classification via calibrated label ranking. *Machine Learning* 73, 2 (2008), 133–153.
- [20] GHAMRAWI, N.; MCCALLUM, A. Collective multi-label classification. In Proceedings of the 14th ACM International Conference on Information and Knowledge Management (2005), ACM, pp. 195–200.
- [21] GODBOLE, S.; SARAWAGI, S. Discriminative methods for multi-labeled classification. In Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining (2004), Springer, pp. 22–30.
- [22] GONÇALVES, E. C. Novel Classifier Chain Methods for Multi-label Classification Based on Genetic Algorithms. PhD Thesis, Universidade Federal Fluminense (UFF), Brazil, 2015.
- [23] GONÇALVES, E. C.; PLASTINO, A.; FREITAS, A. A. A genetic algorithm for optimizing the label ordering in multi-label classifier chains. In *Proceedings of the* 25th International Conference on Tools with Artificial (2013), pp. 469–476.
- [24] GONÇALVES, E. C.; PLASTINO, A.; FREITAS, A. A. Simpler is better: a novel genetic algorithm to induce compact multi-label chain classifiers. In *Proceedings of* the Genetic and Evolutionary Computation Conference (2015), pp. 559–566.
- [25] GU, Q.; LI, Z.; HAN, J. Correlated multi-label feature selection. In Proceedings of the 20th ACM International Conference on Information and Knowledge Management (2011), pp. 1087–1096.
- [26] GUYON, I.; ELISSEEFF, A. An introduction to feature extraction. In *Feature Extraction, Foundations and Applications*. Springer, 2006, pp. 1–24.

- [27] GUYON, I.; GUNN, S.; NIKRAVESH, M.; ZADEH, L., Eds. Feature Extraction, Foundations and Applications. Springer, 2006.
- [28] HALL, M. A. A correlation-based feature selection for discrete and numeric class machine learning. In Proceedings of the 17th International Conference on Machine Learning (2000).
- [29] HÜLLERMEIER, E.; FÜRNKRANZ, J.; CHENG, W.; BRINKER, K. Label ranking by learning pairwise preferences. Artificial Intelligence 172, 16-17 (2008), 1897–1916.
- [30] JUNGJIT, S.; MICHAELIS, M.; FREITAS, A. A.; CINATL, J. Two extensions to multi-label correlation-based feature selection: a case study in bioinformatics. In *IEEE International Conference on Systems, Man, and Cybernetics* (2013), IEEE, pp. 1519–1524.
- [31] KIRA, K.; RENDELL, L. A practical approach to feature selection. In *Proceedings* of the 9th International Conference on Machine Learning (1992), pp. 249–256.
- [32] KOCEV, D.; SLAVKOV, I.; DZEROSKI, S. Feature ranking for multi-label classification using predictive clustering trees. In International Workshop on Solving Complex Machine Learning Problems with Ensemble Methods, in Conjunction with ECML/PKDD (2013), pp. 56–68.
- [33] KOHAVI, R.; JOHN, G. H. Wrappers for feature subset selection. Artificial Intelligence 97, 1-2 (1997), 273–324.
- [34] KONG, X.; YU, P. S. gmlc: a multi-label feature selection framework for graph classification. *Knowledge Information Systems* 31, 2 (2012), 281–305.
- [35] KONONENKO, I. Estimating attributes: Analysis and extensions of relief. In Proceedings of the 7th European Conference on Machine Learning (1994), pp. 171–182.
- [36] LASTRA, G.; LUACES, O.; QUEVEDO, J. R.; BAHAMONDE, A. Graphical feature selection for multilabel classification tasks. In *Proceedings of the 10th International Conference on Advances in Intelligent Data Analysis* (2011), pp. 246–257.
- [37] LEE, J.; KIM, D.-W. Feature selection for multi-label classification using multivariate mutual information. *Pattern Recognition Letters* 34, 3 (2013), 349–357.
- [38] LI, G.-Z.; YOU, M.; GE, L.; YANG, J. Y.; YANG, M. Q. Feature selection for semi-supervised multi-label learning with application to gene function analysis. In Proceedings of the 1st ACM International Conference on Bioinformatics and Computational Biology (2010), pp. 354-357.
- [39] LI, L.; LIU, H.; MA, Z.; MO, Y.; DUAN, Z.; ZHOU, J.; ZHAO, J. Multi-label feature selection via information gain. In Advanced Data Mining and Applications, Lecture Notes in Computer Science. Springer International Publishing, 2014, pp. 345– 355.
- [40] LIU, B.; HSU, W.; MA, Y. Integrating classification and association rule mining. In Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining (1998), pp. 80-86.

- [41] LIU, H.; MOTODA, H. Computational Methods of Feature Selection. Chapman & Hall/CRC, 2008.
- [42] LIU, H.; MOTODA, H. Less is more. In Computational Methods of Feature Selection. Chapman & Hall/CRC, 2008, pp. 3–17.
- [43] LIU, H.; SETIONO, R. A probabilistic approach to feature selection: A filter solution. In Proceedings of the 13th International Conference on Machine Learning (1996), pp. 319-327.
- [44] LIU, H.; YU, L. Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering* 17, 4 (2005), 491–502.
- [45] LIU, Y.; JIN, R.; YANG, L. Semi-supervised multi-label learning by constrained non-negative matrix factorization. In *Proceedings of the 21st National Conference on Artificial Intelligence* (2006), pp. 421–426.
- [46] MADJAROV, G.; KOCEV, D.; GJORGJEVIKJ, D.; DVZEROSKI, S. An extensive experimental comparison of methods for multi-label learning. *Pattern Recognition* 45, 9 (2012), 3084–3104.
- [47] MENCÍA, E. L.; FURNKRANZ, J. Pairwise learning of multilabel classifications with perceptrons. In Proceeding of the 2008 IEEE International Joint Conference on Neural Networks (2008), pp. 2899–2906.
- [48] MOLINA, L. C.; BELANCHE, L.; NEBOT, A. Feature selection algorithms: A survey and experimental evaluation. In *Proceedings of the 2002 IEEE International Conference on Data Mining* (2002), pp. 306–313.
- [49] OLSSON, J.; OARD, D. W. Combining feature selectors for text classification. In Proceedings of the 15th ACM International Conference on Information and Knowledge Management (2006), ACM, pp. 798–799.
- [50] PEREIRA, R. B.; PLASTINO, A.; ZADROZNY, B.; MERSCHMANN, L. H. Information gain feature selection for multi-label classification. *Journal of Information and Data Management* 6, 1 (2015), 48.
- [51] PEREIRA, R. B.; PLASTINO, A.; ZADROZNY, B.; MERSCHMANN, L. H. D. C.; FREITAS, A. A. Improving lazy attribute selection. *Journal of Information and Data Management 2*, 3 (2011), 447.
- [52] PEREIRA, R. B.; PLASTINO, A.; ZADROZNY, B.; MERSCHMANN, L. H. D. C.; FREITAS, A. A. Lazy attribute selection – choosing attributes at classification time. Intelligent Data Analysis 15, 5 (2011), 715–732.
- [53] PUPO, O. G. R.; MORELL, C.; SOTO, S. V. Relieff-ml: An extension of relieff algorithm to multi-label learning. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications.* Springer, 2013, pp. 528–535.
- [54] QUINLAN, J. R. Induction of decision trees. Machine Learning 1 (1986), 81–106.
- [55] QUINLAN, J. R. C4.5: Programs for Machine Learning. Morgan Kaufmann, 1993.

- [56] READ, J. A pruned problem transformation method for multi-label classification. In Proceedings of the New Zealand Computer Science Research Student Conference (2008), pp. 143-150.
- [57] READ, J. Scalable Multilabel Classification. Phd dissertation, Hamilton, New Zealand, 2010.
- [58] READ, J.; PFAHRINGER, B.; HOLMES, G.; FRANK, E. Classifier chains for multilabel classification. In Proceedings of the 20th European Conference on Machine Learning and Knowledge Discovery in Databases (2009), pp. 254–269.
- [59] READ, J.; PFAHRINGER, B.; HOLMES, G.; FRANK, E. Classifier chains for multilabel classification. *Machine learning* 85, 3 (2011), 333–359.
- [60] RIPLEY, B. D. Pattern Recognition and Neural Networks. Cambridge University Press, 1996.
- [61] RISSANEN, J. Stochastic complexity and modeling. Annals of Statistic 14, 3 (1986), 1080–1100.
- [62] ROGATI, M.; YANG, Y. High-performing feature selection for text classification. In Proceedings of the 11th International Conference on Information and Knowledge Management (2002), ACM, pp. 659–661.
- [63] SAEYS, Y.; INZA, I.; LARRANAGA, P. A review of feature selection techniques in bioinformatics. *Bioinformatics* 23, 19 (2007), 2507–2517.
- [64] SECHIDIS, K.; NIKOLAOU, N.; BROWN, G. Information theoretic feature selection in multi-label data through composite likelihood. In *Structural, Syntactic, and Statistical Pattern Recognition*. Springer, 2014, pp. 143–152.
- [65] SHAO, H.; LI, G.; LIU, G.; WANG, Y. Symptom selection for multi-label data of inquiry diagnosis in traditional chinese medicine. *Science China Information Sciences* 56, 5 (2013), 1–13.
- [66] SOKOLOVA, M.; LAPALME, G. A systematic analysis of performance measures for classification tasks. *Information Processing & Management* 45, 4 (2009), 427–437.
- [67] SOROWER, M. S. A literature survey on algorithms for multi-label learning. Oregon State University, Corvallis 63 (2010).
- [68] SPOLAÔR, N.; CHERMAN, E. A.; MONARD, M. C.; LEE, H. D. A comparison of multi-label feature selection methods using the problem transformation approach. *Electronic Notes in Theoretical Computer Science 292* (2013), 135–151.
- [69] SPOLAÔR, N.; CHERMAN, E. A.; MONARD, M. C.; LEE, H. D. Relieff for multilabel feature selection. In Proceedings of the 2nd Brazilian Conference on Intelligent Systems (2013), IEEE, pp. 6–11.
- [70] SPOLAÔR, N.; MONARD, M. C. Evaluating relieff-based multi-label feature selection algorithm. In Proceedings of the 14th edition of the Ibero-American Conference on Artificial Intelligence. Springer, 2014, pp. 194–205.

- [71] SPOLAÔR, N.; TSOUMAKAS, G. Evaluating feature selection methods for multi-label text classification. *BioASQ Workshop* (2013).
- [72] TANG, L.; RAJAN, S.; NARAYANAN, V. K. Large scale multi-label classification via metalabeler. In Proceedings of the 18th International Conference on World Wide Web (2009), ACM, pp. 211–220.
- [73] TROHIDIS, K.; TSOUMAKAS, G.; KALLIRIS, G.; VLAHAVAS, I. P. Multi-label classification of music into emotions. In *Proceedings of the 9th International Conference* on *Music Information Retrieval* (2008), J. P. Bello, E. Chew, and D. Turnbull, Eds., pp. 325–330.
- [74] TSOUMAKAS, G.; DIMOU, A.; SPYROMITROS, E.; MEZARIS, V.; KOMPATSIARIS, I.; VLAHAVAS, I. Correlation based pruning of stacked binary relevance models for Multi-Label learning. In *Proceedings of the 1st International Workshop on Learning* from Multi-Label Data (2009), pp. 101–116.
- [75] TSOUMAKAS, G.; KATAKIS, I. Multi-label classification: An overview. International Journal of Data Warehousing and Mining (2007), 1–13.
- [76] TSOUMAKAS, G.; KATAKIS, I.; VLAHAVAS, I. Effective and efficient multilabel classification in domains with large number of labels. In ECML/PKDD 2008 Workshop on Mining Multidimensional Data (2008), pp. 30–44.
- [77] TSOUMAKAS, G.; KATAKIS, I.; VLAHAVAS, I. Mining multi-label data. In Data Mining and Knowledge Discovery Handbook, O. Maimon and L. Rokach, Eds. Springer US, 2010, pp. 667–685.
- [78] TSOUMAKAS, G.; VLAHAVAS, I. Random k-labelsets: An ensemble method for multilabel classification. In Proceedings of the 18th European Conference on Machine Learning (2007), pp. 406–417.
- [79] WITTEN, I. H.; FRANK, E.; M., H. Data Mining: Practical Machine Learning Tools and Techniques, 3rd ed. Morgan Kaufmann, 2011.
- [80] YANG, Y.; PEDERSEN, J. O. A comparative study on feature selection in text categorization. In Proceedings of the 14th International Conference on Machine Learning (1997), pp. 412–420.
- [81] YU, K.; YU, S.; TRESP, V. Multi-label informed latent semantic indexing. In Proceedings of the 28th ACM SIGIR Conference on Research and Development in Information Retrieval (2005), pp. 258-265.
- [82] YU, L.; LIU, H. Efficient feature selection via analysis of relevance and redundancy. The Journal of Machine Learning Research 5 (2004), 1205–1224.
- [83] ZHANG, M.-L.; PEÑA, J. M.; ROBLES, V. Feature selection for multi-label naive bayes classification. *Information Sciences* 179, 19 (2009), 3218–3229.
- [84] ZHANG, M.-L.; ZHOU, Z.-H. Multilabel neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering 18* (2006), 1338–1351.

- [85] ZHANG, M.-L.; ZHOU, Z.-H. Ml-knn: A lazy learning approach to multi-label learning. *Pattern Recognition* 40, 7 (2007), 2038–2048.
- [86] ZHANG, M.-L.; ZHOU, Z.-H. A review on multi-label learning algorithms. IEEE Transactions on Knowledge and Data Engineering 26 (2014), 1819–1837.
- [87] ZHANG, X.; YUAN, Q.; ZHAO, S.; FAN, W.; ZHENG, W.; WANG, Z. Multilabel Classification without the Multi-label cost. In Proceedings of the Tenth SIAM International Conference on Data Mining (2010).
- [88] ZHANG, Y.; ZHOU, Z.-H. Multilabel dimensionality reduction via dependence maximization. ACM Transactions Knowledge Discovery Data 4, 3 (2010), 1411–1421.
- [89] ZHENG, Z.; WU, X.; SRIHARI, R. Feature selection for text categorization on imbalanced data. ACM SIGKDD Explorations Newsletter 6, 1 (2004), 80–89.
- [90] ZOU, K. H.; TUNCALI, K.; SILVERMAN, S. G. Correlation and simple linear regression 1. Radiology 227, 3 (2003), 617–628.

Appendix A - Transformations on Multi-Label Data Sets

This Appendix shows a multi-label data set and the corresponding single-label data sets after applying the most common transformation methods.

Table 7.1 presents a simple multi-label example. The data set is composed of two features -X, Y - and their labels -A, B or C.

	-	
– X –	– Y –	– Labels –
1	1	A
1	2	В
1	3	В
1	4	A,B
2	1	A
2	2	A
2	3	A
2	4	A,B
3	1	B,C
3	2	B,C
3	3	B,C
3	4	A,B
4	1	В
4	2	A,C
4	3	A
4	4	A,B

Table 7.1: Multi-label Data Set Example

Table 7.2 shows the result of transforming the previous multi-label example into a single-label data set using a copy transformation. This transformation consists in copying each multi-label instance n times, where n is the number of labels assigned to that instance. Each copied instance is then assigned one distinct single label from the original set. One of the characteristics of this transformation is the increase of the number of instances. The original multi-label data set has 16 instances, and the transformed data set has 24 instances. Another characteristic is the presence of instances with the same feature values, due to the copying process.

Table 7.3 shows the result of transforming the multi-label example into a single-label data set using a label powerset transformation. It creates one new label for each different subset of labels that exists in the original multi-label data set. One of the characteristics

- X -	– Y –	– Labels –
1	1	A
1	2	В
1	3	В
1	4	A
1	4	В
2	1	A
2	2	A
2	3	A
2	4	A
2	4	В
3	1	В
3	2	В
3	3	В
3	4	A
3	1	C
3	2	C
3	3	C
3	4	В
4	1	В
4	2	A
4	2	C
4	3	A
4	4	A
4	4	В

Table 7.2: Data Set Example after Copy transformation

of this transformation is preserving the number of instances. However, the number of distinct labels (or classes) increases: 3 labels in the original multi-label data set becomes the powerset A, B, AB, AC and BC in the example. The missing C and ABC label combinations do not appear in the original data set.

– X –	– Y –	– Labels –
1	1	A
1	2	В
1	3	В
1	4	AB
2	1	A
2	2	A
2	3	A
2	4	AB
3	1	BC
3	2	BC
3	3	BC
3	4	AB
4	1	В
4	2	AC
4	3	A
4	4	AB

Table 7.3: Data Set Example after LP transformation

Table 7.4 shows the result of transforming the multi-label example into three singlelabel data sets using a binary relevance transformation. It produces a binary classifier for each different label of the original data set. The new labels L_A , L_B and L_C are binary labels, which can assume the value of 0 or 1.

[C	Data Set L_A			Data Set	L_B	Data Set L_C		
- X -	– Y –	- L _A -	– X –	– Y –	- L _B -	– X –	– Y –	- L _C -
1	1	1	1	1	0	1	1	0
1	2	0	1	2	1	1	2	0
1	3	0	1	3	1	1	3	0
1	4	1	1	4	1	1	4	0
2	1	1	2	1	0	2	1	0
2	2	1	2	2	0	2	2	0
2	3	1	2	3	0	2	3	0
2	4	1	2	4	1	2	4	0
3	1	0	3	1	1	3	1	1
3	2	0	3	2	1	3	2	1
3	3	0	3	3	1	3	3	1
3	4	1	3	4	1	3	4	0
4	1	0	4	1	1	4	1	0
4	2	1	4	2	0	4	2	1
4	3	1	4	3	0	4	3	0
4	4	1	4	4	1	4	4	0

Table 7.4: Data Set Example after BR transformation

The three transformation strategies presented in this Appendix create a resulting single-label data set which can be used by any traditional single-label classification technique. However, each case has an overhead created by the transformation: the increase of instances (for the Copy transformation), the increase of labels or classes (for the LP transformation) or the increase of the number of data sets (for the BR transformation). This thesis focuses on feature selection techniques which do not rely on transformation. Instead, the proposed techniques are able to handle the original multi-label data set directly.

Appendix B - MLInfoGain Compared with Transformation-based Techniques

This Appendix shows the complete results of each feature selection technique used in Chapter 5 coupled with various classification algorithms. The tables are analogous to Table 5.4. Each table section presents the result for a specific performance measure. The first column indicates the data set used. "BR+InfoGain", "Copy+InfoGain" and "LP+InfoGain" stand for a transformation followed by the single-label information gain measure to rank and select features. "MLInfoGain" corresponds to the multi-label information gain technique proposed in this work. "No Sel." is the result without feature selection, and also our baseline. Each cell shows the result of the multi-label measure achieved in the best case among the different percentages used in the experiment. The evaluation measures vary between 0 and 1, and the lower the value, the better. In parenthesis it is indicated the percentage of selected features that achieved the best value for each technique, and in case of ties the smaller percentage is reported. Bold values show the results that achieved a result equal or better than the baseline. Underlined values show the best result achieved in each row, for the given data set. The "n/a" value indicates that the experiment did not finish due to an out of memory error.

At the end of each table the results are summarized. The "Best values (underlined)" shows the number of times that the technique achieved the best value in the experiment. The " \leq baseline score (bold)" shows the number of times that the technique achieved a value equal or better than the classification without feature selection.

HAMMING LO	DSS				
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.0141~(10%)	0.0146~(20%)	0.0161~(20%)	0.0146~(20%)	0.0201
birds	0.0501 (90%)	0.0498 $(80%)$	0.0505 $(80%)$	0.0498~(50%)	0.0510
CAL500	0.1726(60%)	$\overline{0.1718}$ (40%)	0.1723(20%)	$\overline{0.1727}$ (30%)	0.1741
Corel5 k	n/a	n/a	n/a Ś	n/a	n/a
emotions	0.2004(70%)	$0.1999^{'}$ (70%)	0.2027 (60%)	0.1985 (60%)	0.2095
enron	0.0531(10%)	0.0579 (90%)	0.0532(10%)	$\overline{0.0573(90\%)}$	0.0578
flagsml	$\frac{0.0001}{0.2460}$ (50%)	0.2576 (80%)	0.2516 (20%)	0.2482(50%)	0.2592
genhase	$\frac{0.2100}{0.0022}$ (10%)	0.0022 (10%)	0.0022 (10%)	0.0022 (10%)	0.0022
medical	$\frac{0.0022}{0.0132}$ (10%)	$\frac{0.0022}{0.0153}$ (10%)	$\frac{0.0022}{0.0152}$ (10%)	$\frac{0.0022}{0.0153}$ (10%)	0.0171
scono	$\frac{0.0102}{0.0064}$ (00%)	0.0100(1070)	0.0102 (10%)	0.0100(1070)	0.0037
veset	0.0904 (9070)	0.0949 (9070)	0.0932 (9078)	0.0940 (80%)	0.0937 0.2182
	0.2100 (0070)	0.2110 (0070)	0.2110 (0070)	0.2102 (0070)	0.2102
Deta Set		Conv InfoCoin	ID InfoCain	MIInfoCoin	No Sol
bibtov	0.8440 (10%)	$\frac{0.8744}{10\%}$	$\frac{1100000}{0000}$	0.8718 (10%)	0.0586
DIDLEX	$\frac{0.8449}{0.4000}$ (1070)	0.8744 (1076)	0.9202 (30%)	0.8718(1076)	0.9580
DIrds	$\frac{0.4900(60\%)}{1.0000(100\%)}$	0.5100(80%)	0.5100(80%)	0.3007 (80%)	0.3102
CAL500	$\frac{1.0000(10\%)}{1.0000}$	$\frac{1.0000(10\%)}{1.0000}$	1.0000(10%)	$\frac{1.0000(10\%)}{1.0000}$	1.0000
Corel5 k				n/a	n/a
emotions	$\frac{0.6833}{0.6833}$	0.6866 (70%)	0.6866 (60%)	0.6850 (80%)	0.7137
enron	0.8743(10%)	0.8813(50%)	0.8825(50%)	$\frac{0.8731}{0.8731}$ (50%)	0.9060
flagsml	0.7976(20%)	$0.8245 \ (60\%)$	0.7990 (10%)	0.7832(10%)	0.8340
genbase	0.0468~(10%)	0.0468~(10%)	0.0468~(10%)	0.0468~(10%)	0.0468
medical	0.4059~(10%)	0.4837~(10%)	0.4816~(10%)	0.4796~(10%)	0.5277
scene	0.3818~(90%)	0.3747~(90%)	0.3760~(90%)	0.3698~(80%)	0.3727
yeast	0.7989 $(80%)$	0.7973~(80%)	0.7956~(90%)	$\overline{0.7923} (90\%)$	0.7989
EXAMPLE BA	SED ACCURAC	CY (inverted)			
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.7049~(10%)	0.7522~(10%)	0.8169 $(30%)$	0.7492~(10%)	0.8733
birds	0.4122(60%)	0.4312 (80%)	0.4320 (80%)	0.4173(20%)	0.4379
CAL500	$\overline{0.7439}$ (60%)	0.7412(40%)	0.7417(20%)	0.7437(60%)	0.7460
Corel5 k	n/a í	n/a	n/a	n/a Ó	n/a
emotions	0.4222 (70%)	0.4250 (80%)	0.4305(60%)	0.4243 (60%)	0.4412
enron	$\overline{0.6054(10\%)}$	0.6478(10%)	0.6355(10%)	0.6401(50%)	0.7387
flagsml	$\overline{0.3686}$ (60%)	0.3792(60%)	0.3711(20%)	0.3654(50%)	0.3857
genbase	0.0235 (10%)	0.0235 (10%)	0.0235 (10%)	$\frac{0.0235}{0.0235}$ (10%)	0.0235
medical	$\frac{0.0200(10\%)}{0.3277(10\%)}$	$\frac{0.0200}{0.4074}$ (10%)	$\frac{0.0200(10\%)}{0.4063(10\%)}$	$\frac{0.0200(10\%)}{0.4053(10\%)}$	$\frac{0.0200}{0.4547}$
scene	$\frac{0.0211}{0.3446}$ (00%)	0.3385 (00%)	0.3387 (00%)	0.3327 (80%)	0.3300
voast	0.4575 (00%)	0.0000 (0070)	0.0001 (00%)	$\frac{0.0021}{0.4583}$ (60%)	0.3550
		0.4000 (0070)	0.4300 (3070)	0.4000 (0070)	0.4000
Deta Set	DD InfoCain	Conv InfoCoin	ID InfoCain	MIInfoCoin	No Sol
bibtov	0.2643 (10%)	0.2085 (10%)	1100000000000000000000000000000000000	0.2040(10%)	0.3868
biblex	$\frac{0.2043(1070)}{0.1771(0007)}$	0.2303 (1070)	0.3442 (3070)	0.2949(1070)	0.3808
DIFUS	0.1771 (90%)	0.1713 (20%)	0.1755(70%)	$\frac{0.1043}{0.9720}$ (10%)	0.1813
CAL500	0.3080 (70%)	$\frac{0.3674}{10\%}$	0.3675 (20%)	0.3720(90%)	0.3734
Corel5k			n/a		n/a
emotions	$\frac{0.1929}{0.0500}$	0.1947 (80%)	0.2026 (30%)	0.1988 (60%)	0.2091
enron	0.2523(10%)	$\frac{0.2505}{0.2505}$	0.2702(10%)	0.2626(10%)	0.2887
flagsml	$\frac{0.2275}{0.2275}$ (50%)	0.2382 (60%)	0.2390(20%)	0.2369 (90%)	0.2479
genbase	0.0071 (10%)	0.0071 (10%)	0.0071 (10%)	0.0071 (10%)	0.0071
medical	$0.0934 \ (10\%)$	0.1208~(10%)	0.1138~(10%)	0.1161~(10%)	0.1405
scene	$0.1420 \ (90\%)$	0.1405~(90%)	0.1397~(90%)	0.1374~(80%)	0.1421
yeast	0.2357 (90%)	0.2377 $(80%)$	0.2365 (90%)	$0.2357\ (60\%)$	0.2366
Best values	23	10	6	17	6
(underlined)	20	10	U	± 1	
\leq baseline score	37	36	38	39	
(bold)		50	50	50	

