Efficient Pairwise Multilabel Classification

Eneldo Loza Mencía
Knowledge Engineering Group, TU Darmstadt
eneldo@ke.tu-darmstadt.de

Outline

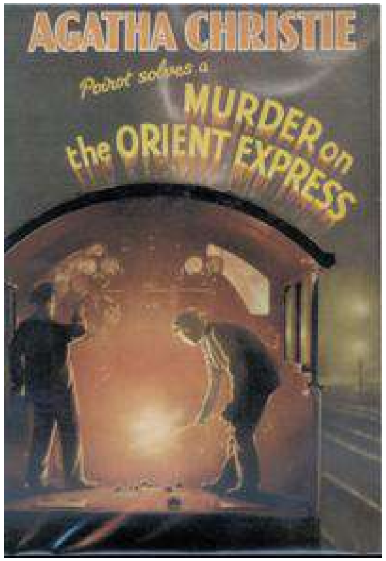
Introduction

Foundations: QVoting and perceptrons

Large number of labels

Summary

Multilabel setting



Genres:

Crime, Mystery, Thriller

Subjects (LOC):

Private Investigators, Orient Express, ...

Keywords:

mystery, fiction, crime, murder, british, poirot, ...

…

- assignment of an object x to a **subset** of a set of label Y
- in contrast to
  - multiclass classification: mapping to exactly one class
  - binary classification: mapping to one of only two classes

Typical application areas

- text: tagging/indexing of news, web pages, blogs, … with keywords, topics, genres, authors, languages, writing styles, …
- multimedia: detection of scenes/object (images), instruments, emotions, music styles (audio)
- biology: classification of functions of genomes and protein

Classification learning: formal definition

- the process of learning of assignments between objects and classes in order to automatically predict these mapping

  Given input:
  - a set of training objects $x_1, \ldots, x_m$, x_i vectors in $\mathbb{R}^a$
  - a set of label mappings $y_1, \ldots, y_m$
    - binary: y_i element of $Y=\{0,1\}$
    - multiclass: y_i element of $Y=\{\lambda_1, \ldots, \lambda_n\}$
    - multilabel: y_i subset of $Y=\{\lambda_1, \ldots, \lambda_n\}$

  Objective:
  - find a function h: $\mathbb{R}^a \rightarrow Y$ which maps x_i to y_i
    - as accurate as possible
    - as efficient as possible

Decompositive approaches

Main solutions in order to solve multilabel problems:

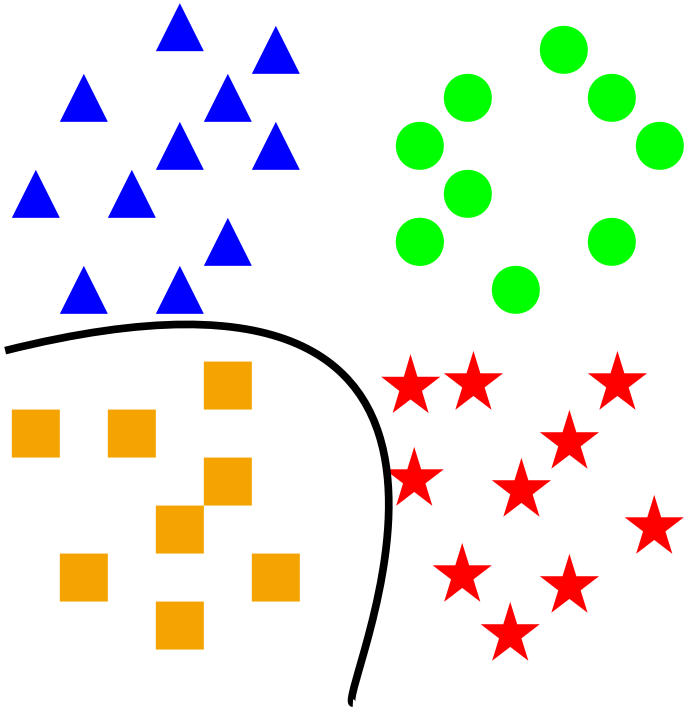
- adaptation of algorithms to learn multilabel data
  - not trivial and often not possible
- decomposition of multilabel problems into binary problems
  - well known problem setting, clear semantics
  - many state-of-the-art binary learners: SVMs, rule learners

Two competing decompositive approaches:

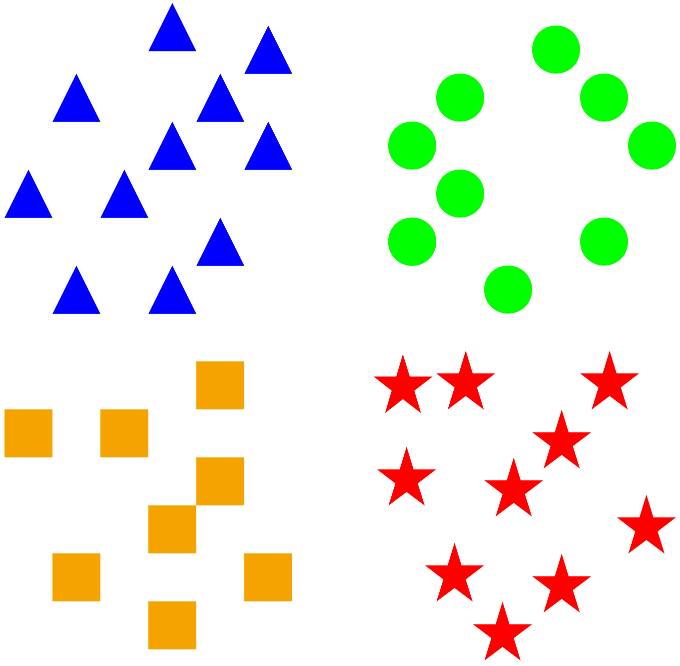
- binary relevance decomposition: learn one classifier for *each class*
  - aka one-against-all
- pairwise decomposition: learn one classifier for *each pair of classes*
  - aka one-against-one, round robin