Table 7.1: Best results achieved with the HOMER + K-NN classifier

HAMMING LC	255				
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	n/a	n/a	n/a	n/a	n/a
birds	0.0489~(10%)	0.0543~(90%)	0.0545~(90%)	0.0521~(70%)	0.0530
CAL500	n/a	n/a	n/a	n/a	n/a
Corel5 k	$0.0125 \ (10\%)$	0.0125~(10%)	0.0120~(40%)	0.0124~(10%)	0.0139
emotions	0.2457 (50%)	0.2488~(60%)	0.2463~(90%)	0.2466~(60%)	0.2491
enron	0.0603 (10%)	0.0678~(10%)	0.0616~(10%)	0.0681~(10%)	0.0689
flagsml	0.2695(60%)	$0.2717\ (90\%)$	0.2674~(80%)	0.2658~(90%)	0.2709
genbase	0.0059 (10%)	0.0059~(10%)	0.0059~(10%)	$\overline{0.0059} (10\%)$	<u>0.0059</u>
medical	$\overline{0.0193}$ (10%)	$\overline{0.0217}$ (10%)	0.0220 (10%)	$\overline{0.0215}$ (10%)	0.0250
scene	0.1246 (90%)	0.1240(90%)	0.1247(90%)	0.1228(70%)	0.1214
yeast	0.2500 (70%)	0.2491 (90%)	0.2540(80%)	0.2516(80%)	0.2512
SUBSET 0/1 L	OSS	<u>, , ,</u>			
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	n/a	n/a	n/a	n/a	n/a
birds	0.5054 (60%)	0.5443~(90%)	0.5348~(10%)	0.5303~(80%)	0.5287
CAL500	n/a	n/a	n/a	n/a	n/a
Corel5 k	0.9920 (60%)	0.9940 (60%)	0.9938 $(30%)$	0.9942~(60%)	0.9982
emotions	0.7673(60%)	0.7792 (70%)	0.7893 (90%)	0.7675~(60%)	0.8011
enron	$\overline{0.8878(10\%)}$	0.8972 (70%)	0.8966(10%)	0.8961(60%)	0.9213
flagsml	$\overline{0.8708}$ (50%)	0.8808 (70%)	0.8755(60%)	0.8655(50%)	0.8858
genbase	0.1072 (10%)	0.1072(10%)	0.1072(10%)	$\overline{0.1072(10\%)}$	0.1072
medical	$\overline{0.5501(10\%)}$	$\overline{0.6064(10\%)}$	$\overline{0.6064(10\%)}$	$\overline{0.6003(10\%)}$	0.6882
scene	0.5251 (90%)	0.5251 (90%)	0.5293 (90%)	0.5160(70%)	0.5160
veast	0.8664 (90%)	0.8622 (90%)	0.8626 (90%)	0.8655(90%)	$\frac{0.8622}{0.8622}$
EXAMPLE BA	SED ACCURAC	CY (inverted)			
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
		10			
bibtex	n/a	n/a	n/a	n/a	n/a
bibtex birds	n/a 0.3998 (60%)	$^{n/a}_{0.4363}$ (90%)	n/a 0.4162 (10%)	$^{n/a}_{0.4209\ (10\%)}$	n/a 0.4191
bibtex birds CAL500	$\frac{\frac{n/a}{a}}{\frac{0.3998}{n/a}}$	${n/a} \ 0.4363 \ (90\%) \ n/a$	n/a 0.4162 (10%) n/a	${n/a} \ 0.4209 \ (10\%) \ n/a$	n/a 0.4191 n/a
bibtex birds CAL500 Corel5k	$\begin{array}{c} n/a \\ \underline{0.3998} \ (60\%) \\ \hline n/a \\ 0.9028 \ (90\%) \end{array}$	n/a 0.4363 (90%) n/a 0.9061 (20%)	$rac{\mathrm{n/a}}{0.4162\ (10\%)}$ $\mathrm{n/a}$ $0.9215\ (30\%)$	${n/a} \ 0.4209 \ (10\%) \ n/a \ 0.9137 \ (60\%)$	n/a 0.4191 n/a 0.9088
bibtex birds CAL500 Corel5k emotions	$\frac{\frac{\mathrm{n/a}}{\mathrm{0.3998~(60\%)}}}{\frac{\mathrm{n/a}}{\mathrm{n/a}}}$ $\frac{0.9028~(90\%)}{\mathrm{0.4327~(60\%)}}$	n/a 0.4363 (90%) n/a 0.9061 (20%) 0.4406 (80%)	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%)	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%)	$\begin{array}{c} {\rm n/a}\\ {\rm 0.4191}\\ {\rm n/a}\\ {\rm 0.9088}\\ {\rm 0.4405} \end{array}$
bibtex birds CAL500 Corel5k emotions enron	$\begin{array}{c} n/a \\ \underline{0.3998\ (60\%)} \\ n/a \\ \underline{0.9028\ (90\%)} \\ \underline{0.4327\ (60\%)} \\ \overline{0.5803\ (10\%)} \end{array}$	n/a 0.4363 (90%) n/a 0.9061 (20%) 0.4406 (80%) 0.6400 (10%)	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%)	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%)	$\begin{array}{c} {\rm n/a}\\ {\rm 0.4191}\\ {\rm n/a}\\ {\rm 0.9088}\\ {\rm 0.4405}\\ {\rm 0.6806} \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml	$\begin{array}{c} {\rm n/a}\\ {\color{red}\underline{0.3998}\ (60\%)}\\ {\rm n/a}\\ {\color{red}\underline{0.9028}\ (90\%)}\\ {\color{red}\underline{0.4327}\ (60\%)}\\ {\color{red}\underline{0.5803}\ (10\%)}\\ {\color{red}\underline{0.3740}\ (60\%)} \end{array}$	n/a 0.4363 (90%) n/a 0.9061 (20%) 0.4406 (80%) 0.6400 (10%) 0.3814 (70%)	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%) 0.3786 (60%)	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%)	$\begin{array}{c} {\rm n/a}\\ {\rm 0.4191}\\ {\rm n/a}\\ {\rm 0.9088}\\ {\rm 0.4405}\\ {\rm 0.6806}\\ {\rm 0.3803} \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase	$\begin{array}{c} {\rm n/a}\\ {\color{red}\underline{0.3998}\ (60\%)}\\ {\rm n/a}\\ {\color{red}\underline{0.9028}\ (90\%)}\\ {\color{red}\underline{0.4327}\ (60\%)}\\ {\color{red}\underline{0.5803}\ (10\%)}\\ {\color{red}\underline{0.3740}\ (60\%)}\\ {\color{red}\underline{0.0595}\ (10\%)} \end{array}$	n/a 0.4363 (90%) n/a 0.9061 (20%) 0.4406 (80%) 0.6400 (10%) 0.3814 (70%) 0.0595 (10%)	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%) 0.3786 (60%) 0.0595 (10%)	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%)	$\begin{array}{c} {\rm n/a}\\ {\rm 0.4191}\\ {\rm n/a}\\ {\rm 0.9088}\\ {\rm 0.4405}\\ {\rm 0.6806}\\ {\rm 0.3803}\\ {\rm 0.0595} \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical	$\begin{array}{c} {\rm n/a}\\ {\color{red}\underline{0.3998}\ (60\%)}\\ {\rm n/a}\\ {\color{red}\underline{0.9028}\ (90\%)}\\ {\color{red}\underline{0.4327}\ (60\%)}\\ {\color{red}\underline{0.5803}\ (10\%)}\\ {\color{red}\underline{0.3740}\ (60\%)}\\ {\color{red}\underline{0.0595}\ (10\%)}\\ {\color{red}\underline{0.3606}\ (10\%)}\\ \end{array}$	n/a 0.4363 (90%) n/a 0.9061 (20%) 0.4406 (80%) 0.6400 (10%) 0.3814 (70%) <u>0.0595 (10%)</u> 0.4023 (10%)	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%) 0.3786 (60%) 0.0595 (10%) 0.4051 (10%)	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3981 (10%)	$\begin{array}{r} n/a \\ 0.4191 \\ n/a \\ 0.9088 \\ 0.4405 \\ 0.6806 \\ 0.3803 \\ \underline{0.0595} \\ 0.4624 \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene	$\begin{array}{c} {\rm n/a}\\ \hline 0.3998~(60\%)\\ {\rm n/a}\\ \hline 0.9028~(90\%)\\ \hline 0.4327~(60\%)\\ \hline 0.5803~(10\%)\\ \hline 0.3740~(60\%)\\ \hline 0.0595~(10\%)\\ \hline 0.3606~(10\%)\\ \hline 0.3329~(90\%)\\ \end{array}$	n/a 0.4363 (90%) n/a 0.9061 (20%) 0.4406 (80%) 0.6400 (10%) 0.3814 (70%) 0.0595 (10%) 0.4023 (10%) 0.3319 (90%)	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%) 0.3786 (60%) 0.0595 (10%) 0.4051 (10%) 0.3338 (90%)	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3981 (10%) 0.3273 (70%)	$\begin{array}{c} n/a \\ 0.4191 \\ n/a \\ 0.9088 \\ 0.4405 \\ 0.6806 \\ 0.3803 \\ \underline{0.0595} \\ 0.4624 \\ 0.3244 \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast	$\begin{array}{c} {\rm n/a}\\ \hline 0.3998~(60\%)\\ {\rm n/a}\\ \hline 0.9028~(90\%)\\ \hline 0.4327~(60\%)\\ \hline 0.5803~(10\%)\\ \hline 0.3740~(60\%)\\ \hline 0.0595~(10\%)\\ \hline 0.3606~(10\%)\\ \hline 0.3329~(90\%)\\ \hline 0.4630~(80\%)\\ \end{array}$	n/a 0.4363 (90%) n/a 0.9061 (20%) 0.4406 (80%) 0.6400 (10%) 0.3814 (70%) 0.0595 (10%) 0.4023 (10%) 0.3319 (90%) 0.4625 (90%)	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%) 0.3786 (60%) 0.0595 (10%) 0.4051 (10%) 0.3338 (90%) 0.4671 (90%)	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3981 (10%) 0.3273 (70%) 0.4657 (80%)	$\begin{array}{r} n/a \\ 0.4191 \\ n/a \\ 0.9088 \\ 0.4405 \\ 0.6806 \\ 0.3803 \\ \underline{0.0595} \\ 0.4624 \\ \underline{0.3244} \\ 0.4651 \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO	$\begin{array}{c} n/a\\ \underline{0.3998} \ (60\%)\\ \overline{n/a}\\ \underline{0.9028} \ (90\%)\\ \overline{0.4327} \ (60\%)\\ \overline{0.5803} \ (10\%)\\ \overline{0.3740} \ (60\%)\\ \underline{0.0595} \ (10\%)\\ \overline{0.3606} \ (10\%)\\ \overline{0.3329} \ (90\%)\\ \overline{0.4630} \ (80\%)\\ \end{array}$	n/a 0.4363 (90%) n/a 0.9061 (20%) 0.4406 (80%) 0.6400 (10%) 0.3814 (70%) 0.0595 (10%) 0.4023 (10%) 0.3319 (90%) 0.4625 (90%)	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%) 0.3786 (60%) 0.0595 (10%) 0.4051 (10%) 0.3338 (90%) 0.4671 (90%)	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3981 (10%) 0.3273 (70%) 0.4657 (80%)	$\begin{array}{c} {\rm n/a} \\ {\rm 0.4191} \\ {\rm n/a} \\ {\rm 0.9088} \\ {\rm 0.4405} \\ {\rm 0.6806} \\ {\rm 0.3803} \\ \underline{{\rm 0.0595}} \\ {\rm 0.4624} \\ \underline{{\rm 0.3244}} \\ {\rm 0.4651} \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set	n/a 0.3998 (60%) n/a 0.9028 (90%) 0.4327 (60%) 0.5803 (10%) 0.3740 (60%) 0.0595 (10%) 0.3606 (10%) 0.329 (90%) 0.4630 (80%) SS BR+InfoGain	n/a 0.4363 (90%) n/a 0.9061 (20%) 0.4406 (80%) 0.6400 (10%) 0.3814 (70%) 0.0595 (10%) 0.4023 (10%) 0.3319 (90%) 0.4625 (90%) Copy+InfoG ain	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%) 0.3786 (60%) 0.0595 (10%) 0.4051 (10%) 0.3338 (90%) 0.4671 (90%) LP+InfoGain	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3981 (10%) 0.3273 (70%) 0.4657 (80%) MLInfoGain	n/a 0.4191 n/a 0.9088 0.4405 0.6806 0.3803 <u>0.0595</u> 0.4624 <u>0.3244</u> 0.4651 No Sel.
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex	$\begin{array}{c} n/a \\ \hline 0.3998 \ (60\%) \\ \hline n/a \\ \hline 0.9028 \ (90\%) \\ \hline 0.4327 \ (60\%) \\ \hline 0.5803 \ (10\%) \\ \hline 0.3740 \ (60\%) \\ \hline 0.0595 \ (10\%) \\ \hline 0.3606 \ (10\%) \\ \hline 0.329 \ (90\%) \\ \hline 0.4630 \ (80\%) \\ \hline \textbf{SS} \\ \hline \textbf{BR+InfoGain} \\ \hline n/a \end{array}$	$\begin{array}{c} n/a \\ 0.4363 (90\%) \\ n/a \\ \textbf{0.9061} (20\%) \\ 0.4406 (80\%) \\ \textbf{0.6400} (10\%) \\ 0.3814 (70\%) \\ \textbf{0.0595} (10\%) \\ \textbf{0.4023} (10\%) \\ \textbf{0.3319} (90\%) \\ \textbf{0.4625} (90\%) \\ \hline \end{array}$	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%) 0.3786 (60%) 0.0595 (10%) 0.4051 (10%) 0.3338 (90%) 0.4671 (90%) LP+InfoGain n/a	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3981 (10%) 0.3273 (70%) 0.4657 (80%) MLInfoGain n/a	n/a 0.4191 n/a 0.9088 0.4405 0.6806 0.3803 <u>0.0595</u> 0.4624 <u>0.3244</u> 0.4651 No Sel. n/a
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds	$\begin{array}{c} {\rm n/a}\\ {\color{red}\underline{0.3998}\ (60\%)}\\ {\rm n/a}\\ {\color{red}\underline{0.9028}\ (90\%)}\\ {\color{red}\underline{0.4327}\ (60\%)}\\ {\color{red}\underline{0.5803}\ (10\%)}\\ {\color{red}\underline{0.3740}\ (60\%)}\\ {\color{red}\underline{0.3740}\ (60\%)}\\ {\color{red}\underline{0.3606}\ (10\%)}\\ {\color{red}\underline{0.3229}\ (90\%)}\\ {\color{red}\underline{0.4630}\ (80\%)}\\ {\color{red}{\rm SS}}\\ \hline {\color{red}{\rm BR+InfoGain}}\\ {\color{red}{n/a}}\\ {\color{red}\underline{0.1179}\ (80\%)}\\ \end{array}}$	$\begin{array}{c} n/a \\ 0.4363 (90\%) \\ n/a \\ 0.9061 (20\%) \\ 0.4406 (80\%) \\ 0.6400 (10\%) \\ 0.3814 (70\%) \\ 0.0595 (10\%) \\ 0.4023 (10\%) \\ 0.3319 (90\%) \\ 0.4625 (90\%) \\ \hline \\ $	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%) 0.3786 (60%) 0.0595 (10%) 0.4051 (10%) 0.3338 (90%) 0.4671 (90%) LP+InfoGain n/a 0.1190 (50%)	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3981 (10%) 0.3273 (70%) 0.4657 (80%) MLInfoGain n/a 0.1193 (50%)	$\begin{array}{c} n/a \\ 0.4191 \\ n/a \\ 0.9088 \\ 0.4405 \\ 0.6806 \\ 0.3803 \\ \underline{0.0595} \\ 0.4624 \\ \underline{0.3244} \\ 0.4651 \\ \hline \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds CAL500	$\begin{array}{c} {n/a}\\ {\color{red} 0.3998~(60\%)}\\ {n/a}\\ {\color{red} 0.9028~(90\%)}\\ {\color{red} 0.4327~(60\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.3740~(60\%)}\\ {\color{red} 0.0595~(10\%)}\\ {\color{red} 0.3606~(10\%)}\\ {\color{red} 0.329~(90\%)}\\ {\color{red} 0.4630~(80\%)}\\ \hline {\color{red} SS}\\ \hline {\color{red} BR+InfoGain}\\ {\color{red} n/a}\\ {\color{red} 0.1179~(80\%)}\\ {\color{red} n/a}\\ \hline {\color{red} n/a}\\ \hline \end{array}$	$\begin{array}{c} {n/a}\\ 0.4363 \ (90\%)\\ {n/a}\\ 0.9061 \ (20\%)\\ 0.4406 \ (80\%)\\ 0.6400 \ (10\%)\\ 0.3814 \ (70\%)\\ \hline 0.0595 \ (10\%)\\ \hline 0.4023 \ (10\%)\\ 0.3319 \ (90\%)\\ \hline 0.4625 \ (90\%)\\ \hline \hline \hline \\ \hline$	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%) 0.3786 (60%) 0.0595 (10%) 0.4051 (10%) 0.3338 (90%) 0.4671 (90%) LP+InfoGain n/a 0.1190 (50%) n/a	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3981 (10%) 0.3273 (70%) 0.4657 (80%) MLInfoGain n/a 0.1193 (50%) n/a	$\begin{array}{c} {n/a}\\ {0.4191}\\ {n/a}\\ {0.9088}\\ {0.4405}\\ {0.6806}\\ {0.3803}\\ \underline{{0.0595}}\\ {0.4624}\\ \underline{{0.3244}}\\ {0.4651}\\ \hline \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds CAL500 Corel5k	$\begin{array}{c} {n/a}\\ {\color{red} 0.3998~(60\%)}\\ {n/a}\\ {\color{red} 0.9028~(90\%)}\\ {\color{red} 0.4327~(60\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.3740~(60\%)}\\ {\color{red} 0.3740~(60\%)}\\ {\color{red} 0.3740~(60\%)}\\ {\color{red} 0.3740~(60\%)}\\ {\color{red} 0.3329~(90\%)}\\ {\color{red} 0.4630~(80\%)}\\ {\color{red} SS}\\ \hline {\color{red} BR+InfoGain}\\ {\color{red} n/a}\\ {\color{red} 0.1179~(80\%)}\\ {\color{red} n/a}\\ {\color{red} 0.1974~(20\%)}\\ \end{array}}$	n/a 0.4363 (90%) n/a 0.9061 (20%) 0.4406 (80%) 0.6400 (10%) 0.3814 (70%) 0.0595 (10%) 0.4023 (10%) 0.3319 (90%) 0.4625 (90%) Copy+InfoGain n/a 0.1200 (50%) n/a 0.1978 (20%)	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%) 0.3786 (60%) 0.0595 (10%) 0.4051 (10%) 0.3338 (90%) 0.4671 (90%) LP+InfoGain n/a 0.1190 (50%) n/a 0.2001 (20%)	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3981 (10%) 0.3273 (70%) 0.4657 (80%) MLInfoGain n/a 0.1193 (50%) n/a 0.1985 (10%)	$\begin{array}{c} {n/a}\\ {n/a}\\ {0.4191}\\ {n/a}\\ {0.9088}\\ {0.4405}\\ {0.6806}\\ {0.3803}\\ \underline{{0.0595}}\\ {0.4624}\\ \underline{{0.3244}}\\ {0.4651}\\ \hline \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds CAL500 Corel5k emotions	$\begin{array}{c} {n/a}\\ {\color{red} 0.3998~(60\%)}\\ {\color{red} n/a}\\ {\color{red} 0.9028~(90\%)}\\ {\color{red} 0.4327~(60\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.3740~(60\%)}\\ {\color{red} 0.0595~(10\%)}\\ {\color{red} 0.3606~(10\%)}\\ {\color{red} 0.329~(90\%)}\\ {\color{red} 0.4630~(80\%)}\\ \hline {\color{red} SS}\\ \hline {\color{red} BR+InfoGain}\\ {\color{red} n/a}\\ {\color{red} 0.1179~(80\%)}\\ {\color{red} n/a}\\ {\color{red} 0.1974~(20\%)}\\ {\color{red} 0.1556~(90\%)}\\ \hline \end{array}}$	$\begin{array}{c} {\rm n/a}\\ 0.4363\ (90\%)\\ {\rm n/a}\\ 0.9061\ (20\%)\\ 0.6400\ (10\%)\\ 0.6400\ (10\%)\\ 0.3814\ (70\%)\\ \hline 0.0595\ (10\%)\\ 0.4023\ (10\%)\\ 0.4023\ (10\%)\\ 0.3319\ (90\%)\\ \hline 0.4625\ (90\%)\\ \hline \hline \\ \hline \\$	n/a 0.4162 (10%) n/a 0.9215 (30%) 0.4361 (90%) 0.5929 (10%) 0.3786 (60%) 0.0595 (10%) 0.4051 (10%) 0.3338 (90%) 0.4671 (90%) LP+InfoGain n/a 0.1190 (50%) n/a 0.2001 (20%) 0.1600 (90%)	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3273 (70%) 0.4657 (80%) MLInfoGain n/a 0.1193 (50%) n/a 0.1985 (10%) 0.1543 (90%)	$\begin{array}{c} {n/a}\\ {n/a}\\ {0.4191}\\ {n/a}\\ {0.9088}\\ {0.4405}\\ {0.6806}\\ {0.3803}\\ \underline{{0.0595}}\\ {0.4624}\\ \underline{{0.3244}}\\ {0.4651}\\ \hline \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds CAL500 Corel5k emotions enron	$\begin{array}{c} {n/a}\\ {\color{red} 0.3998\ (60\%)}\\ {\color{red} n/a}\\ {\color{red} 0.9028\ (90\%)}\\ {\color{red} 0.4327\ (60\%)}\\ {\color{red} 0.5803\ (10\%)}\\ {\color{red} 0.5803\ (10\%)}\\ {\color{red} 0.3740\ (60\%)}\\ {\color{red} 0.0595\ (10\%)}\\ {\color{red} 0.3606\ (10\%)}\\ {\color{red} 0.329\ (90\%)}\\ {\color{red} 0.4630\ (80\%)}\\ \hline {\color{red} SS}\\ \hline {\color{red} BR+InfoGain}\\ {\color{red} n/a}\\ {\color{red} 0.1179\ (80\%)}\\ {\color{red} n/a}\\ {\color{red} 0.1974\ (20\%)}\\ {\color{red} 0.1556\ (90\%)}\\ {\color{red} 0.1089\ (10\%)}\\ \hline \end{array}$	$\begin{array}{c} {\rm n/a}\\ 0.4363\ (90\%)\\ {\rm n/a}\\ 0.9061\ (20\%)\\ 0.6400\ (10\%)\\ 0.6400\ (10\%)\\ 0.3814\ (70\%)\\ \hline 0.0595\ (10\%)\\ \hline 0.4023\ (10\%)\\ 0.4023\ (10\%)\\ \hline 0.4023\ (10\%)\\ \hline 0.4625\ (90\%)\\ \hline \hline \\ \hline \\$	$\begin{array}{c} {\rm n/a} \\ {\rm 0.4162~(10\%)} \\ {\rm n/a} \\ {\rm 0.9215~(30\%)} \\ {\rm 0.4361~(90\%)} \\ {\rm 0.5929~(10\%)} \\ {\rm 0.3786~(60\%)} \\ {\rm 0.0595~(10\%)} \\ {\rm 0.4051~(10\%)} \\ {\rm 0.4051~(10\%)} \\ {\rm 0.3338~(90\%)} \\ {\rm 0.4671~(90\%)} \\ \hline \\ {\rm LP+InfoGain} \\ {\rm n/a} \\ {\rm 0.1190~(50\%)} \\ {\rm n/a} \\ {\rm 0.2001~(20\%)} \\ {\rm 0.1600~(90\%)} \\ {\rm 0.1120~(10\%)} \end{array}$	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3981 (10%) 0.3273 (70%) 0.4657 (80%) MLInfoGain n/a 0.1193 (50%) n/a 0.1985 (10%) 0.1543 (90%) 0.1124 (10%)	$\begin{array}{c} {n/a}\\ {n/a}\\ {0.4191}\\ {n/a}\\ {0.9088}\\ {0.4405}\\ {0.6806}\\ {0.3803}\\ \underline{{0.0595}}\\ {0.4624}\\ \underline{{0.3244}}\\ {0.4651}\\ \hline \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds CAL500 Corel5k emotions enron flagsml	$\begin{array}{c} {n/a}\\ {\color{red} 0.3998~(60\%)}\\ {\color{red} n/a}\\ {\color{red} 0.9028~(90\%)}\\ {\color{red} 0.4327~(60\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.595~(10\%)}\\ {\color{red} 0.3740~(60\%)}\\ {\color{red} 0.3740~(60\%)}\\ {\color{red} 0.3740~(60\%)}\\ {\color{red} 0.3740~(60\%)}\\ {\color{red} 0.3329~(90\%)}\\ {\color{red} 0.4630~(80\%)}\\ \hline {\color{red} 0.3329~(90\%)}\\ {\color{red} 0.4630~(80\%)}\\ {\color{red} 0.3329~(90\%)}\\ {\color{red} 0.1179~(80\%)}\\ {\color{red} n/a}\\ {\color{red} 0.1179~(80\%)}\\ {\color{red} 0.1974~(20\%)}\\ {\color{red} 0.1556~(90\%)}\\ {\color{red} 0.1089~(10\%)}\\ {\color{red} 0.1818~(30\%)}\\ \hline \end{array}}$	$\begin{array}{c} {\rm n/a}\\ 0.4363\ (90\%)\\ {\rm n/a}\\ 0.9061\ (20\%)\\ 0.4406\ (80\%)\\ 0.6400\ (10\%)\\ 0.3814\ (70\%)\\ \hline 0.0595\ (10\%)\\ \hline 0.4023\ (10\%)\\ 0.4023\ (10\%)\\ \hline 0.4023\ (10\%)\\ \hline 0.4625\ (90\%)\\ \hline 0.4625\ (90\%)\\ \hline 0.1200\ (50\%)\\ {\rm n/a}\\ 0.1978\ (20\%)\\ 0.1552\ (90\%)\\ \hline 0.1129\ (10\%)\\ 0.1820\ (20\%)\\ \hline \end{array}$	$\begin{array}{c} {\rm n/a} \\ {\rm 0.4162~(10\%)} \\ {\rm n/a} \\ {\rm 0.9215~(30\%)} \\ {\rm 0.4361~(90\%)} \\ {\rm 0.5929~(10\%)} \\ {\rm 0.3786~(60\%)} \\ {\rm 0.0595~(10\%)} \\ {\rm 0.4051~(10\%)} \\ {\rm 0.4051~(10\%)} \\ {\rm 0.3338~(90\%)} \\ {\rm 0.4671~(90\%)} \\ \hline \\ {\rm LP+InfoGain} \\ {\rm n/a} \\ {\rm 0.1190~(50\%)} \\ {\rm n/a} \\ {\rm 0.2001~(20\%)} \\ {\rm 0.1600~(90\%)} \\ {\rm 0.1120~(10\%)} \\ {\rm 0.1840~(30\%)} \\ \end{array}$	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3981 (10%) 0.3273 (70%) 0.4657 (80%) MLInfoGain n/a 0.1193 (50%) n/a 0.1985 (10%) 0.1543 (90%) 0.1124 (10%) 0.1775 (50%)	$\begin{array}{c} {n/a}\\ {n/a}\\ {0.4191}\\ {n/a}\\ {0.9088}\\ {0.4405}\\ {0.6806}\\ {0.3803}\\ \underline{{0.0595}}\\ {0.4624}\\ \underline{{0.3244}}\\ {0.4651}\\ \hline \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds CAL500 Corel5k emotions enron flagsml genbase	$\begin{array}{c} {\rm n/a}\\ {\color{red} 0.3998~(60\%)}\\ {\rm n/a}\\ {\color{red} 0.9028~(90\%)}\\ {\color{red} 0.4327~(60\%)}\\ {\color{red} 0.4327~(60\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.3740~(60\%)}\\ {\color{red} 0.0595~(10\%)}\\ {\color{red} 0.3606~(10\%)}\\ {\color{red} 0.329~(90\%)}\\ {\color{red} 0.4630~(80\%)}\\ {\color{red} 0.3329~(90\%)}\\ {\color{red} 0.4630~(80\%)}\\ {\color{red} 0.3329~(90\%)}\\ {\color{red} 0.4630~(80\%)}\\ {\color{red} 0.1179~(80\%)}\\ {\color{red} n/a}\\ {\color{red} 0.1179~(80\%)}\\ {\color{red} 0.1974~(20\%)}\\ {\color{red} 0.1556~(90\%)}\\ {\color{red} 0.1089~(10\%)}\\ {\color{red} 0.1231~(10\%)}\\ {\color{red} 0.231~(10\%)} \end{array}$	$\begin{array}{c} {\rm n/a}\\ 0.4363\ (90\%)\\ {\rm n/a}\\ 0.9061\ (20\%)\\ 0.4406\ (80\%)\\ 0.6400\ (10\%)\\ 0.3814\ (70\%)\\ \hline 0.0595\ (10\%)\\ \hline 0.4023\ (10\%)\\ 0.4023\ (10\%)\\ \hline 0.4023\ (10\%)\\ \hline 0.4625\ (90\%)\\ \hline 0.4625\ (90\%)\\ \hline 0.1200\ (50\%)\\ {\rm n/a}\\ 0.1978\ (20\%)\\ 0.1552\ (90\%)\\ \hline 0.1129\ (10\%)\\ 0.1820\ (20\%)\\ \hline 0.0231\ (10\%)\\ \hline \end{array}$	$\begin{array}{c} {\rm n/a} \\ {\rm 0.4162} \ (10\%) \\ {\rm n/a} \\ {\rm 0.9215} \ (30\%) \\ {\rm 0.4361} \ (90\%) \\ {\rm 0.5929} \ (10\%) \\ {\rm 0.3786} \ (60\%) \\ {\rm 0.0595} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4671} \ (90\%) \\ \hline \end{array}$	n/a 0.4209 (10%) n/a 0.9137 (60%) 0.4348 (60%) 0.6398 (60%) 0.3734 (90%) 0.0595 (10%) 0.3981 (10%) 0.3273 (70%) 0.4657 (80%) MLInfoGain n/a 0.1193 (50%) n/a 0.1985 (10%) 0.1543 (90%) 0.1124 (10%) 0.231 (10%)	$\begin{array}{c} {n/a}\\ {n/a}\\ {0.4191}\\ {n/a}\\ {0.9088}\\ {0.4405}\\ {0.6806}\\ {0.3803}\\ \underline{{0.0595}}\\ {0.4624}\\ \underline{{0.3244}}\\ {0.4651}\\ \hline \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical	$\begin{array}{c} {\rm n/a}\\ {\color{red} 0.3998~(60\%)}\\ {\rm n/a}\\ {\color{red} 0.9028~(90\%)}\\ {\color{red} 0.4327~(60\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.5803~(10\%)}\\ {\color{red} 0.3740~(60\%)}\\ {\color{red} 0.0595~(10\%)}\\ {\color{red} 0.3606~(10\%)}\\ {\color{red} 0.329~(90\%)}\\ {\color{red} 0.4630~(80\%)}\\ {\color{red} SS}\\ \hline {\color{red} {\bf BR+InfoGain}}\\ {\color{red} n/a}\\ {\color{red} 0.1179~(80\%)}\\ {\color{red} n/a}\\ {\color{red} 0.1974~(20\%)}\\ {\color{red} 0.1556~(90\%)}\\ {\color{red} 0.1089~(10\%)}\\ {\color{red} 0.1818~(30\%)}\\ {\color{red} 0.0231~(10\%)}\\ {\color{red} 0.0424~(10\%)} \end{array}}$	$\begin{array}{c} {\rm n/a}\\ 0.4363\ (90\%)\\ {\rm n/a}\\ 0.9061\ (20\%)\\ 0.6400\ (10\%)\\ 0.6400\ (10\%)\\ 0.3814\ (70\%)\\ \hline 0.0595\ (10\%)\\ \hline 0.4023\ (10\%)\\ 0.4023\ (10\%)\\ \hline 0.4023\ (10\%)\\ \hline 0.4625\ (90\%)\\ \hline 0.4625\ (90\%)\\ \hline 0.1200\ (50\%)\\ {\rm n/a}\\ 0.1978\ (20\%)\\ 0.1552\ (90\%)\\ 0.1552\ (90\%)\\ 0.1129\ (10\%)\\ 0.0231\ (10\%)\\ \hline 0.0470\ (10\%)\\ \hline \end{array}$	$\begin{array}{c} {\rm n/a} \\ {\rm 0.4162} \ (10\%) \\ {\rm n/a} \\ {\rm 0.9215} \ (30\%) \\ {\rm 0.4361} \ (90\%) \\ {\rm 0.5929} \ (10\%) \\ {\rm 0.3786} \ (60\%) \\ {\rm 0.0595} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4671} \ (90\%) \\ \hline \end{array}$	$\begin{array}{r} {n/a}\\ 0.4209~(10\%)\\ {n/a}\\ 0.9137~(60\%)\\ 0.4348~(60\%)\\ 0.6398~(60\%)\\ 0.3734~(90\%)\\ \hline 0.0595~(10\%)\\ 0.3981~(10\%)\\ 0.3981~(10\%)\\ 0.3273~(70\%)\\ 0.4657~(80\%)\\ \hline \\ \hline \\ {\rm MLInfoGain}\\ {n/a}\\ 0.1193~(50\%)\\ {n/a}\\ 0.1985~(10\%)\\ 0.1543~(90\%)\\ \hline 0.1124~(10\%)\\ 0.1775~(50\%)\\ \hline 0.0231~(10\%)\\ 0.0465~(10\%)\\ \hline \end{array}$	$\begin{array}{c} {n/a}\\ {n/a}\\ {0.4191}\\ {n/a}\\ {0.9088}\\ {0.4405}\\ {0.6806}\\ {0.3803}\\ \underline{{0.0595}}\\ {0.4624}\\ \underline{{0.3244}}\\ {0.4651}\\ \hline \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene	$\begin{array}{r} {\rm n/a} \\ \hline 0.3998 \ (60\%) \\ {\rm n/a} \\ \hline 0.9028 \ (90\%) \\ \hline 0.4327 \ (60\%) \\ \hline 0.5803 \ (10\%) \\ \hline 0.5803 \ (10\%) \\ \hline 0.3740 \ (60\%) \\ \hline 0.0595 \ (10\%) \\ \hline 0.3606 \ (10\%) \\ \hline 0.329 \ (90\%) \\ \hline 0.4630 \ (80\%) \\ \hline {\rm SS} \\ \hline {\rm BR+InfoGain} \\ {\rm n/a} \\ \hline 0.1179 \ (80\%) \\ \hline {\rm n/a} \\ \hline 0.1974 \ (20\%) \\ \hline 0.1556 \ (90\%) \\ \hline 0.1089 \ (10\%) \\ \hline 0.1818 \ (30\%) \\ \hline 0.0231 \ (10\%) \\ \hline 0.0912 \ (90\%) \\ \hline \end{array}$	$\begin{array}{c} {\rm n/a}\\ 0.4363\ (90\%)\\ {\rm n/a}\\ 0.9061\ (20\%)\\ 0.4406\ (80\%)\\ 0.6400\ (10\%)\\ 0.3814\ (70\%)\\ \hline 0.0595\ (10\%)\\ \hline 0.4023\ (10\%)\\ 0.4023\ (10\%)\\ \hline 0.4023\ (10\%)\\ \hline 0.4625\ (90\%)\\ \hline 0.4625\ (90\%)\\ \hline 0.4625\ (90\%)\\ \hline 0.1200\ (50\%)\\ {\rm n/a}\\ 0.1200\ (50\%)\\ {\rm n/a}\\ 0.1978\ (20\%)\\ 0.1552\ (90\%)\\ \hline 0.1552\ (90\%)\\ 0.1129\ (10\%)\\ \hline 0.0231\ (10\%)\\ \hline 0.0470\ (10\%)\\ \hline 0.0888\ (90\%)\\ \end{array}$	$\begin{array}{c} {\rm n/a} \\ {\rm 0.4162} \ (10\%) \\ {\rm n/a} \\ {\rm 0.9215} \ (30\%) \\ {\rm 0.4361} \ (90\%) \\ {\rm 0.5929} \ (10\%) \\ {\rm 0.3786} \ (60\%) \\ {\rm 0.0595} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4671} \ (90\%) \\ \hline \end{array}$	$\begin{array}{r} {n/a}\\ 0.4209\ (10\%)\\ {n/a}\\ 0.9137\ (60\%)\\ 0.4348\ (60\%)\\ 0.6398\ (60\%)\\ 0.3734\ (90\%)\\ \hline 0.0595\ (10\%)\\ 0.3981\ (10\%)\\ 0.3981\ (10\%)\\ 0.3273\ (70\%)\\ 0.4657\ (80\%)\\ \hline \\ \hline$	$\begin{array}{c} {\rm n/a} \\ {\rm 0.4191} \\ {\rm n/a} \\ {\rm 0.9088} \\ {\rm 0.4405} \\ {\rm 0.6806} \\ {\rm 0.3803} \\ \hline {\rm 0.0595} \\ {\rm 0.4624} \\ \hline {\rm 0.3244} \\ {\rm 0.4651} \\ \hline \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast	$\begin{array}{c} {\rm n/a}\\ \hline 0.3998 \ (60\%)\\ \hline {\rm n/a}\\ \hline 0.9028 \ (90\%)\\ \hline 0.4327 \ (60\%)\\ \hline 0.5803 \ (10\%)\\ \hline 0.5803 \ (10\%)\\ \hline 0.3740 \ (60\%)\\ \hline 0.0595 \ (10\%)\\ \hline 0.3606 \ (10\%)\\ \hline 0.329 \ (90\%)\\ \hline 0.4630 \ (80\%)\\ \hline {\rm SS}\\ \hline {\rm BR+InfoGain}\\ \hline {\rm n/a}\\ \hline 0.1179 \ (80\%)\\ \hline {\rm n/a}\\ \hline 0.1974 \ (20\%)\\ \hline 0.1556 \ (90\%)\\ \hline 0.1089 \ (10\%)\\ \hline 0.1818 \ (30\%)\\ \hline 0.0231 \ (10\%)\\ \hline 0.0912 \ (90\%)\\ \hline 0.1673 \ (80\%)\\ \hline \end{array}$	$\begin{array}{c} {\rm n/a}\\ 0.4363\ (90\%)\\ {\rm n/a}\\ 0.9061\ (20\%)\\ 0.4406\ (80\%)\\ 0.6400\ (10\%)\\ 0.3814\ (70\%)\\ \hline 0.0595\ (10\%)\\ \hline 0.4023\ (10\%)\\ 0.4023\ (10\%)\\ \hline 0.4023\ (10\%)\\ \hline 0.4625\ (90\%)\\ \hline 0.4625\ (90\%)\\ \hline 0.4625\ (90\%)\\ \hline 0.1200\ (50\%)\\ {\rm n/a}\\ 0.1200\ (50\%)\\ {\rm n/a}\\ 0.1978\ (20\%)\\ 0.1552\ (90\%)\\ \hline 0.1552\ (90\%)\\ \hline 0.1820\ (20\%)\\ \hline 0.0231\ (10\%)\\ \hline 0.0888\ (90\%)\\ \hline 0.1653\ (90\%)\\ \hline \end{array}$	$\begin{array}{c} {\rm n/a} \\ {\rm 0.4162} \ (10\%) \\ {\rm n/a} \\ {\rm 0.9215} \ (30\%) \\ {\rm 0.4361} \ (90\%) \\ {\rm 0.5929} \ (10\%) \\ {\rm 0.3786} \ (60\%) \\ {\rm 0.0595} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4671} \ (90\%) \\ \hline \end{array}$	$\begin{array}{r} {n/a}\\ 0.4209\ (10\%)\\ {n/a}\\ 0.9137\ (60\%)\\ 0.4348\ (60\%)\\ 0.6398\ (60\%)\\ 0.3734\ (90\%)\\ \hline 0.0595\ (10\%)\\ 0.3981\ (10\%)\\ 0.3981\ (10\%)\\ 0.3273\ (70\%)\\ 0.4657\ (80\%)\\ \hline \end{array}$	$\begin{array}{c} {n/a}\\ {n/a}\\ {0.4191}\\ {n/a}\\ {0.9088}\\ {0.4405}\\ {0.6806}\\ {0.3803}\\ \underline{{0.0595}}\\ {0.4624}\\ \underline{{0.3244}}\\ {0.4651}\\ \hline \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast Best values	$\begin{array}{c} {\rm n/a}\\ \hline 0.3998 \ (60\%)\\ {\rm n/a}\\ \hline 0.9028 \ (90\%)\\ \hline 0.4327 \ (60\%)\\ \hline 0.5803 \ (10\%)\\ \hline 0.5803 \ (10\%)\\ \hline 0.3740 \ (60\%)\\ \hline 0.0595 \ (10\%)\\ \hline 0.3606 \ (10\%)\\ \hline 0.329 \ (90\%)\\ \hline 0.4630 \ (80\%)\\ \hline {\rm SS}\\ \hline {\rm BR+InfoGain}\\ {\rm n/a}\\ \hline 0.1179 \ (80\%)\\ \hline {\rm n/a}\\ \hline 0.1974 \ (20\%)\\ \hline 0.1556 \ (90\%)\\ \hline 0.1089 \ (10\%)\\ \hline 0.1818 \ (30\%)\\ \hline 0.0231 \ (10\%)\\ \hline 0.0912 \ (90\%)\\ \hline 0.1673 \ (80\%)\\ \hline \end{array}$	$\begin{array}{c} n/a \\ 0.4363 (90\%) \\ n/a \\ 0.9061 (20\%) \\ 0.4406 (80\%) \\ 0.6400 (10\%) \\ 0.3814 (70\%) \\ 0.0595 (10\%) \\ 0.4023 (10\%) \\ 0.4023 (10\%) \\ 0.4023 (10\%) \\ 0.4625 (90\%) \\ \hline \\ $	$\begin{array}{c} {\rm n/a} \\ {\rm 0.4162} \ (10\%) \\ {\rm n/a} \\ {\rm 0.9215} \ (30\%) \\ {\rm 0.4361} \ (90\%) \\ {\rm 0.5929} \ (10\%) \\ {\rm 0.3786} \ (60\%) \\ {\rm 0.0595} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4051} \ (10\%) \\ {\rm 0.4671} \ (90\%) \\ \hline \end{array}$	$\begin{array}{c} {\rm n/a} \\ 0.4209 \ (10\%) \\ {\rm n/a} \\ 0.9137 \ (60\%) \\ 0.4348 \ (60\%) \\ 0.6398 \ (60\%) \\ 0.3734 \ (90\%) \\ \hline 0.0595 \ (10\%) \\ 0.3981 \ (10\%) \\ 0.3981 \ (10\%) \\ 0.3273 \ (70\%) \\ 0.4657 \ (80\%) \\ \hline \end{array}$	$\begin{array}{c} n/a \\ 0.4191 \\ n/a \\ 0.9088 \\ 0.4405 \\ 0.6806 \\ 0.3803 \\ \underline{0.0595} \\ 0.4624 \\ \underline{0.3244} \\ 0.4651 \\ \hline \end{array}$ $\begin{array}{c} No \ Sel. \\ n/a \\ 0.1197 \\ n/a \\ 0.2015 \\ 0.1615 \\ 0.1235 \\ 0.1615 \\ 0.1235 \\ 0.1867 \\ \underline{0.0231} \\ 0.0506 \\ \underline{0.0884} \\ 0.1661 \\ \hline \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast Best values (underlined)	$\begin{array}{c} {\rm n/a}\\ \hline 0.3998 \ (60\%)\\ {\rm n/a}\\ \hline 0.9028 \ (90\%)\\ \hline 0.4327 \ (60\%)\\ \hline 0.5803 \ (10\%)\\ \hline 0.5803 \ (10\%)\\ \hline 0.3740 \ (60\%)\\ \hline 0.0595 \ (10\%)\\ \hline 0.3606 \ (10\%)\\ \hline 0.329 \ (90\%)\\ \hline 0.4630 \ (80\%)\\ \hline {\rm SS}\\ \hline {\rm BR+InfoGain}\\ {\rm n/a}\\ \hline 0.1179 \ (80\%)\\ \hline {\rm n/a}\\ \hline 0.1974 \ (20\%)\\ \hline 0.1556 \ (90\%)\\ \hline 0.1089 \ (10\%)\\ \hline 0.1818 \ (30\%)\\ \hline 0.0912 \ (90\%)\\ \hline 0.1673 \ (80\%)\\ \hline \hline 22 \\ \end{array}$	$\begin{array}{r} n/a \\ 0.4363 (90\%) \\ n/a \\ 0.9061 (20\%) \\ 0.4406 (80\%) \\ 0.6400 (10\%) \\ 0.3814 (70\%) \\ 0.0595 (10\%) \\ 0.4023 (10\%) \\ 0.4023 (10\%) \\ 0.4023 (10\%) \\ 0.4625 (90\%) \\ \hline \end{array}$	$\begin{array}{c} {\rm n/a} \\ {\rm 0.4162~(10\%)} \\ {\rm n/a} \\ {\rm 0.9215~(30\%)} \\ {\rm 0.4361~(90\%)} \\ {\rm 0.5929~(10\%)} \\ {\rm 0.5929~(10\%)} \\ {\rm 0.3786~(60\%)} \\ {\rm 0.0595~(10\%)} \\ {\rm 0.4051~(10\%)} \\ {\rm 0.4051~(10\%)} \\ {\rm 0.4051~(10\%)} \\ {\rm 0.4671~(90\%)} \\ \hline \\ {\rm LP+InfoGain} \\ {\rm n/a} \\ {\rm 0.1190~(50\%)} \\ {\rm n/a} \\ {\rm 0.2001~(20\%)} \\ {\rm 0.1600~(90\%)} \\ {\rm 0.1120~(10\%)} \\ {\rm 0.0231~(10\%)} \\ {\rm 0.0911~(90\%)} \\ {\rm 0.1690~(90\%)} \\ \hline \\ {\rm 0.1690~(90\%)} \\ \hline \end{array}$	$\begin{array}{r} n/a \\ 0.4209 \ (10\%) \\ n/a \\ 0.9137 \ (60\%) \\ 0.4348 \ (60\%) \\ 0.6398 \ (60\%) \\ 0.3734 \ (90\%) \\ \hline 0.0595 \ (10\%) \\ 0.3981 \ (10\%) \\ 0.3273 \ (70\%) \\ 0.4657 \ (80\%) \\ \hline \end{array}$ $\begin{array}{r} \textbf{MLInfoGain} \\ n/a \\ 0.1193 \ (50\%) \\ n/a \\ 0.1985 \ (10\%) \\ 0.1543 \ (90\%) \\ \hline 0.1124 \ (10\%) \\ 0.1775 \ (50\%) \\ \hline 0.0231 \ (10\%) \\ 0.0889 \ (90\%) \\ 0.1668 \ (80\%) \\ \hline \end{array}$	$\begin{array}{c} {n/a}\\ {n/a}\\ {0.4191}\\ {n/a}\\ {0.9088}\\ {0.4405}\\ {0.6806}\\ {0.3803}\\ \underline{{0.0595}}\\ {0.4624}\\ \underline{{0.3244}}\\ {0.4651}\\ \hline \end{array}$
bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast RANKING LO Data Set bibtex birds CAL500 Corel5k emotions enron flagsml genbase medical scene yeast Best values (underlined) \leq baseline score	$\begin{array}{c} {\rm n/a}\\ {\color{red} 0.3998\ (60\%)}\\ {\rm n/a}\\ {\color{red} 0.9028\ (90\%)}\\ {\color{red} 0.4327\ (60\%)}\\ {\color{red} 0.5803\ (10\%)}\\ {\color{red} 0.5803\ (10\%)}\\ {\color{red} 0.3740\ (60\%)}\\ {\color{red} 0.0595\ (10\%)}\\ {\color{red} 0.3606\ (10\%)}\\ {\color{red} 0.329\ (90\%)}\\ {\color{red} 0.4630\ (80\%)}\\ \hline {\color{red} SS}\\ \hline {\color{red} {\bf BR+InfoGain}}\\ {\color{red} n/a}\\ {\color{red} 0.1179\ (80\%)}\\ {\color{red} 0.1179\ (80\%)}\\ {\color{red} 0.1974\ (20\%)}\\ {\color{red} 0.1556\ (90\%)}\\ {\color{red} 0.1089\ (10\%)}\\ {\color{red} 0.0231\ (10\%)}\\ {\color{red} 0.0912\ (90\%)}\\ {\color{red} 0.1673\ (80\%)}\\ \hline \end{array}}$	$\begin{array}{r} n/a \\ 0.4363 (90\%) \\ n/a \\ 0.9061 (20\%) \\ 0.4406 (80\%) \\ 0.6400 (10\%) \\ 0.3814 (70\%) \\ 0.0595 (10\%) \\ 0.4023 (10\%) \\ 0.4023 (10\%) \\ 0.4023 (10\%) \\ 0.4023 (10\%) \\ 0.4625 (90\%) \\ \hline 0.4625 (90\%) \\ 0.4625 (90\%) \\ 0.14625 (90\%) \\ 0.1978 (20\%) \\ 0.1978 (20\%) \\ 0.1552 (90\%) \\ 0.1552 (90\%) \\ 0.1552 (90\%) \\ 0.1552 (90\%) \\ 0.1129 (10\%) \\ 0.0888 (90\%) \\ 0.1653 (90\%) \\ \hline 8 \\ 25 \\ \end{array}$	$\begin{array}{c} {\rm n/a} \\ {\rm 0.4162~(10\%)} \\ {\rm n/a} \\ {\rm 0.9215~(30\%)} \\ {\rm 0.4361~(90\%)} \\ {\rm 0.5929~(10\%)} \\ {\rm 0.5929~(10\%)} \\ {\rm 0.3786~(60\%)} \\ {\rm 0.0595~(10\%)} \\ {\rm 0.4051~(10\%)} \\ {\rm 0.4051~(10\%)} \\ {\rm 0.4051~(10\%)} \\ {\rm 0.4671~(90\%)} \\ \hline \\ {\rm LP+InfoGain} \\ {\rm n/a} \\ {\rm 0.1190~(50\%)} \\ {\rm n/a} \\ {\rm 0.2001~(20\%)} \\ {\rm 0.1600~(90\%)} \\ {\rm 0.1120~(10\%)} \\ {\rm 0.0231~(10\%)} \\ {\rm 0.0911~(90\%)} \\ {\rm 0.1690~(90\%)} \\ \hline \\ 5 \\ \hline \end{array}$	$\begin{array}{c} {\rm n/a}\\ 0.4209~(10\%)\\ {\rm n/a}\\ 0.9137~(60\%)\\ 0.4348~(60\%)\\ 0.6398~(60\%)\\ 0.3734~(90\%)\\ \hline 0.0595~(10\%)\\ 0.3981~(10\%)\\ 0.3981~(10\%)\\ 0.3273~(70\%)\\ 0.4657~(80\%)\\ \hline \\ {\rm MLInfoGain}\\ {\rm n/a}\\ 0.1193~(50\%)\\ {\rm n/a}\\ 0.1985~(10\%)\\ 0.1543~(90\%)\\ \hline 0.1124~(10\%)\\ 0.1775~(50\%)\\ \hline 0.0231~(10\%)\\ 0.0889~(90\%)\\ 0.1668~(80\%)\\ \hline \\ 10\\ \hline \end{array}$	$\begin{array}{c} n/a \\ 0.4191 \\ n/a \\ 0.9088 \\ 0.4405 \\ 0.6806 \\ 0.3803 \\ \underline{0.0595} \\ 0.4624 \\ \underline{0.3244} \\ 0.4651 \\ \hline \end{array}$

Table 7.2: Best results achieved with the PPT + K-NN classifier

HAMMING LO	DSS				
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	n/a	n/a	0.0140 (20%)	0.0130(10%)	0.0140
birds	0.0393 (10%)	0.0413 (50%)	0.0419 (50%)	0.0388 (10%)	0.0441
CAL500	0.1510 (10%)	0.1501(10%)	0.1525(10%)	0.1503(10%)	0.1684
Corel5 k	0.0094 (10%)	0.0095(10%)	0.0094(10%)	0.0095(10%)	0.0097
emotions	$\overline{0.2093(60\%)}$	0.2099(60%)	0.2131(80%)	0.2119(60%)	0.2178
enron	$\frac{1}{0.0472}$ (20%)	0.0485(90%)	0.0476(40%)	0.0483(80%)	0.0485
flagsml	$\overline{0.2400(90\%)}$	0.2348 (90%)	0.2348(90%)	0.2417(80%)	0.2414
genbase	0.0011 (10%)	$\frac{0.0011}{0.0011}$ (10%)	$\frac{0.0011}{0.0011}$ (10%)	0.0011 (10%)	0.0011
medical	$\frac{1}{0.0098}$ (20%)	$\frac{0.0100}{0.0100}$ (10%)	$\frac{0.0100}{0.0100}$ (10%)	$\frac{0.0100}{0.0100}$ (10%)	0.0102
scene	$\frac{1}{0.1008}$ (60%)	0.1021 (90%)	0.1010 (90%)	0.1007(70%)	0.1012
veast	0.2230(70%)	0.2237 (40%)	0.2286 (90%)	$\frac{0.2245}{0.2245}$ (30%)	0.2257
SUBSET 0/1 L	OSS				
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	n/a	n/a	0.8522 (60%)	0.8569(10%)	0.8534
birds	0.4448(10%)	0.4557 (40%)	$\frac{0.4680}{0.4680}$ (50%)	0.4494(20%)	0 4820
CAL500	$\frac{0.1110}{1.0000}$ (10%)	1,0000,(10%)	1,0000,(10%)	1,0000,(10%)	1 0000
Corel5k	$\frac{1.0000}{0.9974}$ (30%)	$\frac{100000}{0.9960}$ (90%)	$\frac{100000}{0.9982}$ (90%)	$\frac{1.0000}{0.9962}$ (80%)	$\frac{110000}{0.9976}$
emotions	0.7385 (60%)	$\frac{0.0000}{0.7303}$ (20%)	0.7318 (10%)	0.7334 (50%)	0.7522
enron	0.8672(20%)	$\frac{0.1000}{0.8802}$ (80%)	0.8784 (50%)	0.8737 (80%)	0.8843
flagsml	$\frac{0.0012}{0.7563}$ (90%)	0.7471 (90%)	0.7471 (90%)	0.7568 (90%)	0.7466
genbase	0.0287 (10%)	0.0287 (10%)	0.0287 (10%)	0.0287 (10%)	$\frac{0.1100}{0.0287}$
medical	$\frac{0.0201}{0.3201}$ (10%)	$\frac{0.0201}{0.3273}$ (10%)	$\frac{0.0261}{0.3262}$ (10%)	$\frac{0.0201}{0.3273}$ (10%)	$\frac{0.3231}{0.3416}$
scene	$\frac{0.0201}{0.4354}$ (80%)	0.4371 (50%)	0.4387 (80%)	0.4346 (70%)	0 4396
veast	0.8759 (60%)	0.8800(20%)	0.8870 (80%)	$\frac{0.1010}{0.8788}$ (70%)	0.8837
EXAMPLE BA	SED ACCURAC	CV (inverted)			0.0001
Data Set	BB+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	n/a	n/a	0.6944 (60%)	0.7211 (10%)	0.6893
birds	0.3578 (10%)	0.3643 (40%)	0.3779(50%)	0.3585(10%)	$\frac{0.3946}{0.3946}$
CAL500	$\overline{0.7649}(60\%)$	0.7646(30%)	0.7660 (70%)	0.7666 (90%)	0.7707
Corel5 k	0.9468 (90%)	0.9432(90%)	0.9487(90%)	0.9432(90%)	0.9425
emotions	0.4765 (60%)	0.4746~(60%)	0.4752(80%)	0.4759(50%)	$\overline{0.4902}$
enron	0.5432(20%)	$\overline{0.5662}$ (90%)	0.5483(80%)	0.5599(80%)	0.5665
flagsml	$\overline{0.3680(80\%)}$	0.3626(90%)	0.3626(90%)	0.3723(90%)	0.3699
genbase	0.0138 (10%)	$\overline{0.0138}$ (10%)	0.0138(10%)	0.0138(10%)	0.0138
medical	$\overline{0.2348(20\%)}$	0.2408(10%)	0.2391 (10%)	0.2416(10%)	0.2520
scene	$\overline{0.3750(80\%)}$	0.3767(50%)	0.3749(90%)	0.3729(70%)	0.3753
veast	0.5108 (70%)	0.5129(40%)	0.5208(90%)	0.5121(30%)	0.5126
RANKING LO	SS		. ,	. ,	
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	n/a	n/a	0.2648 (60%)	0.3113 (10%)	0.2618
birds	0.1221 (20%)	$0.1065\ (50\%)$	0.1094~(50%)	0.1146(30%)	0.1366
CAL500	0.2869 (70%)	0.2830(80%)	0.2825 $(80%)$	0.2852 $(80%)$	0.2869
Corel5 k	0.6650 (90%)	0.6577 (90%)	0.6630 (90%)	0.6564 (90%)	0.6565
emotions	0.1779 (60%)	0.1781~(60%)	0.1816 $(80%)$	0.1767 (60%)	0.1891
enron	0.1930 (60%)	0.2031 (90%)	0.1926(80%)	0.2015 (90%)	0.2005
flagsml	0.2121 (80%)	0.2184 (90%)	0.2184(90%)	0.2238 (80%)	0.2330
genbase	$\overline{0.0026}$ (10%)	0.0026 (10%)	0.0026(10%)	0.0026(10%)	0.0026
medical	$\overline{0.0715}$ (20%)	$\overline{0.0750}$ (30%)	0.0748(30%)	0.0752(30%)	0.0777
scene	0.0979(60%)	0.1003 (80%)	0.1005 (90%)	0.1000 (80%)	0.0999
yeast	0.2085(70%)	0.2111 (50%)	0.2146(80%)	0.2125(50%)	0.2135
Best values		12		12	6
(underlined)	22	13	11	12	9
\leq baseline score	27	20	<u> </u>	9 5	
(bold)	31	32	<u>აა</u>	30	

Table 7.3: Best results achieved with the RK + DecisionTree classifier

HAMMING LO	DSS				
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.0128~(10%)	0.0132~(10%)	0.0136~(30%)	0.0132~(10%)	0.0143
birds	$\overline{0.0450}$ (30%)	0.0461~(90%)	0.0467~(90%)	0.0450~(60%)	0.0458
CAL500	$\overline{0.1456}$ (20%)	0.1470~(40%)	0.1456~(50%)	0.1461 (30%)	0.1473
Corel5 k	$\overline{0.0094}$ (10%)	0.0094(10%)	0.0094(10%)	0.0094(10%)	n/a
emotions	$\overline{0.1937}(50\%)$	0.1974(90%)	0.1982 (90%)	$\overline{0.1954}$ (90%)	0.1955
enron	$\overline{0.0525(10\%)}$	0.0581(10%)	0.0519(10%)	0.0566(70%)	0.0580
flagsml	0.2526(50%)	0.2613 (20%)	0.2534(50%)	0.2546(20%)	0.2791
genbase	$\overline{0.0038(10\%)}$	0.0038(10%)	0.0038(10%)	0.0038(10%)	0.0038
medical	$\overline{0.0137(10\%)}$	$\overline{0.0158}$ (10%)	0.0159(10%)	$\overline{0.0160(10\%)}$	0.0178
scene	0.0936 (90%)	0.0931(90%)	0.0959(90%)	0.0922(80%)	0.0924
yeast	0.1998 (70%)	0.2000 (90%)	0.2036~(90%)	0.2021 (80%)	0.2015
SUBSET 0/1 L	OSS			. ,	
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.8786 (10%)	0.9081 (10%)	0.9474(30%)	0.9078 (10%)	0.9735
birds	$\overline{0.4929}$ (70%)	0.5054(90%)	0.5037(80%)	0.4867(60%)	0.5038
CAL500	1.0000 (10%)	1.0000 (10%)	1.0000(10%)	1.0000(10%)	1.0000
Corel5 k	$\overline{0.9990}$ (50%)	$\overline{0.9994}$ (10%)	1.0000(10%)	0.9990(40%)	n/a
emotions	$\overline{0.6580(50\%)}$	0.6613(20%)	0.6615(50%)	$\overline{0.6648}$ (90%)	0.6666
enron	$\overline{0.8855(10\%)}$	0.8837(40%)	0.8908 (50%)	0.8843(30%)	0.9130
flagsml	0.7621 (10%)	$\overline{0.8082}$ (90%)	0.7616(50%)	0.7776(10%)	0.8292
genbase	0.0785 (10%)	0.0785(10%)	0.0785(10%)	0.0785(10%)	0.0785
medical	$\overline{0.4417(10\%)}$	$\overline{0.5236}$ (10%)	0.5297 (10%)	0.5266(10%)	0.5808
scene	0.3357 (90%)	0.3336(90%)	0.3378(90%)	0.3286(80%)	0.3295
veast	0.7795 (90%)	0.7815(80%)	0.7795(90%)	0.7774(80%)	0.7770
EXAMPLE BA	SED ACCURAC	CY (inverted)	. ,		
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.7836~(10%)	0.8308~(10%)	$0.8791 \ (30\%)$	0.8298~(10%)	0.9247
birds	$\overline{0.4254} (30\%)$	$0.4392\ (70\%)$	0.4386~(80%)	0.4284~(60%)	0.4375
CAL500	$\overline{0.7759} (80\%)$	0.7780~(50%)	0.7744~(50%)	0.7762~(40%)	0.7784
Corel5 k	0.9912 (40%)	0.9921~(30%)	$\overline{0.9947} (30\%)$	0.9910~(50%)	n/a
emotions	0.4253~(50%)	0.4292~(70%)	0.4287~(50%)	0.4267(90%)	0.4266
enron	$\overline{0.6420}$ (10%)	0.7256~(20%)	0.6844~(10%)	0.7083~(70%)	0.7827
flagsml	$\overline{0.3826}$ (20%)	0.3925~(20%)	0.3798~(50%)	0.3809~(20%)	0.4172
genbase	0.0463 (10%)	0.0463~(10%)	$\overline{0.0463} (10\%)$	0.0463~(10%)	0.0463
medical	$\overline{0.3669} (10\%)$	$\overline{0.4562} (10\%)$	0.4644~(10%)	$\overline{0.4617~(10\%)}$	0.5228
scene	0.2974 (90%)	0.2957~(90%)	0.3022~(90%)	0.2913 $(80%)$	0.2931
yeast	0.4649 (90%)	0.4655~(90%)	0.4695~(90%)	$\overline{0.4679} (80\%)$	0.4686
RANKING LO	ss				
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.3783~(10%)	0.4101~(10%)	0.4505~(30%)	0.4090~(10%)	0.4845
birds	$\overline{0.1963} \ (30\%)$	0.2031~(70%)	0.2056~(70%)	0.2009~(50%)	0.2059
CAL500	0.4187 (70%)	0.4157~(40%)	0.4139~(90%)	0.4162~(80%)	0.4147
Corel5 k	0.7433~(40%)	0.7439~(30%)	$\overline{0.7472} (30\%)$	0.7422~(30%)	n/a
emotions	0.2163~(80%)	0.2105~(90%)	0.2176~(90%)	0.2138 (90%)	0.2104
enron	0.3229~(10%)	0.3825~(10%)	0.3453~(10%)	0.3622~(50%)	0.4416
flagsml	$\overline{0.2397}$ (50%)	0.2315~(20%)	0.2472~(50%)	0.2399~(90%)	0.2679
genbase	0.0231 (10%)	0.0231 (10%)	0.0231~(10%)	$0.0231 \ (10\%)$	<u>0.0231</u>
medical	0.1400(10%)	$\overline{0.1736\ (20\%)}$	$\overline{0.1702} \ (10\%)$	$\overline{0.1716~(20\%)}$	0.1958
scene	0.1448 (90%)	$0.1431 \ (90\%)$	0.1476~(90%)	0.1426 $(80%)$	0.1447
yeast	0.2501 (90%)	0.2500 (90%)	$0.2536\ (80\%)$	0.2515(30%)	0.2527
Best values		0	10	15	7
(underlined)	Z1	9	12	10	<u> </u>
\leq baseline score	37	39	31	30	
(bold)	"	02	01	00	