Decompositive Approaches

TECHNISCHE UNIVERSITÄT DARMSTADT

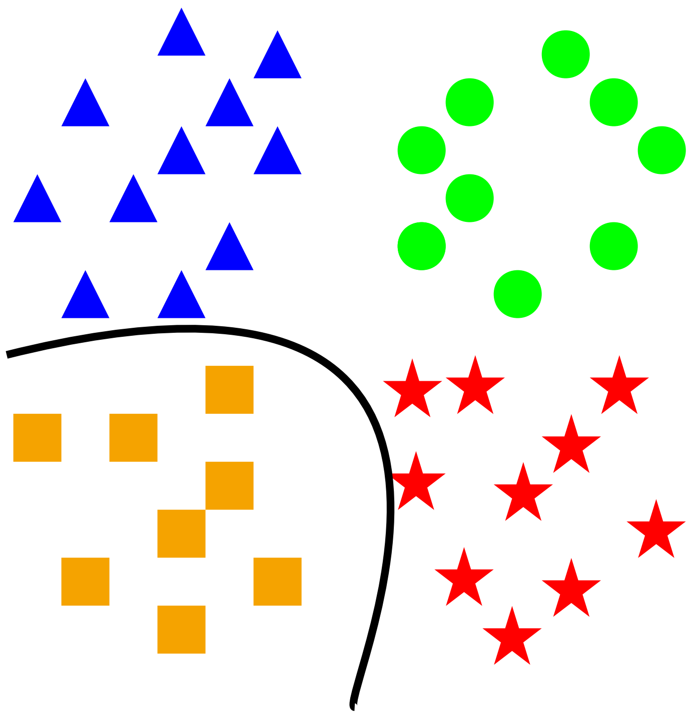


binary relevance decomposition

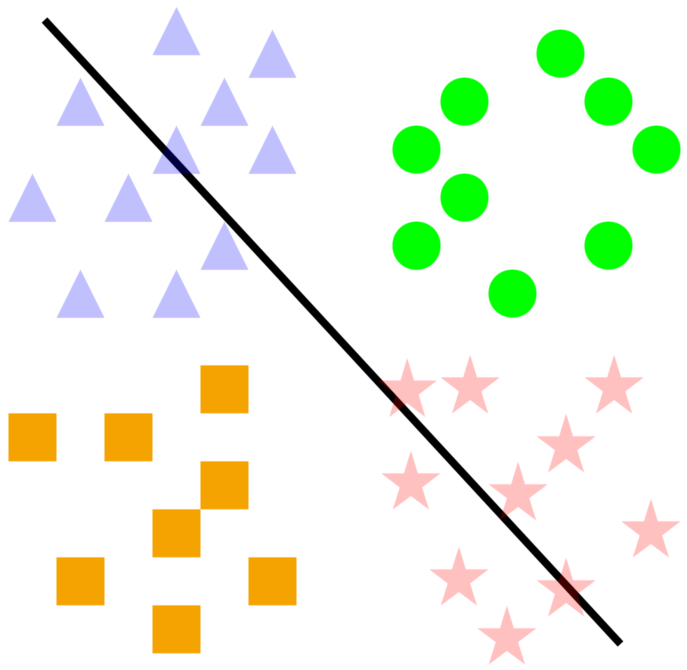


pairwise decomposition

Decompositive Approaches

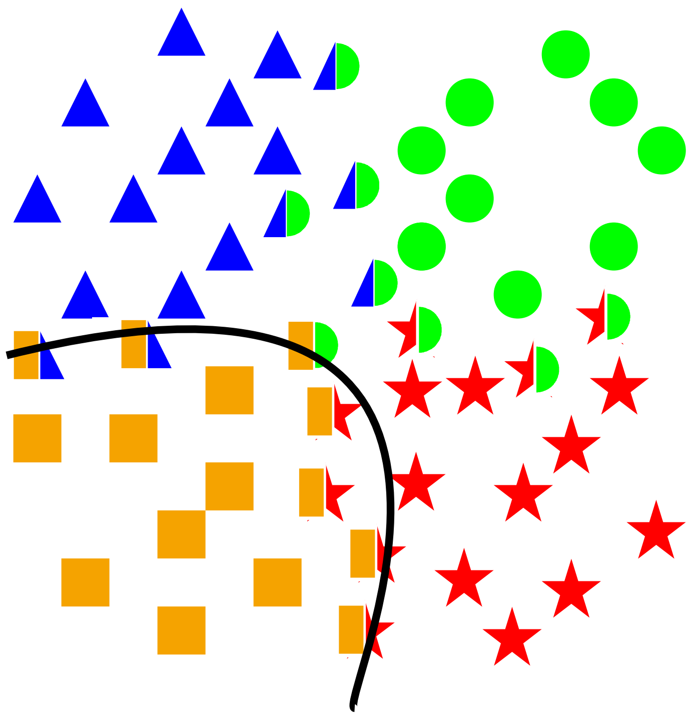


binary relevance decomposition

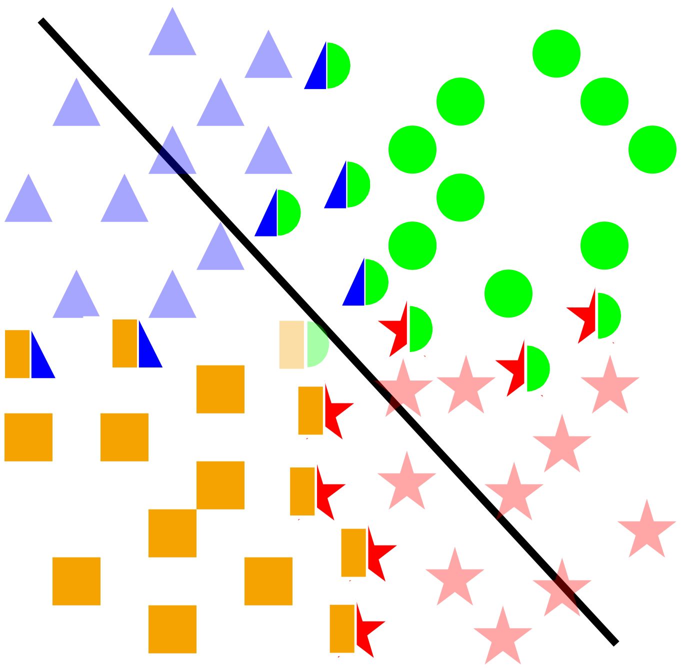


pairwise decomposition

Decompositive Approaches Multilabel



binary relevance decomposition

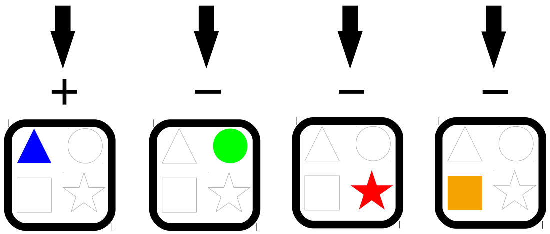


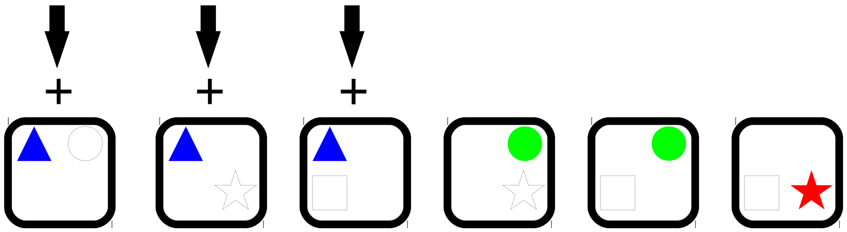
pairwise decomposition

Decompositive Approaches
Training

in: x_1: x_1:

h:

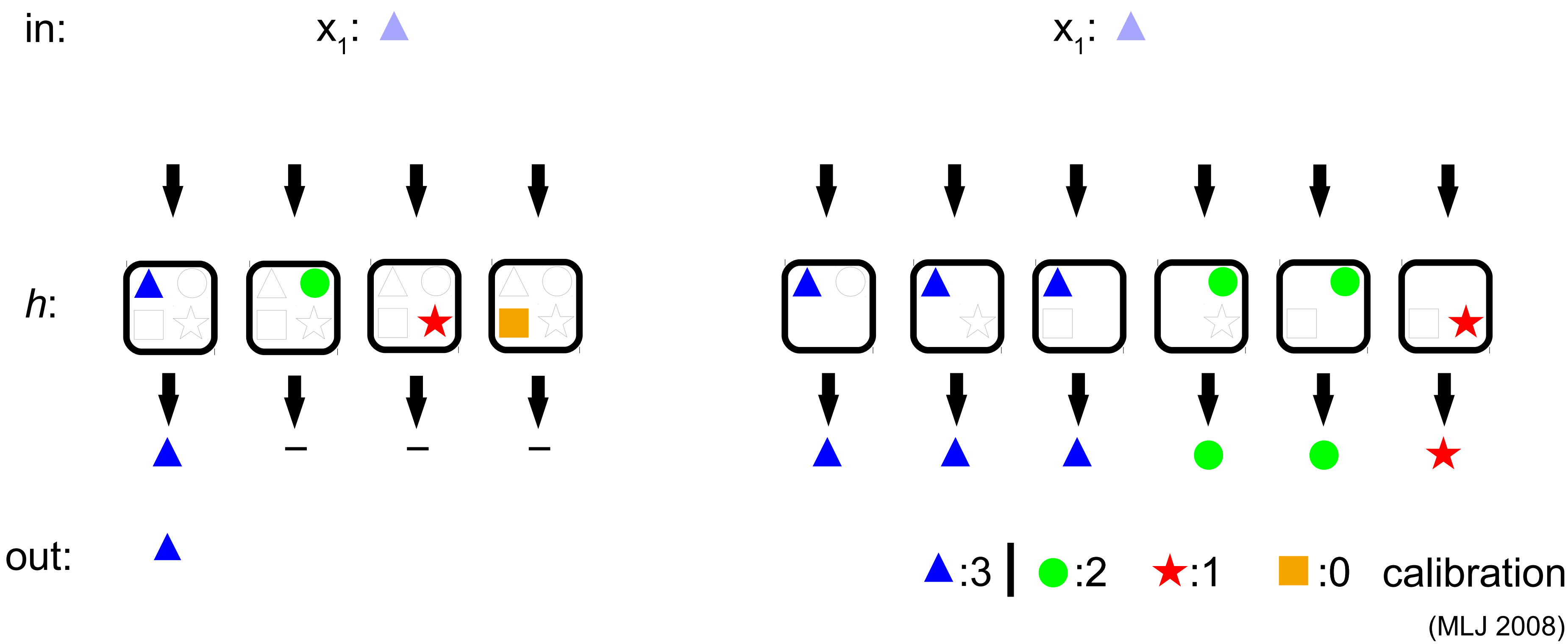




binary relevance decomposition

pairwise decomposition

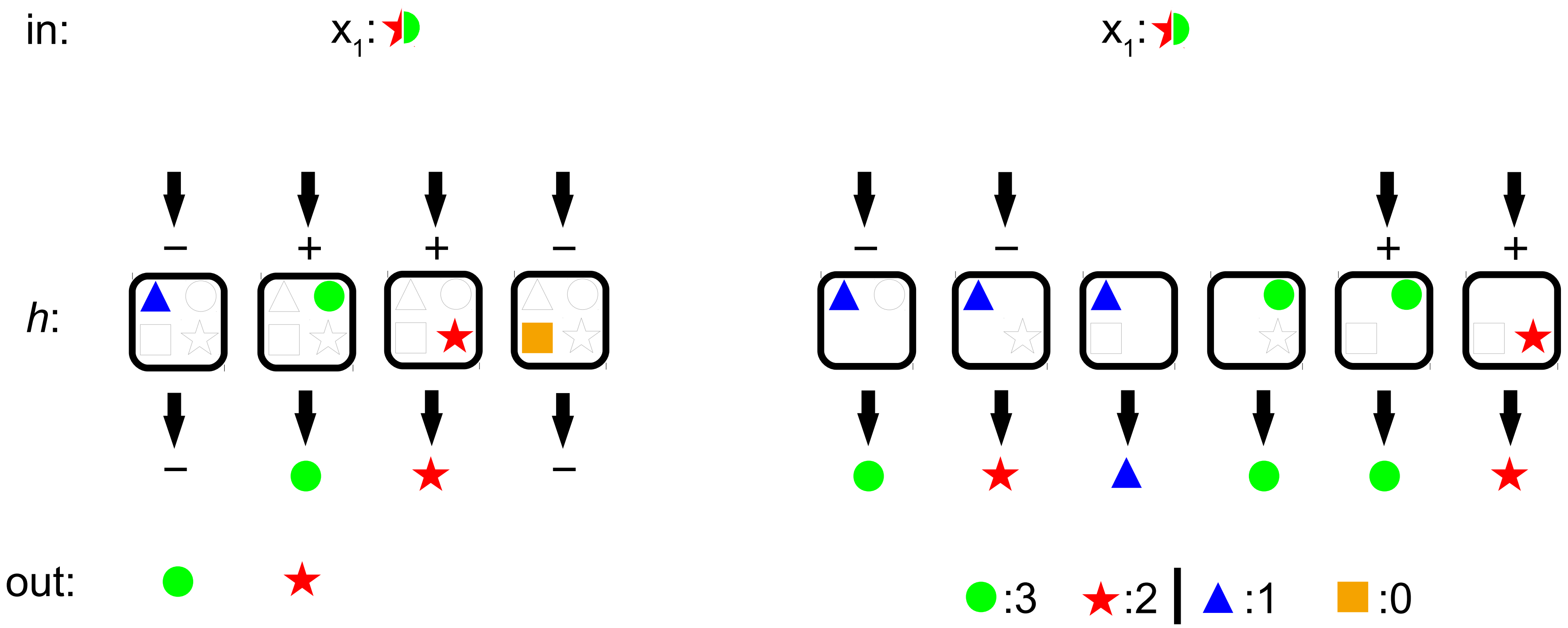
Decompositive Approaches
Predicting



binary relevance decomposition

pairwise decomposition

Decompositive Approaches
Training and predicting 2



binary relevance decomposition

pairwise decomposition

Advantages vs. Disadvantages

TECHNISCHE UNIVERSITÄT DARMSTADT

Binary relevance:

– sub-problems of same size
  - hard to learn

+ linear number of sub-problems
  - but comparable or even more training costs

– learn each problem separately and independently
  - loss of label interdependencies

+ parameter free, works out of the box

Pairwise decomposition:

+ small sub-problems
  - easier to learn, faster to train

– quadratic number of sub-problems
  - high memory costs
  - high testing costs

+ consideration of pairwise relations
  - but loss of information in the label intersections

+ high degree of parallelization

+ class incremental

Challenges in efficient pairwise multilabel classification

- **Dimensionality of label space:**
  - the number of labels

General challenges in multilabel classification:

- Quantity of data:
  - the number of training and testing examples
- Availability of data:
  - real-time processing
- Dependencies between the labels:
  - exploit label correlations

Starting point

memory: $O(n^2)$

- P1 and P2 on y-axis

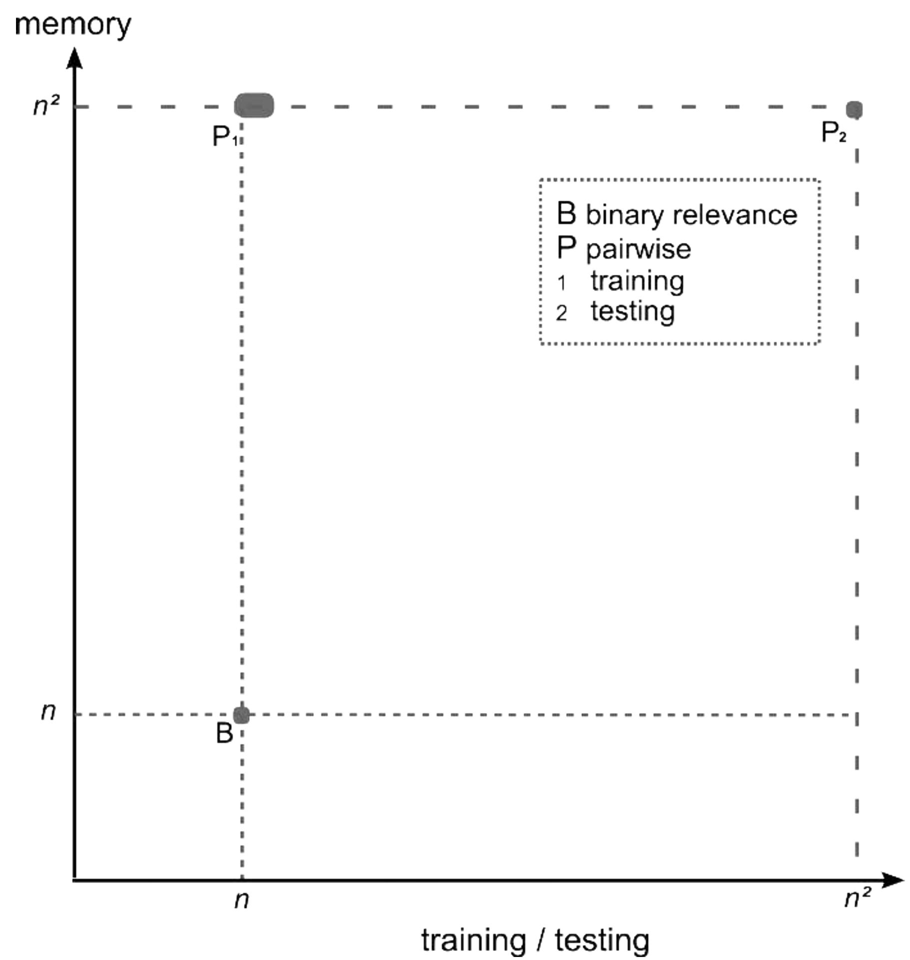
training: $O(d\ n)$

- P_1 on x-axis

testing: $O(n^2)$

- P_2 on y-axis

objective: come as close as possible to binary relevance (B)



n: labels
d: avg. labelset size
m: training instances

Quick Voting

(NC 2010)

Quick Voting

basic idea: many situations where a particular label cannot win anymore

- number of lost comparisons is greater than the amount of losses that the top labels can have at the end
- similar to the situation in sports when the winner of a league is known before the last day of play

  → further matches can be safely omitted

example from FIFA World Cup 2006:

12 Jun	AUS : JPN	3:1
13 Jun	BRA : CRO	1:0
18 Jun	BRA : AUS	2:0
18 Jun	JPN : CRO	0:0
22 Jun	JPN : BRA	1:4
22 Jun	CRO : AUS	2:2

pos.	country	pts.
1.	Brazil	9
2.	Australia	4
3.	Croatia	2
4.	Japan	1

- strategy: let the best team (Brazil) play first
- best case: only three matches
- worst case: max. 6

Quick Voting

basic idea: many situations where a particular label cannot win anymore

- number of lost comparisons is greater than the amount of losses that the top labels can have at the end
- similar to the situation in sports when the winner of a league is known before the last day of play
  - → further matches can be safely omitted

example from FIFA World Cup 2006:

12 Jun	AUS : JPN	3:1
13 Jun	BRA : CRO	1:0
18 Jun	BRA : AUS	2:0
18 Jun	JPN : CRO	0:0
22 Jun	JPN : BRA	1:4
22 Jun	CRO : AUS	2:2

pos.	country	pts.
1.	Brazil	9
2.	Australia	4
3.	Croatia	2
4.	Japan	1

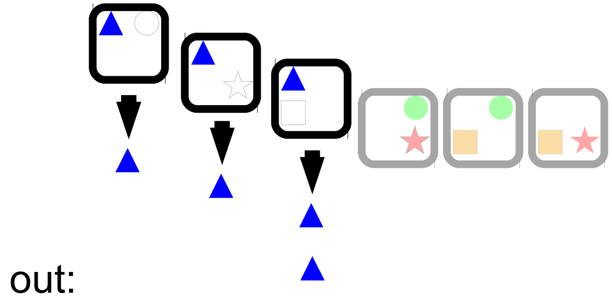
- strategy: let the best team (Brazil) play first
- best case: only three matches
- worst case: max. 6

Quick Voting

algorithm sketch:

- select best label with lowest number of losses
- select second best label and compare to best, if not yet done
- repeat until all relevant labels found

- algorithm is guaranteed to return the same winners as full voting
  - it just induces an order on the evaluations



Results: efficiency

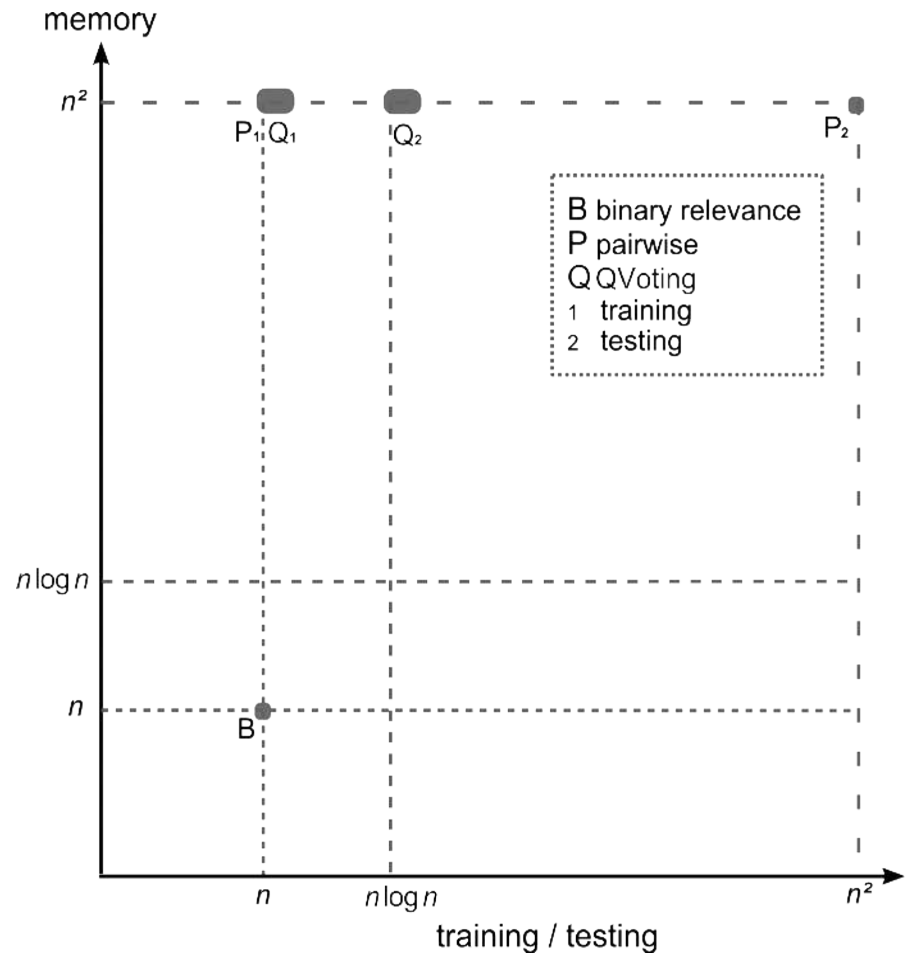
testing:

from O(n) to ~O(d n log(n)) in practice

- without any loss in quality

memory:

still O(n²)



n: labels
d: avg. labelset size
m: training instances

Results: predictive quality

- usage of perceptron algorithm as base learner
- pairwise approach consistently better than BR
- competitive to SVMs
  - but: SVMs often not applicable

		HamLoss		Prec		Rec		F1	
dataset	n	BR	CMLPP	BR	CMLPP	BR	CMLPP	BR	CMLPP
scene	6	10.42	**10.00**	71.80	**71.83**	71.21	**74.20**	71.19	**72.76**
emotions	6	35.64	**34.08**	46.78	**48.62**	60.15	**61.90**	52.63	**54.47**
yeast	14	24.09	**22.67**	60.47	**62.37**	59.07	**63.31**	59.76	**62.83**
tmc2007	22	7.37	**6.78**	62.57	**64.16**	66.47	**73.61**	64.46	**68.56**
genbase	27	**0.26**	0.48	99.22	**99.59**	**95.49**	90.60	**97.32**	94.88
medical	45	**1.51**	1.51	71.72	**76.02**	**75.84**	66.75	**73.72**	71.08
enron	53	7.56	**6.01**	41.56	**52.82**	47.05	**49.51**	44.13	**51.11**
mediamill	101	4.52	**4.16**	42.28	**56.66**	10.05	**19.70**	16.24	**29.23**
rcv1	103	1.26	**1.03**	80.15	**84.89**	79.70	**81.61**	79.93	**83.22**
r21578	120	0.78	**0.55**	59.98	**72.89**	**78.36**	76.68	67.92	**74.63**
bibtex	159	1.57	**1.35**	46.53	**57.97**	**36.30**	34.84	40.78	**43.53**
eurlex_sm	201	0.76	**0.54**	63.39	**77.88**	**74.11**	71.57	68.32	**74.59**
eurlex_dc	410	0.26	**0.17**	56.26	**79.21**	**70.54**	61.98	62.58	**69.54**
delicious	983	5.58	**3.48**	11.88	**19.77**	**29.59**	26.51	16.95	**22.65**