Table 7.4: Best results achieved with the RAKEL + K-NN classifier

	HAMMING LOSS						
	Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
birds 0.0487 (60%) 0.0497 (90%) 0.0481 (80%) 0.0483 CAL500 0.0100 (40%) 0.0168 (70%) 0.0132 (20%) 0.1625 (20%) 0.1646 Corel5k 0.0100 (40%) 0.2005 (90%) 0.1399 (00%) 0.2005 0.0994 (90%) 0.2087 emotions 0.2005 (70%) 0.2057 (10%) 0.0527 (10%) 0.0537 (90%) 0.2532 genbase 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 0.0044 medical 0.0133 (10%) 0.0145 (10%) 0.0147 (10%) 0.0153 (10%) 0.113 yeast 0.2120 (80%) 0.2144 (80%) 0.2114 (80%) 0.2126 0.2126 SUBST 0/1 LOSS Data Set D.8421 (10%) 0.9040 (30%) 0.894 (10%) 0.9455 bibtex 0.4211 (10%) 0.9000 (10%) 1.9400 (10%) 1.0000 (10%) 1.0000 0.5161 CAL500 1.0000 (10%) 1.0000 (10%) 1.0000 (10%) 1.0000 0.5561 0.5561 CAL500 1.0000 (10%) 1.0000 (10%) 1.0000 0.526 (80%)	bibtex	0.0128~(10%)	0.0132~(10%)	0.0137~(30%)	0.0132~(10%)	0.0143	
$ \begin{array}{c classes} CAL500 & \hline 0.1616 (30\%) & 0.1628 (70\%) & 0.1633 (20\%) & 0.1628 (20\%) & 0.1643 (20\%) & 0.0094 (40\%) & 0.7a \\ \hline Corel5k & 0.000 (70\%) & 0.2033 (90\%) & 0.0098 (30\%) & 0.0094 (40\%) & 0.2098 \\ \hline Corel5k & 0.0024 (10\%) & 0.0587 (90\%) & 0.0527 (10\%) & 0.0577 (90\%) & 0.2588 \\ \hline Corel5k & 0.0044 (10\%) & 0.0512 (40\%) & 0.0217 (10\%) & 0.0044 (10\%) & 0.0588 \\ \hline Corel5k & 0.0038 (90\%) & 0.0153 (10\%) & 0.0147 (10\%) & 0.0044 (10\%) & 0.0044 \\ \hline Corel5k & 0.0358 (90\%) & 0.01513 (10\%) & 0.0147 (10\%) & 0.0143 (10\%) & 0.0147 \\ \hline Corel5k & 0.0358 (90\%) & 0.01513 (10\%) & 0.0147 (10\%) & 0.0144 (10\%) & 0.0144 \\ \hline Corel5k & 0.0358 (90\%) & 0.01513 (10\%) & 0.0147 (10\%) & 0.0144 (10\%) & 0.0988 \\ \hline Corel5k & 0.0358 (90\%) & 0.01510 (70\%) & 0.5101 (70\%) & 0.2104 (90\%) & 0.2124 \\ \hline Data Set & BR+1n & Cain & Cpy+1n & Cain & H-1n & Cain & No Set. \\ \hline Dibtex & 0.8621 (10\%) & 0.9003 (10\%) & 0.9409 (30\%) & 0.8904 (10\%) & 0.9688 \\ \hline Dirds & 0.4901 (70\%) & 0.5100 (70\%) & 0.5131 (80\%) & 0.5006 (60\%) & 0.5161 \\ CAL500 & 1.0000 (10\%) & 1.0000 (10\%) & 1.0000 (10\%) & 1.0000 (10\%) & 1.000 \\ Corel5k & 0.9942 (50\%) & 0.9950 (10\%) & 0.9972 (10\%) & 0.9480 (10\%) & n.4 \\ emotions & 0.6411 (70\%) & 0.6443 (70\%) & 0.6529 (80\%) & 0.6439 (70\%) & 0.6580 \\ enron & 0.6431 (70\%) & 0.8476 (40\%) & 0.7566 (70\%) & 0.7674 (10\%) & 0.8184 \\ genbase & 0.0756 (10\%) & 0.3750 (10\%) & 0.0755 (10\%) & 0.3751 (70\%) & 0.3358 \\ scene & 0.3153 (90\%) & 0.3149 (90\%) & 0.3195 (90\%) & 0.3124 (90\%) & 0.3124 \\ yeast & 0.7626 (80\%) & 0.3723 (10\%) & 0.4645 (50\%) & 0.7516 (70\%) & 0.7551 \\ \hline EXAMPLE BASED ACCURACY (inverted) \\ \hline Data Set & BR+1n & Cain & Cpy+1n & Cain & MLIn & Cain & No Sel. \\ \hline Dibtex & 0.7621 (10\%) & 0.3424 (10\%) & 0.3418 (10\%) & 0.4014 (10\%) & 0.4122 \\ birds & 0.762 (20\%) & 0.3266 (10\%) & 0.3913 (10\%) & 0.0122 (90\%) & 0.3733 (80\%) & 0.3124 (90\%) & 0.3755 \\ Corel5k & 0.9762 (20\%) & 0.3281 (10\%) & 0.3014 (10\%) & 0.4042 (10\%) & 0.4424 (10\%) & 0.4424 (10\%) & 0.4424 (10\%) & 0.4424 (10\%) & 0.4424 (10\%) & 0.4424 (10\%) & 0.4424 (10\%) & 0.4424 (10\%) & 0.4424 (10\%) & 0.44$	birds	0.0480 (90%)	0.0495 (90%)	0.0497 (90%)	0.0481 ($80%$)	0.0495	
	CAL500	0.1616(30%)	0.1628(70%)	0.1633 (20%)	0.1625(20%)	0.1646	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	Corel5 k	$\overline{0.0100(40\%)}$	0.0096(40%)	0.0098 (30%)	0.0094 (90%)	n/a	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	emotions	0.2008(70%)	0.2053 (90%)	0.2055 (90%)	$\frac{0.1999}{0.1999}$ (90%)	0.2098	
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	enron	0.0527 (10%)	0.0587 (90%)	0.0527 (10%)	$\frac{0.0577}{0.0577}$ (90%)	0.0587	
genbase 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0044 (10%) 0.0153 (10%) 0.0173 (10%) 0.0133 (10%) 0.0144 (10%) 0.0044 (10%) 0.0153 (10%) 0.0142 (00%) 0.0143 scene 0.2120 (80%) 0.2144 (80%) 0.2150 (80%) 0.0942 (90%) 0.2133 (80%) 0.2949 (30%) 0.2949 (30%) 0.8941 (10%) 0.2161 birds 0.8621 (10%) 0.9003 (10%) 0.5006 (80%) 0.5161 (70%) 0.5308 (80%) 0.5666 (60%) 0.566 (60%) 0.566 (60%) 0.566 (60%) 0.566 (70%) 0.6435 (70%) 0.6435 (70%) 0.6435 (70%) 0.6435 (70%) 0.5486 (70%) 0.7674 (10%) 0.8848 (10%) 0.6351 (10%) 0.7674 (10%) 0.5864 (10%) 0.666 (70%) 0.7664 (10%) 0.6435 (10%) 0.7675 (10%) 0.7551 (10%) 0.7551 (10%) 0.7551 (10%) 0.7551 (10%) 0.7551 (10%) 0.7551 (10%	flagsml	$\frac{1}{0.2496}$ (20%)	0.2612 (40%)	$\frac{0.2474}{0.2474}$ (70%)	0.2523(20%)	0.2688	
	genbase	0.0044 (10%)	0.0044 (10%)	$\frac{0.0044}{0.0044}$ (10%)	0.0044 (10%)	0.0044	
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	medical	$\frac{0.00112}{0.0133}$ (10%)	$\frac{0.00011}{0.0153}$ (10%)	$\frac{0.00117}{0.0147}$ (10%)	$\frac{0.00012}{0.0153}$ (10%)	0.0174	
$\begin{tabular}{ c c c c c c c c c c c c c c c c c c c$	scene	$\frac{0.0100(100)}{0.0958(90\%)}$	0.0956 (90%)	0.0974 (90%)	0.0942 (90%)	0.0945	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	veset	0.2120 (80%)	0.0000 (90%) 0.2144 (80%)	0.0011(0070) 0.2150(80%)	$\frac{0.0042}{0.2113}$ (80%)	0.0316	
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	SUBSET 0/1 L	OSS	0.2144 (0070)	0.2130 (0070)	0.2115 (8070)	0.2120	
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	Data Set	BB InfoCain	Conv InfoC ain	I D InfoC ain	MUnfoCain	No Sol	
	bibtor	0.8621 (10%)	$\frac{0.003}{10\%}$	$\frac{11 + 1110 \text{Gall}}{0.0400}$	0.8004 (10%)	0.0685	
	binda	$\frac{0.3021}{0.4001}$ (70%)	0.9003 (1070)	0.9409 (3070)	0.8994 (1070)	0.9085	
$\begin{array}{cccc} Carelsk & 0.9942 (50\%) & 0.9950 (10\%) & 0.9907 (10\%) & 0.9480 (10\%) & n/a \\ emotions & 0.6411 (70\%) & 0.6443 (70\%) & 0.6529 (80\%) & 0.6495 (70\%) & 0.6580 \\ enron & 0.3643 (10\%) & 0.8819 (20\%) & 0.3761 (10\%) & 0.8714 (70\%) & 0.8866 \\ flagsml & 0.7524 (20\%) & 0.8076 (40\%) & 0.7661 (10\%) & 0.8714 (70\%) & 0.8184 \\ genbase & 0.0755 (10\%) & 0.0755 (10\%) & 0.0755 (10\%) & 0.07574 (10\%) & 0.8184 \\ genbase & 0.3153 (90\%) & 0.3149 (90\%) & 0.3195 (90\%) & 0.4735 (10\%) & 0.5358 \\ scene & 0.3153 (90\%) & 0.3149 (90\%) & 0.3195 (90\%) & 0.4735 (10\%) & 0.5358 \\ scene & 0.3153 (90\%) & 0.3149 (90\%) & 0.3195 (90\%) & 0.3124 (90\%) & 0.5358 \\ scene & 0.3153 (90\%) & 0.7530 (70\%) & 0.7497 (80\%) & 0.7530 (80\%) & 0.7501 \\ \hline EXAMPLE BASED ACCURACY (inverted) \\ \hline Data Set & BR+HnGGain & Copy+InfoGain & LP+InfoGain & MLInfoGain & No Sel. \\ bibtex & 0.7502 (80\%) & 0.7494 (70\%) & 0.8496 (10\%) & 0.8495 (30\%) & 0.8194 (10\%) & 0.9122 \\ birds & 0.4333 (40\%) & 0.4418 (70\%) & 0.4464 (50\%) & 0.4302 (50\%) & 0.4536 \\ CAL500 & 0.7502 (40\%) & 0.7494 (70\%) & 0.7534 (50\%) & 0.7515 (70\%) & 0.7555 \\ Corel5k & 0.9762 (20\%) & 0.9723 (10\%) & 0.9803 (10\%) & 0.9716 (20\%) & n/a \\ emotions & 0.4109 (70\%) & 0.8398 (10\%) & 0.3913 (10\%) & 0.6804 (50\%) & 0.7363 \\ genbase & 0.0442 (10\%) & 0.03926 (40\%) & 0.3701 (70\%) & 0.3806 (20\%) & 0.4428 \\ enron & 0.6116 (10\%) & 0.03998 (10\%) & 0.3913 (10\%) & 0.4014 (10\%) & 0.4424 \\ enron & 0.6116 (10\%) & 0.0398 (10\%) & 0.3913 (10\%) & 0.1085 (10\%) & 0.4697 \\ \hline RANKING LOSS \\ \hline Data Set & BR+InfoGain & Copy+InfoGain & LP+InfoGain & MLInfoGain & No Sel. \\ bibtex & 0.1297 (10\%) & 0.1422 (90\%) & 0.1432 (90\%) & 0.1695 (10\%) & 0.2649 \\ corel5k & 0.0952 (80\%) & 0.0288 (40\%) & 0.2908 (20\%) & 0.1695 (10\%) & 0.2649 \\ corel5k & 0.02333 (30\%) & 0.2300 (40\%) & 0.1032 (10\%) & 0.1695 (10\%) & 0.2690 \\ corel5k & 0.02333 (30\%) & 0.2300 (40\%) & 0.1322 (10\%) & 0.1664 (30\%) & 0.1664 \\ flagsml & 0.1137 (70\%) & 0.1842 (20\%) & 0.0441 (10\%) & 0.0041 (10\%) & 0.0041 \\ medical & 0.0394 (10\%) & 0.0448 (20\%) & 0.0954 (90\%) & 0.0913 (90\%) & 0.928 \\ yeast &$	CALEGO	$\frac{0.4991}{1.0000}$ (10%)	1,0000 (10%)	1,0000 (1007)	1,0000 (0076)	1 0000	
$\begin{array}{c} \text{Corebac} & 0.3942 \ (30\%) & 0.3950 \ (10\%) & 0.3942 \ (10\%) & 0.3940 \ (10\%) & 0.3940 \ (10\%) & 0.6529 \ (30\%) & 0.6495 \ (70\%) & 0.6580 \ (30\%) & 0.6580 \ (70\%) & 0.6580 \ (30\%) & 0.6580 \ (70\%) & 0.6580 \ (30\%) & 0.7674 \ (10\%) & 0.8866 \ (30\%) & 0.7675 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7755 \ (10\%) & 0.7536 \ (30\%) & 0.3124 \ (90\%) & 0.3137 \ (90\%) & 0.3124 \ (90\%) & 0.3124 \ (90\%) & 0.3137 \ (90\%) & 0.7516 \ (80\%) & 0.7501 \ (80\%) & 0.7750 \ (80\%) & 0.7501 \ (80\%) & 0.7$	CAL500	$\frac{1.0000(10\%)}{0.0040(50\%)}$	$\frac{1.0000(10\%)}{0.0050(10\%)}$	$\frac{1.0000(10\%)}{0.0070(10\%)}$	$\frac{1.0000(10\%)}{0.0480(10\%)}$	<u>1.0000</u>	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	Coreisk	0.9942(50%)	0.9950 (10%)	0.9972 (10%)	$\frac{0.9480}{0.9485}$ (10%)		
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	emotions	$\frac{0.6411}{0.000}$	0.6443 (70%)	0.6529(80%)	0.6495(70%)	0.6580	
$\begin{array}{llllllllllllllllllllllllllllllllllll$	enron	$\frac{0.8643}{0.8643}$	0.8819(20%)	0.8761 (10%)	0.8714(70%)	0.8866	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	flagsml	0.7824(20%)	0.8076(40%)	0.7666 (70%)	0.7674(10%)	0.8184	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	genbase	0.0755 (10%)	0.0755 (10%)	0.0755 (10%)	0.0755(10%)	0.0755	
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	medical	$0.3977 \ (10\%)$	0.4724~(10%)	0.4653~(10%)	0.4735~(10%)	0.5358	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	scene	0.3153 (90%)	0.3149~(90%)	0.3195~(90%)	0.3124~(90%)	0.3137	
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	yeast	0.7526~(80%)	$0.7530\ (70\%)$	0.7497~(80%)	0.7530~(80%)	0.7501	
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	EXAMPLE BA	SED ACCURAC	CY (inverted)				
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	bibtex	$0.7651 \ (10\%)$	$0.8206\ (10\%)$	0.8695~(30%)	0.8194~(10%)	0.9122	
$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	birds	0.4333 $(40%)$	$0.4418\ (70\%)$	0.4464~(50%)	0.4302~(50%)	0.4536	
$\begin{array}{c cccc} Corel5k & 0.9762 (20\%) & 0.9723 (10\%) & 0.9803 (10\%) & 0.9716 (20\%) & n/a \\ emotions & 0.4109 (70\%) & 0.4188 (90\%) & 0.4195 (90\%) & 0.4129 (90\%) & 0.4248 \\ enron & 0.6116 (10\%) & 0.6894 (10\%) & 0.6341 (10\%) & 0.6804 (50\%) & 0.7363 \\ flagsml & 0.3824 (80\%) & 0.3926 (40\%) & 0.3701 (70\%) & 0.3806 (20\%) & 0.4036 \\ genbase & 0.0442 (10\%) & 0.0442 (10\%) & 0.0442 (10\%) & 0.0442 (10\%) \\ medical & 0.3189 (10\%) & 0.3998 (10\%) & 0.3913 (10\%) & 0.4014 (10\%) & 0.4412 \\ medical & 0.2839 (90\%) & 0.2822 (90\%) & 0.2880 (90\%) & 0.2799 (90\%) & 0.2813 \\ yeast & 0.4704 (80\%) & 0.4729 (80\%) & 0.4734 (80\%) & 0.4691 (80\%) & 0.4697 \\ \hline RANKING LOSS \\ \hline Data Set & BR+InfoGain & Copy+InfoGain & LP+InfoGain & MLInfoGain & No Sel. \\ bibtex & 0.1297 (10\%) & 0.1715 (10\%) & 0.2166 (30\%) & 0.1695 (10\%) & 0.2690 \\ birds & 0.0952 (80\%) & 0.0889 (90\%) & 0.1008 (90\%) & 0.1000 (50\%) & 0.1029 \\ CAL500 & 0.2876 (40\%) & 0.2883 (40\%) & 0.22908 (20\%) & 0.2877 (70\%) & 0.2949 \\ Corel5k & 0.1323 (30\%) & 0.2300 (40\%) & 0.2447 (20\%) & 0.2329 (20\%) & n/a \\ emotions & 0.1730 (70\%) & 0.1712 (90\%) & 0.1751 (90\%) & 0.1725 (90\%) & 0.1742 \\ enron & 0.1327 (10\%) & 0.1840 (40\%) & 0.1827 (50\%) & 0.1874 (20\%) & 0.1954 \\ genbase & 0.0041 (10\%) & 0.0041 (10\%) & 0.0041 (10\%) & 0.0041 (10\%) \\ genbase & 0.0041 (10\%) & 0.0041 (10\%) & 0.0041 (10\%) & 0.0041 \\ medical & 0.0394 (10\%) & 0.2185 (90\%) & 0.2188 (80\%) & 0.2164 (80\%) & 0.2182 \\ Best values \\ (underlined) & 24 & 8 & 11 & 17 & 5 \\ \hline \end{array}$	CAL500	0.7502~(40%)	0.7494 (70%)	0.7534~(50%)	$0.7515\ (70\%)$	0.7555	
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	$\operatorname{Corel5k}$	0.9762~(20%)	0.9723~(10%)	0.9803~(10%)	0.9716~(20%)	n/a	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	emotions	0.4109 (70%)	0.4188~(90%)	0.4195~(90%)	0.4129~(90%)	0.4248	
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	enron	$\overline{0.6116}$ (10%)	0.6894~(10%)	0.6341~(10%)	0.6804~(50%)	0.7363	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	flagsml	$\overline{0.3824} \ (80\%)$	0.3926~(40%)	0.3701~(70%)	0.3806~(20%)	0.4036	
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	genbase	0.0442 (10%)	0.0442 (10%)	0.0442 (10%)	0.0442 (10%)	0.0442	
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	medical	$\overline{0.3189(10\%)}$	$\overline{0.3998}$ (10%)	0.3913(10%)	0.4014(10%)	0.4712	
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	scene	0.2839 (90%)	0.2822(90%)	0.2880(90%)	0.2799 (90%)	0.2813	
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	yeast	0.4704 (80%)	0.4729(80%)	0.4734(80%)	0.4691 (80%)	0.4697	
$\begin{array}{ c c c c c c c c c c c c c c c c c c c$	RANKING LO	SS		. ,			
$ \begin{array}{ c c c c c c c c c c c c c c c c c c c$	Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	bibtex	0.1297 (10%)	0.1715~(10%)	0.2166~(30%)	0.1695~(10%)	0.2690	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	birds	0.0952 (80%)	0.0989 (90%)	0.1008 (90%)	0.1000 (50%)	0.1029	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	CAL500	0.2876(40%)	0.2883(40%)	0.2908(20%)	0.2877(70%)	0.2949	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	Corel5 k	$\overline{0.2333}(30\%)$	0.2300(40%)	0.2447 (20%)	0.2329(20%)	n/a	
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	emotions	0.1730 (70%)	$\overline{0.1712}$ (90%)	0.1751(90%)	0.1725(90%)	0.1742	
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	enron	0.1327(10%)	$\overline{0.1461}$ (20%)	0.1392(10%)	0.1482(20%)	0.1664	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	flagsml	$\frac{1}{0.1878}$ (70%)	0.1840 (40%)	0.1827(50%)	0.1874(20%)	0.1954	
$ \begin{array}{c ccccccccccccccccccccccccccccccccccc$	genbase	0.0041 (10%)	0.0041 (10%)	$\frac{0.0041}{0.0041}$ (10%)	0.0041 (10%)	0.0041	
$\begin{array}{c c c c c c c c c c c c c c c c c c c $	medical	$\frac{0.0011}{0.0394}$ (10%)	$\frac{0.0011}{0.0484}$ (20%)	$\frac{0.0011}{0.0478}$ (20%)	$\frac{0.0011}{0.0491}$ (10%)	$\frac{0.0011}{0.0507}$	
yeast 0.2171 (80%) 0.2185 (90%) 0.0354 (30%) 0.0313 (30%) 0.0313 (30%)Best values (underlined)24811175	scene	$\frac{0.0009}{0.0952}$ (00%)		0.0954 (90%)		0.0007	
Best values (underlined) 24 8 11 17 5	veast	0.2171 (80%)		0.2188 (80%)	$\frac{0.0010}{0.2164}$ (80%)	0.2182	
(underlined) 24 8 11 17 5	Best values		0.2100 (0070)	0.2100 (0070)	5.2101 (0070)	0.2102	
	(underlined)	24	8	11	17	5	
< baseline score	\leq baseline score	2-		0.7			
$\overline{(bold)}$ 38 37 35 43	(bold)	38	37	35	43		

Table 7.5: Best results achieved with the CC + K-NN classifier

HAMMING LOSS					
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.0130 (10%)	0.0134 (10%)	0.0146~(10%)	0.0133~(10%)	0.0146
birds	$\overline{0.0446}$ (30%)	0.0462(50%)	0.0458(50%)	0.0445(10%)	0.0494
CAL500	0.1392 (10%)	0.1386(10%)	0.1396(10%)	0.1388(10%)	0.1615
Corel5k	0.0094 (10%)	$\overline{0.0095(10\%)}$	0.0094(10%)	0.0095(10%)	0.0098
emotions	$\frac{1}{0.2317}$ (50%)	0.2311 (20%)	$\frac{0.2313}{0.2313}$ (10%)	0.2254(10%)	0.2474
enron	0.0497 (10%)	0.0513(90%)	0.0504 (70%)	0.0504(80%)	0.0508
flagsml	$\overline{0.2501(90\%)}$	0.2598(90%)	0.2456(80%)	0.2443(50%)	0.2627
genbase	0.0011 (10%)	0.0011 (10%)	0.0011(10%)	$\overline{0.0011(10\%)}$	0.0011
medical	$\frac{0.00911}{0.0096}$ (20%)	$\frac{0.0100}{0.0100}$ (10%)	$\frac{0.0100}{0.0100}$ (10%)	$\frac{0.0100}{0.0100}$ (10%)	0.0103
scene	$\frac{0.0000}{0.1340}$ (60%)	0.1314 (70%)	0.1307 (60%)	0.1305(70%)	0 1368
veast	0.1010(0070) 0.2159(10%)	0.2153 (10%)	0.2253(20%)	$\frac{0.1000}{0.2140}$ (10%)	0.1500
SUBSET 0/1 I	0.2100 (1070)	0.2100 (1070)	0.2200 (2070)	0.2140 (1070)	0.2404
Data Set	BB InfoCain	Conv InfoC ain	I D InfoC ain	MUnfoChin	No Sol
bibtov	0.8344 (10%)	$\frac{\text{Copy}+\text{IntoGam}}{0.8508}$	$\frac{11 + 1110 \text{Gall}}{0.8588}$ (70%)	0.8405 (20%)	0.8602
binda	$\frac{0.8344}{0.4882}$ (10%)	0.8508 (2070)	0.8388 (70%)	0.8493 (2070) 0.4859 (10%)	0.8002
DIFUS	1,4002 (1070)	0.4990 (00%)	0.4900(1076)	$\frac{0.4855(1076)}{1.0000(1007)}$	0.0140
CAL500	$\frac{1.0000(10\%)}{0.0074(00\%)}$	$\frac{1.0000(10\%)}{0.00\%}$	$\frac{1.0000(10\%)}{0.000}$	$\frac{1.0000(10\%)}{0.0000(00\%)}$	1.0000
Corelsk		$\frac{0.9958}{0.9958}$ (90%)	0.9982(90%)	0.9962 (80%)	0.9974
emotions	0.7774 (10%)	0.7724(10%)	$\frac{0.7640(10\%)}{0.7640(10\%)}$	0.7706 (10%)	0.8162
enron	$\frac{0.8673 (10\%)}{(10\%)}$	0.8878 (70%)	0.8884 (70%)	0.8860 (70%)	0.8972
flagsml	0.7884 (10%)	0.8292~(90%)	0.8042~(10%)	0.8090~(50%)	0.8450
genbase	0.0287 (10%)	0.0287~(10%)	0.0287~(10%)	0.0287~(10%)	0.0287
medical	$0.3191 \ (20\%)$	0.3262~(10%)	0.3283~(10%)	0.3242~(10%)	0.3447
scene	$0.5621 \ (60\%)$	$0.5567 \ (70\%)$	0.5571~(60%)	$0.5609\ (70\%)$	0.5734
yeast	0.9007 (20%)	0.8986~(30%)	0.9317~(20%)	0.8920~(20%)	0.9317
EXAMPLE BA	SED ACCURAC	CY (inverted)			
Data Set	BR +InfoGain	$\mathbf{Copy} + \mathbf{InfoGain}$	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.6796 (20%)	0.6878~(60%)	0.6921 (70%)	0.6879~(80%)	0.6936
birds	$\overline{0.3917} (30\%)$	0.4149~(50%)	0.4122~(50%)	0.3982~(10%)	0.4265
CAL500	0.7868~(60%)	0.7819~(80%)	0.7899 $(70%)$	$0.7859\ (70\%)$	0.7933
$\operatorname{Corel5k}$	0.9419 (90%)	0.9370(90%)	0.9432~(90%)	0.9371~(90%)	0.9367
emotions	0.5284 (50%)	0.5235~(30%)	0.5202 $(20%)$	$0.5169\ (10\%)$	0.5377
enron	0.5659 (10%)	0.5918~(90%)	0.5723~(70%)	$\overline{0.5818}$ (80%)	0.5871
flagsml	$\overline{0.3779}(50\%)$	0.3977 (90%)	0.3690(80%)	0.3752(50%)	0.3989
genbase	0.0138 (10%)	0.0138(10%)	0.0138(10%)	0.0138(10%)	0.0138
medical	$\overline{0.2312}$ (20%)	$\overline{0.2405(10\%)}$	0.2398(10%)	0.2401(10%)	0.2535
scene	$\overline{0.4618}$ (90%)	0.4544 (70%)	0.4575(60%)	0.4569(50%)	0.4647
veast	0.5312(20%)	$\frac{0.10112}{0.5268}$ (30%)	0.5589 (80%)	0.5298(30%)	0 5605
BANKING LO	SS (2070)			0.0200 (0070)	0.0000
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.1595 (90%)	0.1630 (90%)	0.1558 (90%)	0.1631 (90%)	0.1585
birds	0.1480 (10%)	0.1454 (40%)	$\frac{0.1459}{0.1459}$ (40%)	0.1461 (10%)	0 1579
CAI 500	0.1400(10%) 0.1827(10%)	$\frac{0.1404}{0.1843}$ (10%)	0.1405 (40%) 0.1835 (10%)	0.1401(10%) 0.1820(10\%)	0.1075
Carolfi	$\frac{0.1027}{0.1418}$ (40%)	0.1040 (1070) 0.1494 (90%)	0.1000 (1070) 0.1498 (1077)	0.1029 (1070) 0.1494 (40%)	0.3043
omotions	$\frac{0.1418}{0.2425} (40\%)$	0.1434 (2070) 0.2287 (10%)	0.1430 (10%)	0.1434 (4070) 0.2212 (10 $\%$)	0.1472
emotions	$\begin{array}{c c} 0.2420 & (1070) \\ 0.1991 & (1007) \end{array}$	0.4401 (1070) 0.1106 (1077)	0.2294 (1070) 0.1605 (1007)	$\frac{0.2212}{0.1161}$ (1070)	0.2910
enron g		0.0000 (000%)	0.1000 (10%)	$\frac{0.1101}{0.0040}$ (10%)	0.1/23
nagsm	0.2180(40%)	$\frac{0.2002}{0.000}$ (20%)	0.2220 (10%)	0.2240 (40%)	0.2802
genbase	$\frac{0.0028 (10\%)}{0.0522 (10\%)}$	$\frac{0.0028 (10\%)}{0.0021 (10\%)}$	$\frac{0.0028 (10\%)}{0.0028 (10\%)}$	$\frac{0.0028 (10\%)}{0.0028 (10\%)}$	0.0028
medical	0.0522 (10%)	0.0661 (10%)	0.0688 (10%)	0.0660 (10%)	0.0743
scene	0.2036 (10%)	0.2425 (70%)	0.2356 (60%)	0.2333 (80%)	0.2465
yeast	0.2028 (10%)	0.2054 (10%)	0.2089 (10%)	0.2033~(10%)	0.3097
Best values	22	13	9	15	6
(underlined)			2		
\leq baseline score	42	40	42	42	
(bold)					