Foundations: QVoting and perceptrons

Multilabel Pairwise Perceptrons

(IJCNN 2008, MLJ 2008)

Multilabel Pairwise Perceptrons (MLPP)

perceptron algorithm: learns a separating hyperplane between positive and negative examples

- simple and fast:

> *classifying:* compute scalar product of instance vector $\bar{x}$ and hyperplane normal vector $\bar{w}$ and predict class $sgn(\bar{x} \cdot \bar{w}) \in \{-1, 1\}$
> *training:* add either $\bar{x}$ or $-\bar{x}$ to normal vector only if training instance is misclassified

- good performance in text-classification (large and sparse feature space)
- on-line learning algorithm
  - efficient alternative to Support Vector Machines

Results: efficiency of perceptrons

TECHNISCHE UNIVERSITÄT DARMSTADT

Reuters Corpus Volume 1 (rcv1)

- 535,987 training news articles
- 268,427 for testing
- 25,000 features
- 103 distinct labels
- ~3.24 labels per example

		HamLoss		Prec		Rec		F1	
dataset	n	BR	CMLPP	BR	CMLPP	BR	CMLPP	BR	CMLPP
scene	6	10.42	10.00	71.80	71.83	71.21	74.20	71.19	72.76
emotions	6	35.64	34.08	46.78	48.62	60.15	61.90	52.63	54.47
yeast	14	24.09	22.67	60.47	62.37	59.07	63.31	59.76	62.83
tmc2007	22	7.37	6.78	62.57	64.16	66.47	73.61	64.46	68.56
genbase	27	0.26	0.48	99.22	99.59	95.49	90.60	97.32	94.88
medical	45	1.51	1.51	71.72	76.02	75.84	66.75	73.72	71.08
enron	53	7.56	6.01	41.56	52.82	47.05	49.51	44.13	51.11
mediamill	101	4.52	4.16	42.28	56.66	10.05	19.70	16.24	29.23
rcv1	103	1.26	1.03	80.15	84.89	79.70	81.61	79.93	83.22
r21578	120	0.78	0.55	59.98	72.89	78.36	76.68	67.92	74.63
bibtex	159	1.57	1.35	46.53	57.97	36.30	34.84	40.78	43.53
eurlex_sm	201	0.76	0.54	63.39	77.88	74.11	71.57	68.32	74.59
eurlex_dc	410	0.26	0.17	56.26	79.21	70.54	61.98	62.58	69.54
delicious	983	5.58	3.48	11.88	19.77	29.59	26.51	16.95	22.65

BR throughput:

- 0.9 ms / training doc
- 1.3 ms / test doc

MLPP throughput:

- 1.3 ms / training doc
- 4.2 ms / test doc
  - 13.5 ms without QVoting

- an efficient SVM only terminated on 8 of 14 datasets

Large number of labels

EUR-Lex and Dual MLPP

(ECML 2008)

EUR-Lex repository

- 19328 (freely accessible) documents of the *Directory of Community legislation in force* of the European Union
  - documents available in several European languages
- multiple classifications of the same documents

EUR-Lex repository

Title and reference

Council Directive 91/250/EEC of 14 May 1991 on the legal protection of computer programs

Classications

EUROVOC descriptor

- **data-processing law**
- **computer piracy**
- **copyright**
- **software**
- **approximation of laws**

Directory Code:

- **Law relating to undertakings/IPR Law**

Subject matter:

- **Internal market**
- **Industrial and commercial property**

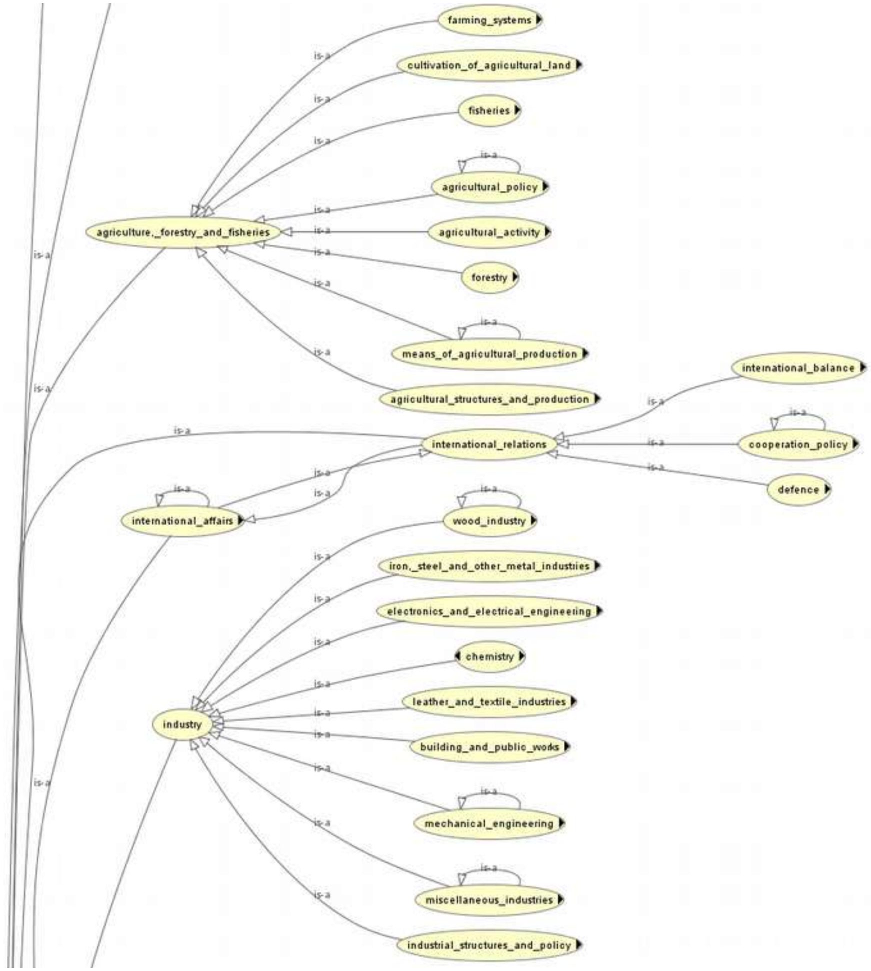
Text

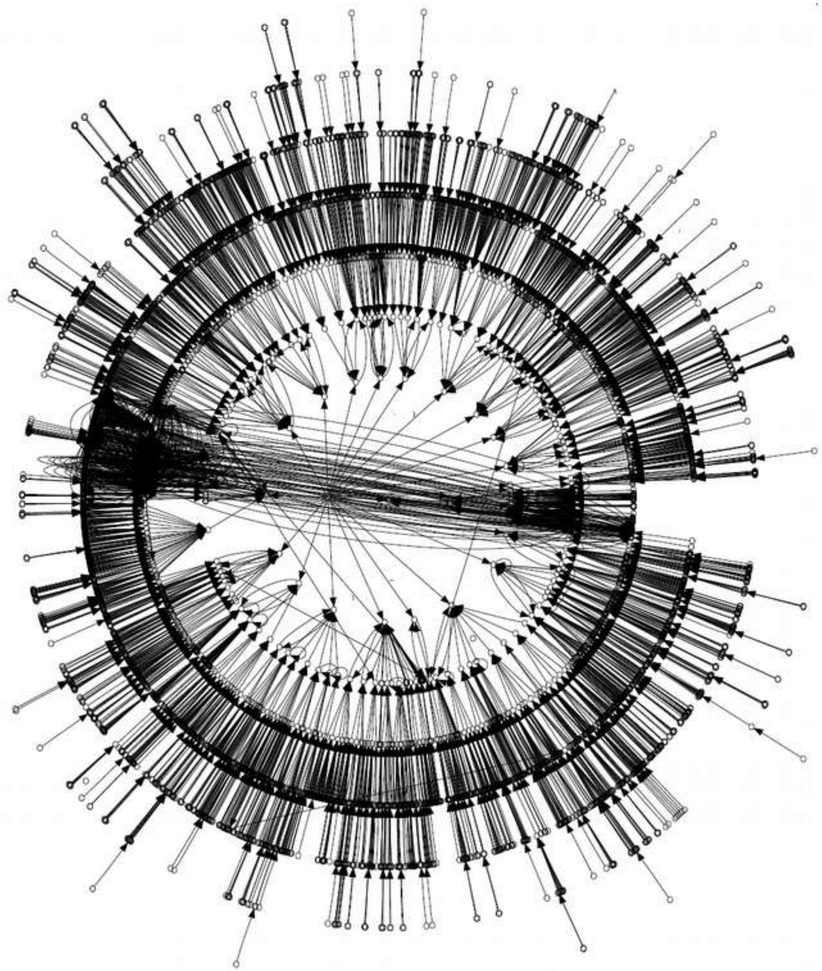
COUNCIL DIRECTIVE of 14 May 1991 on the legal protection of computer programs (91/250/EEC)
THE COUNCIL OF THE EU,
Having Regard to the Treaty establishing the European Economic Community and in particular Article 100a thereof,
Having regard to the proposal of the Commission (1), ...

EUR-Lex repository

- 19328 (freely accessible) documents of the *Directory of Community legislation in force* of the European Union
  - documents available in several European languages
- multiple classifications of the same documents
- most challenging one: **EUROVOC** descriptors associated to each document
  - **3965** descriptors, on average 5.37 labels per document
  - descriptors are organized in a hierarchy with up to 7 levels

EUR-Lex repository





EUR-Lex repository

TECHNISCHE UNIVERSITÄT DARMSTADT

- 19328 (freely accessible) documents of the *Directory of Community legislation in force* of the European Union
  - documents available in several European languages
- multiple classifications of the same documents
- most challenging one: **EUROVOC** descriptors associated to each document
  - **3965** descriptors, on average 5.37 labels per document
  - descriptors are organized in a hierarchy with up to 7 levels
  - but here: EUROVOC problem is considered flat, the hierarchy is ignored!
    - as in *folksonomies* and keyword indexing/tagging

EUROVOC

objective: make the EUROVOC problem amenable to pairwise decomposition

- this means maintaining 8,000,000 base learners in memory!
  - 152 GB in our setting
- training and testing times would not be that problematic
  - training only ~5x, testing ~20x slower

→ solution: use dual form of the perceptron

- perceptron can be reformulated as linear combination of the (misclassified) training instances

Dual MLPP

- perceptron can be reformulated as linear combination of the (misclassified) training instances

  classifying with w: $h'(\mathbf{x}) := \mathbf{x} \cdot \mathbf{w}$

  training rule for w: $\alpha_i = (\mathbf{y}_i - h_i(\mathbf{x}_i)) \quad \mathbf{w}_{i+1} = \mathbf{w}_i + \alpha_i \mathbf{x}_i$

  dual form of w: $\mathbf{w} = \sum_{i=1}^{m} \alpha_i \mathbf{x}_i$

  dual classifying: $h'(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i \cdot \mathbf{x}_i \mathbf{x}$

- we maintain factors α and training instances in memory instead of the w's

Dual MLPP

- additional loop over training examples is necessary, but
  - $x_i x$ can be computed for all perceptrons simultaneously
  - α's are very sparse (only ~1.78 per column are not zero)

n²= 8 mio.
dot products

$$\begin{aligned}
\mathbf{w}_{1,2}\mathbf{x} &= \alpha^1_{1,2}\mathbf{x}_1\mathbf{x} + \alpha^2_{1,2}\mathbf{x}_2\mathbf{x} + \ldots + \alpha^m_{1,2}\mathbf{x}_m\mathbf{x} \\
\mathbf{w}_{1,3}\mathbf{x} &= \alpha^1_{1,3}\mathbf{x}_1\mathbf{x} + \alpha^2_{1,3}\mathbf{x}_2\mathbf{x} + \ldots + \alpha^m_{1,3}\mathbf{x}_m\mathbf{x} \\
&\vdots \\
\mathbf{w}_{1,n}\mathbf{x} &= \alpha^1_{1,n}\mathbf{x}_1\mathbf{x} + \alpha^2_{1,n}\mathbf{x}_2\mathbf{x} + \ldots + \alpha^m_{1,n}\mathbf{x}_m\mathbf{x} \\
\mathbf{w}_{2,3}\mathbf{x} &= \alpha^1_{2,3}\mathbf{x}_1\mathbf{x} + \alpha^2_{2,3}\mathbf{x}_2\mathbf{x} + \ldots + \alpha^m_{2,3}\mathbf{x}_m\mathbf{x} \\
&\vdots
\end{aligned}$$

m= 19328 dot products

n: labels
d: avg. labelset size
m: training instances

Experiments on EUROVOC

TECHNISCHE UNIVERSITÄT DARMSTADT

memory consumption now comparable (~1,3 GB)