Table 7.6: Best results achieved with the BR + DecisionTree classifier

HAMMING LOSS						
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.0641 (10%)	0.0908 (90%)	0.0910 (90%)	0.0909(90%)	0.0874	
birds	$\overline{0.2223(10\%)}$	0.3555(90%)	0.3554(90%)	0.1579(10%)	0.3528	
CAL500	0.1620 (10%)	0.1566(10%)	0.1605(10%)	$\overline{0.1602(10\%)}$	0.3187	
Corel5k	0.0097 (10%)	$\frac{0.2000(10\%)}{0.0113(10\%)}$	0.0103 (10%)	0.0113(10%)	0.0127	
emotions	$\frac{0.00007}{0.2557}$ (90%)	0.2532 (90%)	0.2540.(90%)	0.2568 (90%)	0.2521	
enron	0.2001 (0070)	0.2002(0070) 0.1967(10%)	0.2340(3070)	0.2000(0070) 0.1975(10%)	$\frac{0.2521}{0.2177}$	
flagsml	$\frac{0.1012}{0.2453}$ (20%)	0.2681 (40%)	0.2533(40%)	0.2548(20%)	0.3214	
gonbaso	$\frac{0.2400}{0.0041}$ (10%)	0.2001 (40%)	0.2000 (10%)	0.2048 (20%)	0.0211	
modical	$\frac{0.0041}{0.0188}$ (10%)	$\frac{0.0041}{0.0221}$ (00%)	$\frac{0.0041}{0.0221}$ (00%)	$\frac{0.0041}{0.0220}$ (80%)	0.0353	
nieurcar	$\frac{0.0188(1070)}{0.2160(1070)}$	0.0221 (9070)	0.0221 (9070)	0.0220 (8070)	0.0200	
scene	$\frac{0.2100(1076)}{0.2418(1077)}$	0.2322 (4070)	0.2307 (4070)	0.2208 (3076)	0.2419	
yeast		0.2421 (10%)	0.2344(10%)	0.2455 (10%)	0.3029	
SUBSET 0/1 L	OSS	~				
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.9244 (10%)	0.9404 (90%)	0.9405 (90%)	0.9406(90%)	0.9392	
birds	0.9689 (90%)	$0.9736\ (90\%)$	0.9752~(90%)	0.9271~(10%)	0.9674	
CAL500	1.0000 (10%)	1.0000~(10%)	1.0000~(10%)	1.0000~(10%)	1.0000	
$\operatorname{Corel5k}$	0.9944 (70%)	0.9938 (80%)	0.9958~(80%)	0.9944~(80%)	0.9954	
emotions	$0.7825 \ (10\%)$	$0.7960\ (90\%)$	0.7859~(90%)	0.7925~(10%)	0.7943	
enron	$\overline{0.9865}$ (10%)	0.9977~(10%)	0.9959~(10%)	0.9982~(10%)	0.9994	
flagsml	$\overline{0.8711}$ (20%)	0.8911~(40%)	0.8555~(60%)	0.8708~(50%)	0.9642	
genbase	0.0997 (10%)	0.0997 $(10%)$	0.0997 (10%)	0.0997~(10%)	0.7220	
medical	$\overline{0.5460(10\%)}$	0.6259(20%)	0.6381(30%)	0.6208(20%)	0.7341	
scene	0.8322 (10%)	0.8367(90%)	0.8384(90%)	0.8243(90%)	0.8309	
veast	0.8974 (10%)	0.8854(10%)	0.9003(30%)	$\overline{0.8941(10\%)}$	0.9053	
EXAMPLE BA	SED ACCURAC	CY (inverted)	()	()		
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.7824 (10%)	0.8157 (90%)	0.8164 (90%)	0.8159(90%)	0.8126	
birds	0.8549(90%)	0.8679(90%)	0.8698(90%)	0.8431(10%)	0.8509	
CAL500	0.7727(20%)	0.7882(50%)	0.7823(20%)	$\overline{0.7732}$ (30%)	0.7971	
Corel5k	1000000000000000000000000000000000000	0.8530(90%)	0.8573(90%)	0.8535(90%)	0.8512	
emotions	0.4740 (80%)	0.4718(90%)	0.4716(90%)	0.4748(80%)	0.4708	
enron	0.6919 (10%)	0.8024 (10%)	0.7510(10%)	0.8028(10%)	0.8047	
flagsml	$\overline{0.4049}$ (20%)	0.4184(40%)	0.4105(60%)	0.4201(20%)	0.4962	
genbase	$\overline{0.0524(10\%)}$	0.0524(10%)	0.0524 (10%)	0.0524(10%)	0.6987	
medical	$\overline{0.3716(10\%)}$	$\overline{0.4443}$ (20%)	0.4524(20%)	0.4419(20%)	0.6294	
scene	$\frac{0.0110}{0.5394}$ (10%)	0.5494 (40%)	0.5459 (40%)	0.5270(60%)	0.5476	
veast	0.5399(10%)	0.5359(10%)	0.5656 (60%)	$\frac{0.0210}{0.5402}$ (10%)	0.5804	
BANKING LO	<u> </u>	0.0000 (1070)	0.0000 (0070)	0.0402 (1070)	0.0001	
Data Set	BB+InfoGain	Conv+InfoGain	LP+InfoGain	MLInfoGain	No Sel	
bibtex	0.0658 (10%)	0.0938 (20%)	0.1004 (50%)	0.0936 (20%)	0.1121	
birds	$\frac{0.1660}{0.1660}$	0.1751 (80%)	0.1728 (80%)	0.1240(10%)	0 1674	
CAI 500	0.1000(0070)	0.2178 (10%)	0.1120(0070)	$\frac{0.1240(10\%)}{0.2160(10\%)}$	0.1074	
CarolEl	0.2203 (1070) 0.1277 (20%)	0.2178 (1070) 0.1220 (20%)	0.2218 (1070) 0.1965 (20%)	$\frac{0.2109(1070)}{0.1026(2007)}$	0.3230	
Coreisk	0.1277 (3070)	0.1239(3076)	0.1203(3070)	$\frac{0.1230}{0.2020}$	0.1397	
ennorions	$\begin{array}{c c} 0.2041 (1070) \\ 0.1997 (1007) \end{array}$	0.2014 (9070) 0.1073 (1007)	0.2070 (9070) 0.1505 (1002)	0.2020(1070)	$\frac{0.1981}{0.9970}$	
fageml	$\frac{0.1337(1070)}{0.1826(0007)}$	0.1919 (1070) 0.1919 (1070)	0.1080 (1070)	0.1909 (1070)	0.2010	
nagsmi	$\frac{0.1000 (20\%)}{0.0079 (1007)}$	0.2033 (40%)	0.0029 (1002)	0.1911 (30%)	0.2904	
genbase	$\frac{0.0073 (10\%)}{0.0417 (10\%)}$	$\frac{0.0073}{0.0073}$ (10%)	$\frac{0.0073}{0.0874}$ (10%)	$\frac{0.0073}{0.0859}$ (10%)	0.1810	
medical		$\frac{0.0351}{0.0351}$ (10%)	0.0354 (10%)	0.0352 (10%)	0.1306	
scene	$\frac{0.1347}{0.0104}$	0.1562 (40%)	0.1541 (40%)	0.1512(50%)	0.1857	
yeast	0.2194 (10%)	0.2264 (10%)	0.2180(10%)	$0.2273\ (10\%)$	0.2577	
Best values	24	10	8	13	5	
(underlined)						
\geq basenne score (bold)	37	30	31	37		
(DOID)						

Table 7.7: Best results achieved with the BR + NaiveBayes classifier

HAMMING LOSS						
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.0130 (10%)	0.0134 (10%)	0.0145 (50%)	0.0134(10%)	0.0146	
birds	$\overline{0.0440(10\%)}$	0.0485(30%)	0.0484(60%)	0.0478(20%)	0.0499	
CAL500	$\frac{1}{0.1652}$ (10%)	0.1636 (10%)	0.1652(10%)	0.1616(10%)	0.1761	
Corel5k	0.0095 (10%)	0.0096 (10%)	0.0095 (10%)	$\frac{0.0096}{0.0096}$ (10%)	0.0099	
emotions	$\frac{0.0000(10\%)}{0.2434(20\%)}$	0.0000 (10%) 0.2401 (10%)	$\frac{0.0000}{0.2440}$ (10%)	0.0000 (10%) 0.2291 (10%)	0.2550	
eniorions	0.2434 (2070)	0.2401 (1070)	0.2440 (1070) 0.0522 (60%)	$\frac{0.2291}{0.0530}$ (1070)	0.2550	
fageml	$\frac{0.0516}{0.2545}$ (4070)	0.0000 (90%)	0.0522 (0070)	0.0000 (0070)	0.0525	
11ag 51111	0.2343(9070)	0.2700 (9070)	$\frac{0.2557}{0.0011}$ (10%)	0.2008 (2070)	0.2085	
genbase	$\frac{0.0011(10\%)}{0.005(20\%)}$	$\frac{0.0011}{0.0006}$ (10%)	$\frac{0.0011}{0.0100}$ (10%)	$\frac{0.0011}{0.0006}$ (10%)	0.0011	
medical	$\frac{0.0095(20\%)}{0.1996(60\%)}$	0.0090 (10%)	0.0100 (10%)	0.0090 (10%)	0.0102	
scene	$\frac{0.1380\ (00\%)}{0.0505\ (00\%)}$	0.1417 (60%)	0.1420(60%)	0.1412(70%)	0.1444	
yeast		0.2504(30%)	0.2633(80%)	0.2549(30%)	0.2682	
SUBSET 0/1 L		a II.co.			NO	
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	$\frac{0.8245(10\%)}{0.1750(10\%)}$	0.8421 (20%)	0.8494(70%)	0.8403(20%)	0.8546	
birds	0.4758(10%)	0.5082(60%)	0.5022(50%)	0.4947(20%)	0.5208	
CAL500	1.0000 (10%)	1.0000 (10%)	1.0000 (10%)	1.0000 (10%)	1.0000	
Corel5 k	0.9900 (30%)	0.9922~(80%)	0.9952~(90%)	$0.9920 \ (70\%)$	0.9928	
emotions	0.7470 (20%)	0.7303~(10%)	0.7489~(60%)	$0.7336\ (10\%)$	0.7522	
enron	$0.8596 \ (10\%)$	0.8696~(90%)	0.8579~(40%)	0.8672~(80%)	0.8731	
$_{ m flagsml}$	0.7008 (10%)	0.7416~(90%)	$\overline{0.7166\ (10\%)}$	0.7255~(80%)	0.7574	
genbase	$\overline{0.0287}$ (10%)	0.0287~(10%)	0.0287~(10%)	0.0287~(10%)	0.0287	
medical	0.3017 (10%)	0.2997 (10%)	0.3068~(10%)	0.2946~(10%)	0.3222	
scene	0.4454(60%)	0.4549(60%)	0.4545(60%)	0.4570(70%)	0.4624	
yeast	$\overline{0.8366}$ (30%)	0.8324(20%)	0.8544(70%)	0.8357(20%)	0.8469	
EXAMPLE BA	SED ACCURAC	CY (inverted)	· · · · ·	. ,	1	
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.6791 (20%)	0.6901 (40%)	0.6961 (70%)	0.6919 (80%)	0.6980	
birds	$\overline{0.4035}$ (10%)	0.4233 (50%)	0.4136(50%)	0.4188(40%)	0.4322	
CAL500	$\overline{0.7633}(50\%)$	0.7727(90%)	0.7695(60%)	0.7683 (90%)	0.7706	
Corel5k	0.9417 (90%)	0.9387(90%)	0.9432(90%)	0.9380(90%)	0.9379	
emotions	0.5059 (20%)	0.5068(10%)	0.5182(60%)	0.4936(10%)	0.5297	
enron	0.5665(30%)	0.5829(90%)	0.5605(70%)	0.5776(90%)	0.5767	
flagsml	0.3880 (10%)	0.4076(90%)	0.3855(70%)	0.3905(30%)	0.4081	
genbase	0.0138(10%)	0.0138(10%)	$\frac{0.0138}{0.0138}$ (10%)	0.0138(10%)	0.0138	
medical	$\frac{0.0200(20\%)}{0.2204(20\%)}$	$\frac{0.0200}{0.2211}$ (10%)	$\frac{0.0200(10\%)}{0.2287(10\%)}$	$\frac{0.2190}{0.2190}$ (10%)	$\frac{0.0100}{0.2419}$	
scene	0.3947 (60%)	0.4076 (60%)		$\frac{0.2100}{0.4051}$ (70%)	0.4134	
veset	$\frac{0.5011}{0.5668}$ (70%)	0.5671 (30%)	0.4000 (0070) 0.5732 (00%)	0.5660 (50%)	0.5720	
PANKINC IO	<u> </u>	0.0011 (0070)	0.0102 (0070)	0.0000 (0070)	0.0120	
Data Set	BB+InfoGain	Conv+InfoGain	LP⊥InfoGain	MLInfoGain	No Sel	
bibtex	0.1632 (90%)	0.1660 (90%)	0.1623 (90%)	0.1664 (90%)	0.1646	
birds	0.1411 (10%)	0.1385 (40%)	$\frac{0.1454}{0.1454}$ (50%)	0.1482(20%)	0 1631	
CAL500		$\frac{0.1000}{0.2894}$ (10%)	0.2060 (10%)	0.2804 (10%)	0.3638	
Corol51	0.3003 (1070) 0.1789 (2007)	$\frac{0.2034}{0.1778}$ (10%)	0.2303 (1070) 0.1999 (1077)	$\frac{0.2094}{0.1791}$ (10%)	0.3030	
omotions	0.1100 (00/0)	$\frac{0.1110}{0.2484}$ (107)	0.1020 (1070)	0.1101 (10/0)	0.1003	
ennotions	$0.2000 (1070) \\ 0.1440 (1007)$	U.2404 (1070)	0.2000 (1070)	$\frac{0.2230}{0.1916}$ (1070)	0.3000	
enron de gamel	0.1449 (10%)	0.1320 (10%)	0.10(0.00%)	$\frac{0.0280}{0.1910}$ (10%)	0.1/19	
nagsim		$\frac{0.2050}{0.0000}$ (20%)	0.2589 (20%)	0.2589(20%)	0.2898	
genbase	$\frac{0.0028 (10\%)}{0.0711 (22\%)}$	$\frac{0.0028 (10\%)}{0.0018 (70\%)}$	$\frac{0.0028 (10\%)}{0.0011 (70\%)}$	$\frac{0.0028 (10\%)}{0.0724 (10\%)}$	$\frac{0.0028}{0.0010}$	
medical	$\frac{0.0711}{0.0711}$ (20%)	0.0812 (70%)	0.0811 (50%)	0.0794 (10%)	0.0812	
scene	0.2411 (60%)	0.2394 (60%)	0.2433(60%)	0.2356 (90%)	0.2489	
yeast	0.2758 (10%)	0.2644 (10%)	$0.2814\ (10\%)$	$0.2738\ (10\%)$	0.3238	
Best values	21	13	11	15	6	
(underlined)						
\geq baseline score	43	38	40	40		
(ΔΟΙΔ)						

Table 7.8: Best results achieved with the CC + DecisionTree classifier

HAMMING LOSS						
Data Set	BR+InfoGain	nfoGain Copy+InfoGain LP+InfoGain MLInfoGain				
bibtex	0.1512 (90%)	0.1521 (90%)	0.1535~(90%)	0.1521 (90%)	<u>0.1419</u>	
birds	$0.2474 \ (20\%)$	0.3638~(90%)	0.3638~(90%)	0.1795~(10%)	0.3589	
CAL500	0.3946~(80%)	0.3799~(70%)	0.3788~(90%)	$\overline{0.3743} (90\%)$	0.3940	
$\operatorname{Corel5k}$	0.0212~(90%)	$0.0219\ (90\%)$	0.0227~(90%)	0.0216 (90%)	0.0175	
emotions	0.2501 (90%)	0.2484~(90%)	0.2518~(90%)	0.2515~(90%)	0.2470	
enron	0.1205~(10%)	0.1934~(10%)	0.1668~(10%)	0.1909~(10%)	0.2142	
flagsml	$\overline{0.2469} (20\%)$	0.2756~(40%)	0.2505~(50%)	0.2521~(30%)	0.3046	
genbase	$\overline{0.0035} (20\%)$	0.0035~(20%)	0.0035~(20%)	0.0035~(20%)	0.0336	
medical	$\overline{0.0196}$ (10%)	$\overline{0.0221} \ (90\%)$	$\overline{0.0221} \ (90\%)$	$\overline{0.0221} (90\%)$	0.0249	
scene	0.2023 (10%)	0.2271 (40%)	0.2250 (40%)	0.2161 (50%)	0.2368	
yeast	$\overline{0.2815(10\%)}$	0.2831 (10%)	0.2755(20%)	0.2840(10%)	0.3039	
SUBSET 0/1 L	OSS		<u>`````````````````````````````````</u>			
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.9247 (10%)	0.9375 (10%)	0.9375 (80%)	0.9386(90%)	0.9378	
birds	0.9689 (90%)	0.9736(90%)	0.9736(90%)	0.9271 (10%)	0.9674	
CAL500	1.0000 (10%)	1.0000(10%)	1.0000(10%)	1.0000(10%)	1.0000	
Corel5k	$\frac{(73)}{0.9900(90\%)}$	$\overline{0.9886}$ (80%)	0.9918(90%)	$\overline{0.9892}$ (80%)	0.9914	
emotions	0.7823(20%)	$\frac{0.7825}{0.7825}$ (90%)	0.7724 (90%)	0.7842(90%)	0.7740	
enron	0.9612(10%)	0.9947 (10%)	$\frac{0.9941}{0.9941}$ (10%)	0.9947 (10%)	0.9988	
flagsml	$\frac{0.0012}{0.8350}(50\%)$	0.8753 (40%)	0.8453 (60%)	0.8400(50%)	0.9284	
gonbaso	$\frac{0.0000}{0.0785}$ (20%)	0.0785 (20%)	0.0785(20%)	0.0785(20%)	0.7175	
modical	$\frac{0.0783}{0.5300}$ (20%)	$\frac{0.0760}{0.6422}$ (20%)	$\frac{0.0783}{0.6524}$ (20%)	$\frac{0.0783}{0.6361}$ (20%)	0.7170	
Gaopo	$\frac{0.3333}{0.7006}$ (10%)	0.0422 (2070)	0.0024 (2070)	0.0301 (2070) 0.0172 (0.0%)	0.1210	
scene	$\frac{0.7900(10\%)}{0.8862(80\%)}$	0.8320 (9070)	0.0330(9070)	0.8172(90%)	0.0247	
	$\frac{0.0002}{\text{CED}}$	$\frac{0.6904}{3076}$	0.8377(2076)	0.8871 (7076)	0.8910	
Data Sot	BB InfoCain	Conv InfoC oin	I D InfoC ain	MUnfoChin	No Sol	
bibtev	0.8364 (90%)	$\frac{0.8381}{(90\%)}$	$\frac{11 + 1110 \text{Gam}}{0.8382 (90\%)}$	0.8370 (00%)	0.8335	
birds	0.8546 (90%)	0.8680 (90%)	0.8502 (90%) 0.8694 (90%)	0.8319(3070)	0.8508	
CAL500	0.8223(70%)	0.8203 (50%)	0.8166 (90%)	$\frac{0.8159}{0.8159}$ (90%)	0.8216	
Corel5k	0.8629 (90%)	0.8566 (90%)	0.8623 (90%)	$\frac{0.02100}{0.8565}$ (90%)	0.8541	
emotions	0.4680(90%)	0.4660 (90%)	0.4680(90%)	0.4674(90%)	$\frac{0.0011}{0.4636}$	
enron	0.6651 (10%)	0.7918 (10%)	0.7514(10%)	0.7911 (10%)	0.8006	
flagsml	$\overline{0.3913(20\%)}$	0.4166(30%)	0.4070(40%)	0.4005(30%)	0.4699	
genbase	$\overline{0.0443(20\%)}$	0.0443 (20%)	0.0443 (20%)	0.0443(20%)	0.6935	
medical	$\overline{0.3825(10\%)}$	$\overline{0.4439}$ (10%)	0.4574(10%)	0.4455(10%)	0.6247	
scene	$\frac{1}{0.5203}$ (10%)	0.5444(40%)	0.5408(40%)	0.5218(60%)	0.5425	
veast	$\frac{1}{0.5762}$ (80%)	0.5822(90%)	0.5721 (60%)	0.5780(70%)	0.5816	
BANKING LO	SS					
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.1256 (10%)	0.1319 (80%)	0.1340 (90%)	0.1314 (80%)	0.1333	
birds	$\overline{0.1658(90\%)}$	0.1763(80%)	0.1742(80%)	0.1290(10%)	0.1674	
CAL500	0.4265 (80%)	0.4173(70%)	0.4138(90%)	$\overline{0.4100(90\%)}$	0.4271	
Corel5k	0.1575 (90%)	0.1571 (90%)	0.1576(90%)	$\frac{0.1570}{0.1570}$ (90%)	0.1585	
emotions	0.2019(90%)	0.2022(90%)	0.2066 (90%)	$\frac{0.1979}{0.1979}$ (70%)	0 1990	
enron	0.1526 (10%)	0.2208 (10%)	0.1937 (10%)	$\frac{1.2010}{0.2208}$ (10%)	0.2374	
flagsml	$\frac{0.1020(10\%)}{0.1925(20\%)}$	0.2184 (20%)	0.2046 (40%)	0.2031(30%)	0.2827	
genhase	$\frac{3.1020}{0.0071}$ (10%)	0.0071 (10%)	0.0071 (10%)	0.0071 (10%)	0 1786	
medical	$\frac{3.0011}{0.0440}$ (10%)	$\frac{0.00011}{0.0408}$ (10%)	$\frac{0.0411}{0.0411}$ (10%)	$\frac{0.0403}{0.0403}$ (10%)	0 1330	
scono	0 1287 (20%)	0.1531 (40%)	0.1500 (1070)	$\frac{0.0400}{0.1458}$ (50%)	0.1900	
voset	$\left \frac{0.1201}{0.2694} (10\%) \right $	0.1001 (4070) 0.2720 (10%)	0.1000 (4070) 0.2677 (60%)	0.1400 (0070)	0.1000	
Best values	0.200 + (10/0)	0.2120 (1070)	<u>5.2011 (0070)</u>	0.2100 (1070)	0.2111	
(underlined)	22	6	10	15	7	
\leq baseline score						
(bold)	32	29	30	36		

Table 7.9: Best results achieved with the $\rm CC$ + NaiveBayes classifier

HAMMING LOSS						
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	No Sel.		
bibtex	0.0154~(20%)	0.0157~(20%)	0.0159~(50%)	0.0156~(20%)	0.0162	
birds	$\overline{0.0465}$ (30%)	0.0493~(90%)	0.0488~(80%)	0.0476~(10%)	0.0494	
CAL500	$\overline{0.2255}$ (10%)	0.2273 $(10%)$	0.2253 $(10%)$	0.2252 $(10%)$	0.2309	
Corel5 k	0.0147 (10%)	0.0169(10%)	0.0189(10%)	0.0165(10%)	0.0224	
emotions	$\overline{0.1816}$ (80%)	0.1867(80%)	0.1869 (90%)	0.1802 (90%)	0.1883	
enron	0.0537 (10%)	0.0558 (90%)	0.0534(10%)	$\overline{0.0547}$ (70%)	0.0564	
flagsml	0.2408(40%)	0.2568(40%)	0.2501(70%)	0.2499(40%)	0.2703	
genbase	$\overline{0.0029(10\%)}$	0.0029(10%)	0.0029(10%)	0.0029(10%)	0.0029	
medical	$\overline{0.0147(10\%)}$	$\overline{0.0185(40\%)}$	0.0182(10%)	$\overline{0.0180(30\%)}$	0.0189	
scene	0.0887 (90%)	0.0854(90%)	0.0880(90%)	0.0841(90%)	0.0834	
yeast	0.1924 (50%)	0.1930(90%)	0.1940 $(90%)$	0.1923(80%)	0.1934	
SUBSET 0/1 L	OSS		<u>, </u>		1	
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.8729 (10%)	0.8948 (10%)	0.9097 (50%)	0.8921 (20%)	0.9159	
birds	0.4976(60%)	0.5070(80%)	0.5086(80%)	0.4945(30%)	0.5193	
CAL500	1.0000 (10%)	1.0000(10%)	1.0000(10%)	1.0000(10%)	1.0000	
Corel5 k	0.9976 (90%)	0.9982(70%)	0.9994 (90%)	0.9980 (80%)	0.9972	
emotions	0.6580 (80%)	0.6715(80%)	0.6765(40%)	0.6613(90%)	0.6798	
enron	0.8949(10%)	0.9043 (70%)	0.9002 $(30%)$	0.8966~(60%)	0.9348	
flagsml	$\overline{0.7934}$ (40%)	0.8342 $(40%)$	0.7990 (70%)	0.7990 $(50%)$	0.8555	
genbase	$\overline{0.0483}$ (10%)	0.0483~(10%)	0.0483~(10%)	0.0483~(10%)	<u>0.0483</u>	
medical	$\overline{0.4233}$ (10%)	0.5092 (40%)	0.4980(50%)	0.5112 (30%)	0.5267	
scene	0.3677 (90%)	0.3540(90%)	0.3623(90%)	0.3498(80%)	0.3440	
yeast	0.7948 (80%)	0.7981 $(90%)$	0.8022 $(90%)$	0.7919(70%)	0.7964	
EXAMPLE BA	SED ACCURAC	CY (inverted)				
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	$0.7381 \ (10\%)$	$0.7814\ (10\%)$	0.8094~(50%)	$0.7818\ (20\%)$	0.8192	
birds	$\overline{0.4123} \ (90\%)$	0.4276~(90%)	0.4228~(90%)	0.4096~(80%)	0.4314	
CAL500	0.8024 (90%)	0.8033~(70%)	0.8006~(30%)	$\overline{0.8007} (10\%)$	0.8051	
Corel5 k	0.9646~(50%)	0.9663~(40%)	0.9714(90%)	0.9649~(40%)	0.9676	
emotions	$\overline{0.4314}$ (80%)	0.4396~(80%)	0.4478~(60%)	0.4335~(90%)	0.4482	
enron	$\overline{0.6186}$ (10%)	0.6593~(70%)	0.6293~(10%)	$0.6356\ (70\%)$	0.6871	
flagsml	$\overline{0.3715}$ (40%)	0.3895~(40%)	0.3799~(70%)	0.3802~(40%)	0.4117	
genbase	$\overline{0.0242} \ (10\%)$	0.0242~(10%)	0.0242~(10%)	0.0242~(10%)	<u>0.0242</u>	
medical	$\overline{0.3263}$ (10%)	$\overline{0.4028}$ (10%)	$\overline{0.3923}$ (50%)	$\overline{0.4043}$ (30%)	0.4116	
scene	0.3306 (90%)	0.3192(90%)	0.3279(90%)	0.3148(90%)	<u>0.3098</u>	
yeast	0.4742 (80%)	0.4749(90%)	0.4807 (90%)	$0.4728\ (80\%)$	0.4758	
RANKING LO	SS					
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.1388~(10%)	0.1574~(10%)	0.1671~(40%)	0.1575~(10%)	0.1679	
birds	0.0834~(10%)	0.0864~(50%)	0.0866~(50%)	0.0879~(40%)	0.0902	
CAL500	0.3193 (90%)	0.3226~(90%)	0.3223~(30%)	0.3206~(80%)	0.3185	
$\operatorname{Corel5k}$	$0.2522 \ (10\%)$	0.2687~(90%)	0.2648~(90%)	0.2682~(90%)	0.2600	
emotions	0.1496~(50%)	0.1507~(60%)	0.1509~(80%)	0.1493~(90%)	0.1496	
enron	0.1003~(20%)	$0.1027 \ (30\%)$	0.1016~(10%)	$\overline{0.1019}$ (70%)	0.1059	
$_{ m flagsml}$	0.1894 (40%)	0.1911~(50%)	0.1879~(80%)	0.1848 (60%)	0.2036	
genbase	0.0053 (10%)	0.0053~(10%)	0.0053~(10%)	$\overline{0.0053~(10\%)}$	<u>0.0053</u>	
medical	0.0474(10%)	$\overline{0.0611}$ (30%)	0.0627 (60%)	0.0603 (20%)	0.0655	
scene	0.0782 (90%)	0.0766 (90%)	0.0782 (90%)	0.0769(90%)	<u>0.0760</u>	
yeast	0.1626 (70%)	0.1638 $(90%)$	0.1648 $(90%)$	0.1633 $(80%)$	0.1635	
Best values		K.	7	14	11	
(underlined)	Z1	ə	1	14	11	
\leq baseline score	38	34	31	37		
(bold)	00	04	01	01		

Table 7.10: Best results achieved with the IBLR-ML classifier

HAMMING LO	OSS				
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.0192 (10%)	0.0205 (80%)	0.0205 (80%)	0.0204 (70%)	0.0205
birds	0.0542(10%)	0.0552 (10%)	0.0561 (30%)	0.0535(30%)	0.0586
CAL500	0.1990 (20%)	0.1961(40%)	0.1991(50%)	0.1992(90%)	0.1997
Corel5 k	0.0162(20%)	$\overline{0.0162}$ (10%)	0.0167(10%)	0.0161(10%)	0.0168
emotions	0.2593 (70%)	0.2561(40%)	0.2546(30%)	0.2597(20%)	0.2778
enron	0.0669 (10%)	0.0699(20%)	$\overline{0.0683(10\%)}$	0.0696(10%)	0.0717
flagsml	$\overline{0.2646(40\%)}$	0.2969(90%)	0.2681(60%)	0.2664(30%)	0.2931
genbase	$\overline{0.0019(10\%)}$	0.0019 (10%)	0.0019(10%)	0.0019(10%)	0.0019
medical	$\overline{0.0116(10\%)}$	$\overline{0.0127}$ (10%)	0.0132(10%)	0.0125(10%)	0.0135
scene	$\overline{0.1413}(60\%)$	0.1421 (70%)	0.1420(70%)	0.1404(60%)	0.1437
veast	0.2720 (60%)	0.2731(40%)	0.2805(80%)	0.2727(20%)	0.2779
SUBSET 0/1 L	OSS	. ,	. ,	. ,	
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.8366 (10%)	0.8541 (40%)	0.8563 (60%)	0.8541(40%)	0.8544
birds	$\frac{1}{0.5053}$ (10%)	0.5085 (10%)	0.4974(10%)	0.5020(30%)	0.5254
CAL500		1,0000,(10%)	$\frac{0.1011}{1.0000}$ (10%)	1,0000 (10%)	1 0000
Corel5k	$\frac{1.0000(10\%)}{0.9782(20\%)}$	$\frac{1.0000}{0.9822}$ (10%)	$\frac{1.0000}{0.9856}$ (80%)	$\frac{1.0000}{0.9812}$ (10%)	$\frac{1.0000}{0.9868}$
emotions	$\frac{0.7590}{0.7590}$ (60%)	0.7455 (20%)	0.7605(30%)	0.7456 (60%)	0 7944
enron	0.8643 (10%)	$\frac{0.1100}{0.8843}$ (30%)	0.8726 (30%)	0.8796(30%)	0.8913
flagsml	$\frac{0.0010(10\%)}{0.7263(50\%)}$	0.7537 (90%)	0.7371 (10%)	0.7324(10%)	0 7532
genbase	$\frac{0.1200(0070)}{0.0272(10\%)}$	0.0272 (10%)	0.0272 (10%)	0.0272 (10%)	0.0272
medical	$\frac{0.0212}{0.3068}$ (10%)	$\frac{0.0212}{0.3180}$ (10%)	$\frac{0.0212}{0.3241}$ (10%)	$\frac{0.0212}{0.3129}$ (10%)	0.3375
scene	$\frac{0.0000}{0.4512}$ (60%)	0.4512 (70%)	0.4520(70%)	0.4474 (60%)	0.4529
veast	0.4012 (00%) 0.8581 (20%)	0.8556 (30%)	0.4626 (10%) 0.8635 (90%)	$\frac{0.1111}{0.8527}$ (30%)	0.1625
	SED ACCURAC	TV (invented)	0.0000 (0070)		0.0011
Data Set	BB+InfoGain	Conv+InfoGain	LP+InfoGain	MLInfoGain	No Sel
hibter	0.7105 (10%)	0.7378(80%)	0.7377 (90%)	0.7368(80%)	0 7361
birds	$\frac{0.1100(10\%)}{0.4026(10\%)}$	0.4107 (10%)	0.4085 (30%)	0.4079(30%)	0.4266
CAL500	$\frac{0.1020(10\%)}{0.7931(20\%)}$	0.7885 (40%)	0.7937 (70%)	0.7934 (90%)	0 7960
Corel5k	0.9106 (20%)	$\frac{0.9169}{0.9169}$ (10%)	0.9238 (80%)	0.9165(10%)	0.9250
emotions	$\frac{0.5276}{0.5276}$ (70%)	0.5193(20%)	0.5238(30%)	0.5280(60%)	0.5631
enron	0.6217 (10%)	$\frac{0.6564}{0.6564}$ (70%)	0.6356 (60%)	0.6474 (80%)	0.6575
flagsml	$\frac{0.0211}{0.4070}$ (40%)	0.4401 (90%)	0.4076 (60%)	0.4075(30%)	0 4347
genhase	$\frac{0.1070(10\%)}{0.0174(10\%)}$	0.0174 (10%)	0.1070(00%)	0.174(10%)	0.1511
medical	$\frac{0.0114}{0.2349}$ (10%)	$\frac{0.0114}{0.2505}$ (10%)	$\frac{0.0114}{0.2597}$ (10%)	$\frac{0.0114}{0.2455}$ (10%)	$\frac{0.0111}{0.2645}$
scene	$\frac{0.2043(10\%)}{0.4043(60\%)}$	0.2000 (10%) 0.4058 (70%)	0.2031 (10%) 0.4078 (70%)	0.2400(10%)	0.2040
veast	0.1010(0070) 0.5777 (60%)	0.5774 (40%)	0.5891 (80%)	$\frac{0.1021}{0.5781}$ (20%)	0.5862
BANKING LO			0.0001 (0070)	0.0101 (2070)	0.0002
Data Set	BB+InfoGain	Conv+InfoGain	LP+InfoGain	MLInfoGain	No Sel
hibtex	0.4117 (80%)	0.4126 (90%)	0.4126 (80%)	0.4127 (80%)	0 4148
birds	$\frac{0.2367}{0.2367}$	0.2341 (50%)	0.2326 (50%)	0.2355(30%)	0.2365
CAL500	0.6506 (10%)	0.6543 (30%)	$\frac{0.2520}{0.6540}$ (20%)	0.6521 (80%)	0.6576
Corel5k	$\frac{0.00000}{0.7531}$ (80%)	0.7484 (90%)	0.7491 (90%)	0.7478 (90%)	0 7502
emotions	0.3191 (50%)	0.3156 (20%)	0.3066 (40%)	$\frac{0.1416}{0.3206}$ (50%)	0.3442
enron	0.5406 (40%)	0.5595(90%)	$\frac{0.5000}{0.5272}$ (60%)	0.5255(90%) 0.5454(90%)	0.5488
flagsml	0.0400(40%)	0.5000 (90%) 0.5013 (90%)	$\frac{0.0212}{0.5013}$ (00%)	0.5434 (0070)	0.0100
genhase	$\frac{312001}{0.0076}$ (10%)	0.0076 (10%)	0.0076 (10%)	0.0076 (10%)	0.1076
medical	$\frac{0.0010(1070)}{0.1208(10\%)}$	$\frac{0.0010(1070)}{0.1313(10\%)}$	$\frac{0.0010}{0.1360}$ (10%)	$\frac{0.0010}{0.1278}$ (10%)	0 1303
scono	$\begin{array}{c c} 0.1230 (1070) \\ 0.2007 (0.0%) \end{array}$	0.1010 (1070)	0.1008 (1070)	$\frac{0.1210}{0.2133}$ (60%)	0.1000
veset	$\frac{0.2091}{0.3949}$	0.2103 (0070)	0.2100 (1070) 0.3996 (90%)	0.2100 (0070) 0 3070 (50%)	0.2120
Beet values	0.0040 (0070)	0.0002 (4070)	0.0000 (0070)	0.0343 (0070)	0.0000
(underlined)	26	10	10	14	5
\leq baseline score					
	42	38	38	41	

Table 7.11: Best results achieved with the LP $\,+$ DecisionTree classifier

HAMMING LOSS						
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.0163 (10%)	0.0186~(20%)	0.0199~(20%)	$0.0186\ (10\%)$	0.0228	
birds	0.0507 (20%)	0.0544 (70%)	0.0551~(60%)	0.0517~(10%)	0.0552	
CAL500	$\overline{0.1947}$ (90%)	0.1963(40%)	0.1968(90%)	0.1947(60%)	0.1975	
Corel5 k	$\overline{0.0158(80\%)}$	0.0160(60%)	0.0150(70%)	$\overline{0.0160(60\%)}$	0.0162	
emotions	0.2069 (70%)	0.2047(70%)	0.2061(50%)	0.2046(70%)	0.2050	
enron	0.0616 (10%)	0.0670(10%)	0.0627(10%)	$\overline{0.0656}$ (70%)	0.0681	
flagsml	$\overline{0.2652}$ (50%)	0.2798(20%)	0.2703(50%)	0.2645(40%)	0.2941	
genbase	0.0049 (10%)	0.0049(10%)	0.0049(10%)	0.0049(10%)	0.0049	
medical	$\overline{0.0153(10\%)}$	0.0174(10%)	0.0174(10%)	0.0171(10%)	0.0196	
scene	0.0954 (90%)	0.0955(90%)	0.0972(90%)	0.0929(80%)	0.0931	
yeast	0.2170 (80%)	0.2171(60%)	0.2190(80%)	0.2169(90%)	0.2188	
SUBSET 0/1 L	OSS	. ,	. ,		I	
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.8108 (10%)	0.8400 (10%)	0.8809 (30%)	0.8384(10%)	0.9252	
birds	$\overline{0.4867(80\%)}$	0.4914 (70%)	0.4975 (70%)	0.4929(80%)	0.5007	
CAL500	$\frac{1.0000}{1.0000}$	1.0000 (10%)	1.0000 (10%)	1.0000 (10%)	1 0000	
Corel5k	$\frac{1.0000}{0.9766}$ (90%)	$\frac{1.0000}{0.9770}$ (60%)	$\frac{1.0000}{0.9790}$ (80%)	$\frac{1.0000}{0.9762}$ (40%)	$\frac{1.0000}{0.9802}$	
emotions	0.6596 (20%)	0.6647 (70%)	0.6562 (80%)	$\frac{0.0102}{0.6495}$ (70%)	0.6648	
enron	0.0000 (2070) 0.8540 (10%)	0.8784 (80%)	0.0502 (0070) 0.8506 (10%)	$\frac{0.0400}{0.8602}$ (50%)	0.0040	
fageml	$\frac{0.0543(1070)}{0.7568(40\%)}$	0.0704(0070)	0.3330 (10%)	0.3002 (30%)	0.8145	
nagsiin	0.7508 (4070)	0.7932 (9070)	$\frac{0.7413}{0.0945}$ (10%)	0.7313(1070)	0.0145	
genbase	$\frac{0.0843 (10\%)}{0.2057 (10\%)}$	$\frac{0.0843}{0.4977}$ (10%)	$\frac{0.0845}{0.4326}$ (10%)	$\frac{0.0843(10\%)}{0.4965(10\%)}$	$\frac{0.0843}{0.4765}$	
medical	$\frac{0.3937(10\%)}{0.3159(0007)}$	0.4377(1070)	0.4330 (10%)	0.4203 (10%)	0.4700	
scene		0.3141(90%)	0.3174(90%)	$\frac{0.3062}{0.5452}$ (80%)	0.3062	
yeast	0.7489 (70%)	0.7505 (90%)	0.7551(90%)	0.7456(80%)	0.7551	
EXAMPLE BA	SED ACCURAC	CY (inverted)				
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	$\frac{0.6890\ (10\%)}{0.000}$	0.7407 (10%)	0.7904(30%)	0.7404(10%)	0.8502	
birds		0.4163(70%)	0.4240(70%)	$\frac{0.4118}{0.500}$	0.4292	
CAL500	$\frac{0.7824 \ (90\%)}{(90\%)}$	0.7843(40%)	0.7857 (70%)	0.7830(60%)	0.7866	
$\operatorname{Corel5k}$	0.9129 (20%)	0.9132 (30%)	0.9230(50%)	0.9105(30%)	0.9228	
emotions	0.4282 (70%)	0.4293~(70%)	0.4270~(50%)	0.4211 (70%)	0.4268	
enron	$0.6440 \ (10\%)$	0.7204~(10%)	0.6614~(10%)	$0.7019\ (70\%)$	0.7418	
flagsml	$0.3931 \ (50\%)$	0.4093~(20%)	0.3990~(50%)	0.3945~(40%)	0.4426	
genbase	0.0502~(10%)	0.0502~(10%)	0.0502~(10%)	0.0502~(10%)	<u>0.0502</u>	
medical	$\overline{0.3123}$ (10%)	$\overline{0.3598} (10\%)$	$\overline{0.3572} (10\%)$	$\overline{0.3516~(10\%)}$	0.3986	
scene	0.2822 (90%)	0.2817~(90%)	0.2861~(90%)	0.2747~(80%)	0.2751	
yeast	0.4773 (80%)	0.4789~(90%)	0.4791~(80%)	$\overline{0.4771} \ (90\%)$	0.4806	
RANKING LO	ss					
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.5686~(20%)	0.5741~(10%)	0.5805~(10%)	$0.5741 \ (10\%)$	0.5842	
birds	$\overline{0.3437}$ (10%)	0.3469~(40%)	0.3456~(40%)	0.3531~(90%)	0.3513	
CAL500	0.6899(70%)	0.6892 (20%)	0.6867 (10%)	0.6922 (70%)	0.6963	
Corel5 k	0.7623 (80%)	0.7617(80%)	0.7593(30%)	0.7618(80%)	0.7646	
emotions	0.6045 (40%)	0.6239(60%)	$\overline{0.6059}$ (40%)	0.6101(70%)	0.6388	
enron	$\overline{0.6458(60\%)}$	0.6167 (70%)	0.6323(50%)	0.6336(60%)	0.6485	
flagsml	0.7504 (80%)	$\overline{0.7532}$ (60%)	0.7453(10%)	0.7369(80%)	0.7547	
genbase	0.0425 (10%)	0.0425 (10%)	0.0425 (10%)	$\frac{1}{0.0425}$ (10%)	0.0425	
medical	$\frac{3.3428}{0.3428}$ (10%)	$\frac{0.0120(1070)}{0.4150(10\%)}$	$\frac{0.4200}{0.4200}$ (10%)	$\frac{0.4131}{0.4131}$ (10%)	$\frac{0.0120}{0.4650}$	
scene	$\frac{3.0120(1070)}{0.2714(70\%)}$	0.2660 (70%)	0.2665 (80%)	0 2653 (60%)	0.1000	
veast	0.2114(1070) 0.6501(20%)	0.2000 (1070)	0.2000 (0070) 0.6544 (00%)	$\frac{5.2000}{0.6476}$ (20%)	0.2740	
Rest values	3.0001 (2070)	000111 (0070)	0.0011 (00/0)	510110 (2070)	0.0002	
(underlined)	23	7	9	21	6	
\leq baseline score						
(bold)	39	39	36	43		

Table 7.12: Best results achieved with the LP + K-NN classifier

HAMMING LO	DSS				
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.0155~(10%)	0.0158~(10%)	0.0161 (30%)	$0.0158\ (10\%)$	0.0172
birds	0.0766 (90%)	0.0787(80%)	0.0791 ($80%$)	0.0769(90%)	0.0768
CAL500	$\overline{0.1948}$ (70%)	0.1955(70%)	0.1971(90%)	0.1961(10%)	0.1972
Corel5k	0.0161(90%)	0.0160(10%)	0.0163(90%)	0.0160(10%)	0.0160
emotions	0.2373(80%)	$\frac{0.2351}{0.2351}$	0.2334(90%)	$\frac{0.0100(10\%)}{0.2317(70\%)}$	0.2326
enron	0.0559 (20%)	0.0604 (90%)	0.0573 (40%)		0.0589
flagsml	$\frac{0.0000}{0.2596}$ (30%)	0.2853(40%)	0.2494 (10%)	0.2522(20%)	0.3054
gonbaso	0.2000 (0070) 0.0052 (10%)	0.0052 (10%)	$\frac{0.2404}{0.0052}$ (10%)	0.2022 (20%)	0.0536
modical	$\frac{0.0002}{0.0136}$ (10%)	$\frac{0.0002}{0.0143}$ (10%)	$\frac{0.0002}{0.0144}$ (10%)	$\frac{0.0002}{0.0143}$ (10%)	0.0000
scono	$\frac{0.0130(1070)}{0.1361(20\%)}$	0.0143 (1070) 0.1324 (40%)	0.0144 (1070) 0.1914 (40%)	0.0140(1070) 0.1999(50%)	0.0278
scene	0.1301 (2070)	0.1324 (4070)	0.1314(4070)	$\frac{0.1222}{0.2417}$ (00%)	0.1309
SUDSET 0/1 T	0.2419 (9070)	0.2424 (90%)	0.2439 (9070)	0.2417(90%)	0.2413
Deta Sat		Conv InfoCoin	ID InfoCain	MIInfoCain	No Sol
Data Set	0.7057 (10%)	$\frac{\text{Copy}+\text{IntoGain}}{0.007}$	$\frac{LP+III0Gain}{0.9124}$	$\frac{\text{MLIMOGAIN}}{0.0014(10\%)}$	1NO Sel.
DIDIEX	$\frac{0.7957(10\%)}{0.0624(000\%)}$	0.8010 (20%)	0.8124 (40%)	0.8014(10%)	0.8519
DIRUS	0.8034(90%)		0.8000 (80%)	0.8021(10%)	$\frac{0.8003}{1.0000}$
CAL500	$\frac{1.0000(10\%)}{0.0700(000\%)}$	$\frac{1.0000(10\%)}{0.0004(70\%)}$	$\frac{1.0000(10\%)}{0.0716(0007)}$	$\frac{1.0000(10\%)}{0.0000(10\%)}$	$\frac{1.0000}{0.0706}$
Corel5k		0.9694 (70%)	0.9716(90%)	$\frac{0.9688}{0.9688}$	0.9706
emotions		0.7372(90%)	$\frac{0.7238}{0.7238}$ (90%)	0.7422(80%)	0.7321
enron	0.8397(20%)	0.8585(90%)	0.8455(50%)	0.8585(90%)	0.8508
flagsml	0.6961 (50%)	$0.8040 \ (90\%)$	0.6703~(10%)	0.6955~(20%)	0.7990
genbase	0.0922~(10%)	0.0922~(10%)	0.0922~(10%)	0.0922~(10%)	0.6102
medical	0.3528~(10%)	0.3773~(10%)	0.3773~(10%)	0.3804~(10%)	0.6534
scene	$\overline{0.4545} \ (20\%)$	0.4537~(40%)	0.4516~(40%)	0.4284~(50%)	0.4629
yeast	0.8167~(70%)	0.8188~(80%)	0.8200~(90%)	0.8155(90%)	<u>0.8134</u>
EXAMPLE BA	SED ACCURAC	CY (inverted)			
Data Set	BR +InfoGain	${f Copy}+{f InfoGain}$	${f LP+InfoGain}$	MLInfoGain	No Sel.
bibtex	0.6860~(10%)	0.6981~(10%)	0.7164~(30%)	0.6981~(10%)	0.7584
birds	0.7893 (90%)	$0.8016\ (80\%)$	0.8021~(80%)	0.7962~(90%)	0.7924
CAL500	$0.7840\ (70\%)$	0.7867~(70%)	0.7897~(40%)	0.7904~(70%)	0.7899
Corel5 k	0.9139 $(90%)$	0.9034~(30%)	0.9101~(90%)	0.9021~(30%)	0.9139
emotions	0.4938~(50%)	0.4892~(90%)	0.4872~(90%)	$\overline{0.4854}$ (70%)	0.4879
enron	0.5922~(10%)	0.6522~(90%)	0.6026~(20%)	0.6481(90%)	0.6379
flagsml	0.3958(30%)	0.4304~(40%)	0.3832(10%)	0.3867(20%)	0.4619
genbase	0.0523(10%)	0.0523 (10%)	0.0523 (10%)	0.0523(10%)	0.6054
medical	$\overline{0.2810(10\%)}$	$\overline{0.2953(10\%)}$	0.2993(10%)	0.2972(10%)	0.5796
scene	$\overline{0.3826}$ (20%)	0.3669(40%)	0.3645(40%)	0.3442(50%)	0.3853
veast	0.5299(90%)	0.5312(90%)	0.5343(90%)	$\frac{0.5271}{0.5271}$ (90%)	0.5293
BANKING LO	SS	0.0012 (0070)	0.0010 (0070)	<u>(00,0)</u>	0.0200
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.3445(30%)	0.3483 (30%)	0.3536(50%)	0.3478(30%)	0.3652
birds	$\frac{0.0110}{0.1963}$ (90%)	0.2018(80%)	0.2036(80%)	0.1980(90%)	0 1990
CAL500	$\frac{0.1000}{0.4676}$ (70%)	0.4684 (80%)	0.2000(00%) 0.4720(90%)	0.4716(70%)	0.1550
Corol5k	$\frac{0.4010}{0.6630}$	0.4004 (0070)	0.5560 (00%)	0.4110(1070)	0.4000
omotions	0.0030(9070)	0.0300 (0070)	$\frac{0.0300(9070)}{0.0775(00\%)}$	0.0374(0070)	0.0019
	0.2803(9070)	0.2112 (0070)	0.2775(9076)	$\frac{0.2704}{0.2400}$	0.2772
	$\frac{0.3223}{0.4500} (40\%)$	0.3340 (90%)	0.3279 (40%)	0.3490(90%)	0.3410
nagsini gophago	0.4389(90%)	0.4007 (90%)	0.4007 (90%)	0.4420 (90%)	$\frac{0.4418}{0.4515}$
genbase	$\frac{0.0257 (10\%)}{0.1007}$	$\frac{0.0237}{0.1202}$ (10%)	$\frac{0.0257}{0.1406}$ (10%)	$\frac{0.0257(10\%)}{0.1410(10\%)}$	0.4313
medical	$\frac{0.1265 (10\%)}{0.1059 (200\%)}$	0.1393 (10%)	0.1406 (10%)	0.1419(10%)	0.1957
scene		0.1747 (40%)	U.1754 (40%)	<u>0.1653 (50%)</u>	0.1913
yeast	0.3339 (90%)	0.3422 (90%)	0.3430(90%)	0.3331(90%)	<u>0.3297</u>
Best values	23	6	10	16	7
(undernned)					
\geq basenine score (bold)	32	27	30	30	
	1				

Table 7.13: Best results achieved with the LP $\,+$ NaiveBayes classifier

HAMMING LOSS						
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.0126 (10%)	0.0129 (20%)	0.0133 $(30%)$	0.0130~(10%)	0.0136	
birds	0.0479 (30%)	0.0477(80%)	0.0472(90%)	0.0463(10%)	0.0473	
CAL500	0.1381 (40%)	0.1380(50%)	0.1381 (20%)	0.1380(70%)	0.1388	
Corel5 k	0.0094 (10%)	0.0094(10%)	0.0094(10%)	$\overline{0.0094(10\%)}$	0.0094	
emotions	$\overline{0.1903(60\%)}$	$\overline{0.1921}$ (80%)	0.1966 (70%)	$\overline{0.1898}$ (90%)	0.1951	
enron	0.0502 (10%)	0.0531(90%)	0.0502(10%)	$\overline{0.0520}$ (70%)	0.0524	
flagsml	$\overline{0.2489(40\%)}$	0.2622(90%)	0.2622(90%)	0.2570(90%)	0.2536	
genbase	$\overline{0.0048(10\%)}$	0.0048(10%)	0.0048(10%)	0.0048(10%)	0.0048	
medical	$\overline{0.0126(10\%)}$	$\overline{0.0149(10\%)}$	$\overline{0.0148(10\%)}$	$\overline{0.0147(10\%)}$	0.0151	
scene	0.0899 (90%)	0.0879(90%)	0.0911(90%)	0.0867(90%)	0.0862	
yeast	0.1915 (90%)	0.1935~(60%)	0.1945 $(90%)$	0.1925(80%)	0.1933	
SUBSET 0/1 L	OSS		<u>. </u>			
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.8614 (10%)	0.8807~(10%)	0.9154~(30%)	$0.8818\ (10\%)$	0.9396	
birds	$\overline{0.5085~(40\%)}$	$0.5240\ (80\%)$	$0.5210\ (90\%)$	$0.5100\ (10\%)$	0.5085	
CAL500	$\overline{1.0000(10\%)}$	1.0000 (10%)	$1.0000 \ (10\%)$	$1.0000 \ (10\%)$	1.0000	
Corel5 k	0.9972 (90%)	0.9980 (90%)	0.9988(90%)	0.9980 (90%)	0.9982	
emotions	$\overline{0.6816}$ (80%)	0.6866 (70%)	0.7019~(60%)	0.6832 $(70%)$	0.7169	
enron	$\overline{0.9013}$ (10%)	0.9424~(90%)	0.9125~(10%)	0.9172~(60%)	0.9260	
flagsml	$\overline{0.8087(10\%)}$	0.8500(40%)	0.8297 (60%)	0.8603(80%)	0.8453	
genbase	$\overline{0.0890}$ (10%)	0.0890 (10%)	0.0890 $(10%)$	0.0890 $(10%)$	<u>0.0890</u>	
medical	$\overline{0.3967(10\%)}$	0.4816(10%)	0.4776(30%)	0.4633(10%)	0.4940	
scene	$\overline{0.3743}$ (90%)	0.3760~(90%)	0.3797(80%)	0.3685(70%)	0.3752	
yeast	0.8097 (90%)	0.8101 (60%)	0.8192(70%)	0.8113(80%)	0.8126	
EXAMPLE BA	SED ACCURAC	CY (inverted)				
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	$0.7457 \ (10\%)$	0.7840~(10%)	0.8307~(30%)	0.7849~(10%)	0.8640	
birds	0.4519 (80%)	0.4622~(80%)	0.4617~(90%)	0.4511~(10%)	0.4515	
CAL500	0.8018~(40%)	0.8033~(50%)	0.8023~(20%)	$\overline{0.8007} (40\%)$	0.8028	
$\operatorname{Corel5k}$	0.9849 (90%)	0.9829~(90%)	0.9897~(90%)	$\overline{0.9834} \ (90\%)$	0.9853	
emotions	0.4427 $(80%)$	$\overline{0.4507}$ (80%)	0.4601~(60%)	0.4423~(70%)	0.4674	
enron	0.6074~(10%)	0.6898~(90%)	0.6254~(10%)	$\overline{0.6586~(60\%)}$	0.6684	
$_{ m flagsml}$	$\overline{0.3679}$ (40%)	0.3982~(90%)	0.3917~(40%)	0.3863~(40%)	0.3896	
genbase	$\overline{0.0584}$ (10%)	0.0584~(10%)	0.0584~(10%)	0.0584~(10%)	0.0584	
medical	$\overline{0.3245} \ (10\%)$	$\overline{0.4100} (10\%)$	$\overline{0.4045} (30\%)$	$\overline{0.3902} (10\%)$	0.4187	
scene	$\overline{0.3280} (90\%)$	0.3322~(90%)	$0.3356\ (80\%)$	$0.3256\ (70\%)$	0.3330	
yeast	0.4804 (90%)	0.4875~(60%)	0.4889~(90%)	0.4848 (80%)	0.4838	
RANKING LO	SS					
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.	
bibtex	0.1249 (10%)	0.1577~(10%)	$0.1845\ (30\%)$	$0.1563\ (10\%)$	0.2083	
birds	$0.0742 \ (10\%)$	0.0759~(90%)	0.0753~(40%)	0.0724~(80%)	0.0746	
CAL500	0.1830(30%)	0.1820~(40%)	0.1823~(40%)	0.1825~(80%)	0.1828	
Corel5 k	0.1325~(80%)	$\overline{0.1340} \ (80\%)$	0.1346~(90%)	0.1338~(60%)	0.1340	
emotions	0.1624~(50%)	0.1591~(80%)	0.1601~(90%)	0.1546~(60%)	0.1633	
enron	0.0883~(10%)	0.0919~(70%)	0.0898~(10%)	0.0924 (90%)	0.0920	
flagsml	0.1844~(40%)	0.1906~(90%)	0.1906~(90%)	0.1952~(30%)	0.2012	
genbase	0.0062 (10%)	0.0062 (10%)	0.0062 (10%)	0.0062 (10%)	<u>0.0062</u>	
medical	0.0329~(10%)	$\overline{0.0384~(30\%)}$	$\overline{0.0389} (30\%)$	$\overline{0.0393} (90\%)$	0.0395	
scene	0.0799 (90%)	0.0791~(90%)	0.0792~(90%)	0.0787~(90%)	<u>0.0774</u>	
yeast	0.1636 (80%)	0.1644 (90%)	0.1660 (90%)	0.1649 (80%)	0.1652	
Best values	30	0	7	16	a	
(underlined)	50	IJ	I	10	3	
\leq baseline score	39	28	27	37		
(bold)			2.			

Table 7.14: Best results achieved with the ML-KNN classifier

HAMMING LO	DSS				
Data Set	BR +InfoGain	Copy+InfoGain	LP+InfoGain	No Sel.	
bibtex	0.0614 (10%)	0.0845 (90%)	0.0845 (90%)	0.0845(90%)	0.0810
birds	$\overline{0.1691}(30\%)$	0.2198(90%)	0.2213(90%)	0.1323(10%)	0.2142
CAL500	0.1690 (10%)	0.1633(10%)	0.1653 (10%)	$\frac{0.1642}{0.1642}$ (10%)	0.2859
Corel5k	0.0097 (10%)	$\frac{0.1000}{0.0112}$ (10%)	0.0103 (10%)	0.0111 (10%)	0.0127
omotions	$\frac{0.0001}{0.2430}$ (20%)	0.0112 (1070) 0.0512 (80%)	0.0100 (1070) 0.2537 (00%)	0.0111 (1070) 0.0481 (70%)	0.0121
ennor	$\frac{0.2459}{0.000}$ (10%)	0.2312 (8070) 0.1646 (1097)	0.2007 (9070)	0.2401(1070) 0.1649(1077)	0.2340
flagaml	$\frac{0.0992}{0.000}$	0.1040 (1070)	0.1270 (1070)	0.1042 (1070)	0.1739
nagsiin	$\frac{0.2290}{0.0040} (20\%)$	0.2598 (40%)	0.2377(3076)	0.2366 (3076)	0.3239
genbase	$\frac{0.0040(10\%)}{0.0105(10\%)}$	$\frac{0.0040(10\%)}{0.0020(00\%)}$	$\frac{0.0040(10\%)}{0.0040(00\%)}$	$\frac{0.0040(10\%)}{0.0040(10\%)}$	0.0339
medical	$\frac{0.0185(10\%)}{0.1686(000\%)}$	0.0220 (80%)	0.0220 (80%)	0.0219(80%)	0.0247
scene		0.1626 (90%)	0.1632(90%)	$\frac{0.1564}{0.1564}$ (70%)	0.1615
yeast	0.2498 (10%)	0.2434(10%)	0.2431(10%)	0.2456(10%)	0.2721
SUBSET 0/1 L	OSS				-
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	0.9229 (10%)	0.9394~(90%)	0.9393~(90%)	0.9394~(90%)	0.9377
birds	0.9705 (90%)	0.9689~(80%)	0.9736~(80%)	0.9209~(10%)	0.9673
CAL500	1.0000 (10%)	1.0000~(10%)	1.0000~(10%)	1.0000~(10%)	1.0000
Corel5 k	0.9946 (70%)	$\overline{0.9940}$ (80%)	0.9960~(70%)	$\overline{0.9944} \ (80\%)$	0.9952
emotions	0.7470 (10%)	$\overline{0.7740\ (10\%)}$	0.7775~(90%)	0.7673~(80%)	0.7859
enron	0.9847 (10%)	0.9982 (10%)	0.9959 $(20%)$	0.9982 (30%)	0.9994
flagsml	$\overline{0.7405(20\%)}$	0.8800(40%)	0.7429(10%)	0.7579(10%)	0.9026
genbase	$\overline{0.0967(10\%)}$	0.0967(10%)	0.0967(10%)	0.0967(10%)	0.7220
medical	$\overline{0.5409(10\%)}$	$\overline{0.6259}$ (20%)	$\overline{0.6381}$ (30%)	0.6208(20%)	0.7352
scene	$\frac{1}{0.6967}$ (10%)	0.7017(90%)	0.7013(90%)	0.6884 (90%)	0.6980
veast	0.8949 (20%)	0.8867 (10%)	0.8763 (10%)	$\frac{0.0001}{0.8904}$ (10%)	0.8966
FYAMDIE BA	SED ACCURAC	$\frac{100001}{2}$	0.0100 (1070)	0.0001 (10/0)	0.0000
Data Set	BB+InfoGain	Conv+InfoGain	LP+InfoGain	MLInfoGain	No Sel
bibtex	0.7797 (10%)	0.8126 (90%)	0.8132(90%)	0.8127(90%)	0.8093
birds	0.00000000000000000000000000000000000	0.8563 (90%)	0.8581(90%)	0.8273(10%)	0.8441
CAL500	0.7628(20%)	0 7766 (50%)	0.7730(40%)	$\frac{0.0210}{0.7676}$ (30%)	0.7840
Corel5k	$\frac{0.11020}{0.8595}$ (90%)	0.8535 (90%)	0.8576 (90%)	0.8536 (90%)	0.8512
emotions	0.8535(5070)	0.8333 (9070) 0.4834 (60%)	0.8370(3070) 0.4873(90%)	0.3330(3070) 0.4750(80%)	0.3312 0.4891
enron	0.4812(0070)	0.7804 (10%)	0.7492 (10%)	$\frac{0.1100}{0.7820}$ (10%)	0.7770
flagsml	$\frac{0.0010(1070)}{0.3628(20\%)}$	0.4023 (40%)	0.1422 (1070) 0.3803 (30%)	0.7020(1070)	0.4880
rophaco	$\frac{0.0020}{0.0510}$ (10%)	0.4020 (40%)	0.0500 (0070)	0.0510 (0070)	0.4000
genbase	$\frac{0.0510(1070)}{0.2606(1077)}$	$\frac{0.0310}{0.4422}$ (2007)	$\frac{0.0310}{0.4408}$ (2007)	$\frac{0.0310(1070)}{0.4308(2007)}$	0.0995
medical	$\frac{0.3090(1070)}{0.4551(0007)}$	0.4422 (2070)	0.4498 (2070)	0.4398 (2076)	0.0314
scene	0.4551(90%)	0.4537 (90%)	0.4543 (90%)	$\frac{0.4387}{0.4387}$ (90%)	0.4511
yeast		0.5200 (10%)	0.5502(70%)	0.5287 (10%)	0.5507
RANKING LO	SS				
Data Set	BR+InfoGain	Copy+InfoGain	LP+InfoGain	MLInfoGain	No Sel.
bibtex	$\frac{0.1627}{0.1627}$ (30%)	0.1641 (40%)	U.1778 (60%)	0.1643 (40%)	0.1899
birds	$\frac{0.1334}{0.1334}$	0.1442 (90%)	0.1453(90%)	0.1346(80%)	0.1342
CAL500	0.3471 (40%)	0.3473~(50%)	$0.3491 \ (90\%)$	$0.3484 \ (90\%)$	0.3503
Corel5k	0.5650(90%)	0.5562 (90%)	0.5591 (90%)	0.5561 (90%)	0.5524
emotions	0.2161 (90%)	0.2137(90%)	0.2240 (90%)	0.2142(80%)	$\frac{0.2135}{0.2133}$
enron	$\frac{0.2083 (40\%)}{0.2083 (40\%)}$	0.2139(90%)	0.2094 (60%)	0.2135(90%)	0.2122
flagsml	0.2531 (40%)	0.2518 (40%)	0.2493 (50%)	0.2443 (30%)	0.3328
genbase	0.0169 (10%)	0.0169~(10%)	0.0169~(10%)	0.0169~(10%)	0.5226
medical	0.0933 (10%)	0.0809~(10%)	0.0832~(10%)	0.0797~(10%)	0.2181
scene	0.1273 (40%)	0.1287~(40%)	0.1277~(50%)	0.1174~(60%)	0.1383
yeast	0.2629 (80%)	0.2672~(90%)	0.2686~(70%)	0.2651(50%)	0.2639
Best values	26	<u></u>	7	15	А
(underlined)	20	0	1	10	*
\leq baseline score	37	28	29	34	
(bold)		20	<u> </u>	го	

Table 7.15: Best results achieved with the RK + NaiveBayes classifier

Appendix C - Application of the MLInfoGain and Lazy MLInfoGain Equations

This Appendix provides an example of multi-label training data set and applies the equations given in Chapters 5 and 6 on this example.

Table 6.2 presents the multi-label example, already provided in Appendix A and in Chapter 6. The data set, composed of two features -X, Y – and their labels, is represented twice. The left occurrence is ordered by the values of X and the right one is ordered by the values of Y.

Dat	a Set Sor	ted by X	Data Set Sorted by Y			
- X -	– Y –	– Labels –	– X –	– Y –	– Labels –	
1	1	A	1	1	A	
1	2	В	2	1	A	
1	3	В	3	1	B,C	
1	4	A,B	4	1	В	
2	1	A	1	2	В	
2	2	A	2	2	A	
2	3	A	3	2	B,C	
2	4	A,B	4	2	A,B	
3	1	B,C	1	3	В	
3	2	B,C	2	3	A	
3	3	B,C	3	3	B,C	
3	4	A,B	4	3	A	
4	1	В	1	4	A,B	
4	2	A,B	2	4	A,B	
4	3	A	3	4	A,B	
4	4	A,B	4	4	A,B	

Table 7.1: Multi-label Data Set Training Example

C.1. Computing the MLInfoGain measure

The entropy of the label set distribution in D, represented by Ent.ML(D), was defined by Equation 5.2.

$$Ent.ML(D) = -\sum_{i=1}^{l} p(\lambda_i) * \log_2 p(\lambda_i) + q(\lambda_i) * \log_2 q(\lambda_i), \qquad (5.2 \text{ revisited})$$
where $p(\lambda_i)$ is the probability that an arbitrary instance in D belongs to class label λ_i , $q(\lambda_i) = 1 - p(\lambda_i)$, and l is the number of labels in the data set.

The data set consists of three labels -A, B and C. Hence, the sum is computed as follows:

$$= p(\lambda_A) * log_2 p(\lambda_A) + q(\lambda_A) * log_2 q(\lambda_A)$$
 (for label A)

$$= \frac{10}{16} * log_2(\frac{10}{16}) + (1 - \frac{10}{16}) * log_2(1 - \frac{10}{16})$$

$$= 0.625 * -0.6780 + 0.375 * -1.415$$

$$= 0.9544$$
 (for label B)

$$= \frac{10}{16} * log_2(\frac{10}{16}) + (1 - \frac{10}{16}) * log_2(1 - \frac{10}{16})$$

$$= 0.625 * -0.6780 + 0.375 * -1.415$$

$$= 0.9544$$
 (for label C)

$$= p(\lambda_C) * log_2 p(\lambda_C) + q(\lambda_C) * log_2 q(\lambda_C)$$
 (for label C)
$$= \frac{4}{16} * log_2 (\frac{4}{16}) + (1 - \frac{4}{16}) * log_2 (1 - \frac{4}{16})$$

$$= 0.25 * -2 + 0.75 * -0.415$$

$$= 0.8112$$

These intermediate results of each label show that label λ_A has the same entropy than label λ_B , and both have more entropy than label λ_C . This can be observed in the data set by noticing that labels λ_A and λ_B are more evenly distributed across the instances, so it is harder to predict if they belong to a given instance; while label λ_C appears in only four instances, so it is easier to predict if it belongs or not to a given instance. The corresponding sum for the data set which gives Ent.ML(D) is:

$$Ent.ML(D) = 0.9544 + 0.9544 + 0.8112$$
$$= 2.7201$$

The next step is to compute the entropy of the label distribution in D, restricted to the values of feature X_j , $1 \le j \le d_j$, represented by $Ent.ML(D, X_j)$ and defined by Equation 5.3.

$$Ent.ML(D, X_j) = \sum_{i=1}^{d_j} [(\frac{|D_{ji}|}{|D|}) * Ent.ML(D_{ji})], \qquad (5.3 \text{ revisited})$$

where D_{ji} , $1 \leq i \leq d_j$, is the partition of D composed of all instances whose value of feature X_j is equal to x_{ji} .

First, the computation of $Ent.ML(D_{ji})$ is required. It is the same procedure showed before for computing Ent.ML(D), but restricted to a partition of the data set for instances where the feature value occurs. This partition is showed in Table 7.2.

- X -	– Y –	– Labels –	
1	1	A	
1	2	В	
1	3	В	
1	4	A,B	

Table 7.2: Data Set Partition where X = 1

The computation of $Ent.ML(D_{X_1})$ (feature X and value 1) is:

$$= p(\lambda_A) * log_2 p(\lambda_A) + q(\lambda_A) * log_2 q(\lambda_A)$$
 (feature X=1 for label A)

$$= \frac{2}{4} * log_2(\frac{2}{4}) + (1 - \frac{2}{4}) * log_2(1 - \frac{2}{4})$$

$$= 0.5 * -1 + 0.5 * -1$$

$$= 1$$

$$= p(\lambda_B) * log_2 p(\lambda_B) + q(\lambda_B) * log_2 q(\lambda_B)$$
 (feature X=1 for label B)

$$= frac 34 * log_2(\frac{3}{4}) + (1 - \frac{3}{4}) * log_2(1 - \frac{3}{4})$$

$$= 0.75 * -0.415 + 0.25 * -2$$

$$= 0.8112$$

$$= p(\lambda_C) * \log_2 p(\lambda_C) + q(\lambda_C) * \log_2 q(\lambda_C)$$
 (feature X=1 for label C)
$$= \frac{0}{4} * \log_2(\frac{0}{4}) + (1 - \frac{0}{4}) * \log_2(1 - \frac{0}{4})$$

$$= 0 + 1 * 0$$

$$= 0$$

$$=> Ent.ML(D_{X_1}) = 1 + 0.8112 + 0$$
$$= 1.8112$$

The value of Ent.ML(D, X), i.e., the multi-label entropy of data set D restricted to the feature X is the sum of $Ent.ML(D_{X_1})$, $Ent.ML(D_{X_2})$, $Ent.ML(D_{X_3})$ and $Ent.ML(D_{X_4})$, multiplied by the frequency $\frac{|D_{ji}|}{|D|}$ on each case. The value $\frac{|D_{ji}|}{|D|}$ corresponds to $\frac{4}{16} = \frac{1}{4} = 0.25$ for all feature values of X, because each value appears 4 times in the data set. The computation is given below:

$$=> Ent.ML(D, X) = \left(\frac{|D_{X_1}|}{|D|} * Ent.ML(D_{X_1})\right) + \left(\frac{|D_{X_2}|}{|D|} * Ent.ML(D_{X_2})\right) + \left(\frac{|D_{X_3}|}{|D|} * Ent.ML(D_{X_3})\right) + \left(\frac{|D_{X_4}|}{|D|} * Ent.ML(D_{X_4})\right)$$
$$= Ent.ML(D, X) = (0.25 * 1.8112) + (0.25 * 0.8112) + (0.25 * 0.8112) + (0.25 * 1.6225) + (0.25 * 2.6225)$$
$$= Ent.ML(D, X) = 1.7169$$

The multi-label entropy of D restricted to feature X is then equal to 1.7169. The same computation for feature Y yields the value of 2.1556.

The last step for computing the MLInfoGain measure is given by Equation 5.4.

$$MLInfoGain(D, X_j) = Ent.ML(D) - Ent.ML(D, X_j)$$
(5.4 revisited)

Here the entropy of the multi-label data set, Ent.ML(D), is used. The multi-label information gain for features X and Y are computed as follows.

$$=> MLInfoGain(D, X) = Ent.ML(D) - Ent.ML(D, X)$$

= 2.7201 - 1.7169
= 1.0032
$$=> MLInfoGain(D, Y) = Ent.ML(D) - Ent.ML(D, Y)$$

= 2.7201 - 2.1556
= 0.5645

For the information gain, the higher, the better. So, feature X would be ranked better than feature Y, because its multi-label information value is higher.

C.2. Computing the Lazy MLInfoGain measure

One aspect of the Lazy MLInfoGain measure is considering each individual feature values separately from the others. This is important because the lazy strategy works at classification time: when an instance is submitted, with specific values for each feature, then the best suited features are ranked and selected by the strategy.

The entropy of the label distribution in D, restricted to a specific value x_{ji} , $1 \le i \le d_j$ and label l_k , $1 \le k \le q$, of feature X_j , $1 \le j \le d$ is given by Equation 6.1.

$$Ent.ML(D, X_j, x_{ji}, l_k) = Ent.ML(D_{jik}).$$
(6.1 revisited)

This is the same intermediate computation done in section C.1 for $Ent.ML(D_{X_j})$. For instance, the computation of feature X and value 1 is:

$$= p(\lambda_A) * log_2 p(\lambda_A) + q(\lambda_A) * log_2 q(\lambda_A)$$
 (feature X=1 for label A)
$$= \frac{2}{4} * log_2(\frac{2}{4}) + (1 - \frac{2}{4}) * log_2(1 - \frac{2}{4})$$

$$= 0.5 * -1 + 0.5 * -1$$

$$= 1$$

Table 7.3 shows the lazy entropy scores for all feature values and labels in the multi-

label data set. An entropy of 1 indicates that the value is evenly distributed for a specific label. For instance, when Y = 1, half of the instances have the label A, and so the entropy is equal to 1 (harder to predict). On the other hand, when Y = 4, the entropy is 0 for all labels, because they are always the same: A, B and $\neg C$ (not C).

– Value –	– Label A –	– Label B–	– Label C –
X = 1	1	0.8112	0
X = 2	0	0.8112	0
X = 3	0.8112	0	0.8112
X = 4	0.8112	1	0.8112
Y = 1	1	1	0.8112
Y = 2	1	1	1
Y = 3	1	1	0.8112
Y = 4	0	0	0

Table 7.3: Lazy Entropy Scores for each Value and Label in the Example Data Set

After computing $Ent.ML(D, X_j, x_{ji}, l_k)$ for all feature values and labels, the next step is given by Equation 6.2, which aggregates the result for all q labels in D using the min function, in order to identify feature values which best discriminate at least one label.

$$LazyEnt.ML(D, X_j, x_{ji}) = min_{k=1}^{k=q}Ent(D_{jik}).$$
(6.2 revisited)

For instance, when X = 4 the equation LazyEnt.ML(D, X, 4) results in min (0.8112, 1, 0.8112), which is equal to 0.8112. A higher entropy indicates that this feature value is harder to predict, according to the information gathered from the training data set. On the other hand, when Y = 4, the equation results in min (0, 0, 0), which is 0. This indicates that this feature value is strongly correlated with at least one label, so it can aid the classification task.

The last step of the Lazy MLInfoGain technique is computing the LazyML.IG score, which is used in the ranking and is given by Equation 6.3.

$$LazyML.IG(D, X_j, x_{ji}) = Ent.ML(D) - min[Ent.ML(D, X_j), LazyEnt.ML(D, X_j, x_{ji})]$$
(6.3 revisited)

It computes the lazy multi-label information gain of a specific value for the ranking lazy strategy. As an example, the values for an instance with X = 4 and Y = 4 are computed as follows.

$$=> LazyML.IG(D, X, 4) = Ent.ML(D) - min[Ent.ML(D, X), LazyEnt.ML(D, X, 4)]$$
$$= 2.7201 - min(1.0032, 0.8112)$$
$$= 2.7201 - 0.8112$$
$$= 1.0989$$

$$=> LazyML.IG(D, Y, 4) = Ent.ML(D) - min[Ent.ML(D, Y), LazyEnt.ML(D, Y, 4)]$$
$$= 2.7201 - min(0.5645, 0)$$
$$= 2.7201 - 0$$
$$= 2.7201$$

For the information gain, the higher, the better. So, for the Lazy MLInfoGain strategy, the Y feature would be ranked higher than the X feature, for an instance containing X = 4and Y = 4 values. Despite the fact that X has a higher multi-label information gain overall (used in the MLInfoGain "eager" strategy), by postponing the feature selection to the moment of classification, when the values of the instance are known, the lazy strategy can take a more informed decision and keep the features with higher lazy multi-label information gain scores.