- that was our main obstacle

substantial improvement over BR (improved version) in predictive quality

- Average Precision of 53% compared to 38%
- Break-even point of 48% compared to 35%
- but also depends on epochs

comparison for training/testing time less clear

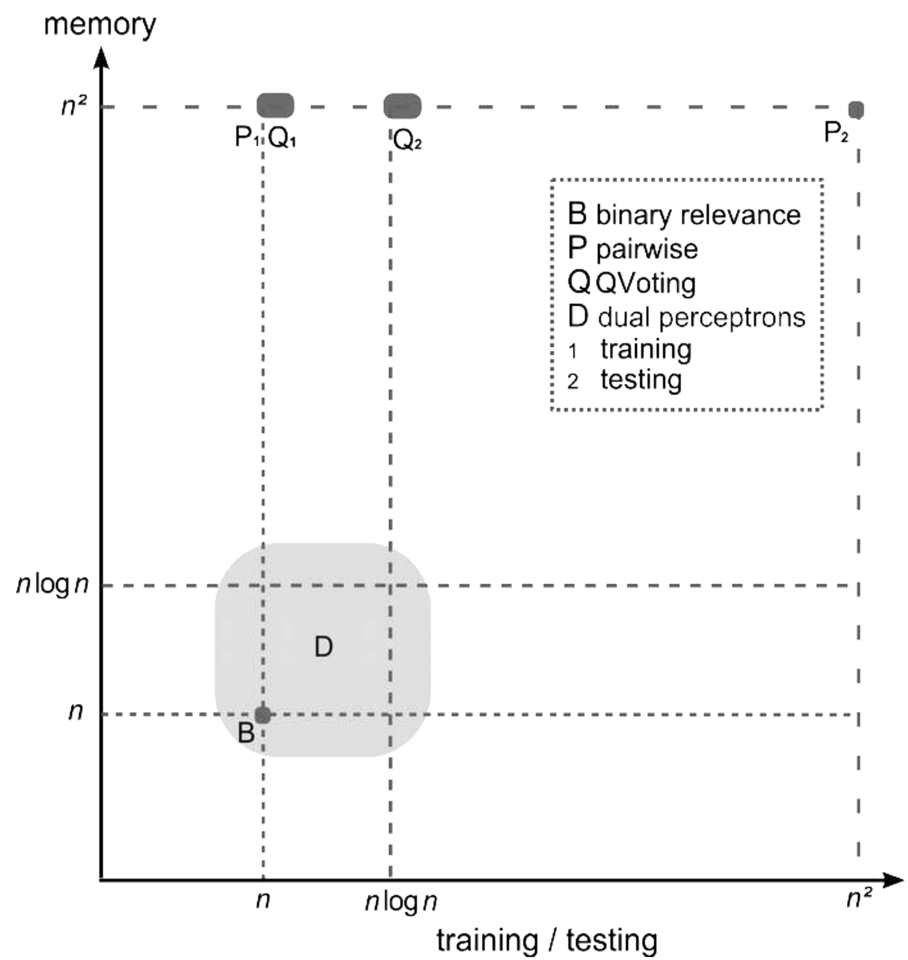
- Dual MLPP even needs less arithmetic operations for training
  - but still much more CPU-time
- BR is clearly faster in testing

Summary

dual form allows to process highly dimensional data

- bottleneck is now mainly the number of training examples employed
- complexity now is roughly O(m d n)

EUROVOC still very challenging for pairwise decomposition



n: labels
d: avg. labelset size
m: training instances

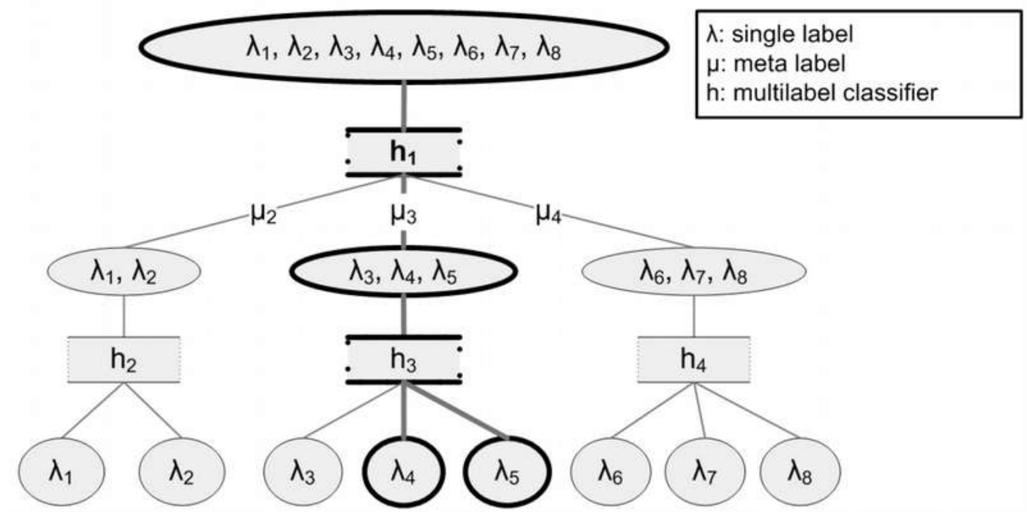
Large number of labels

HOMER:
Hierarchy of Multilabel Classifiers

(PL 2009)

HOMER: Hierarchy of Multilabel Classifiers

- breaks up the problem into subproblems organized in a hierarchy
- k labels are joined to one multilabel, which in turn is one possible label in the parent multilabel problem
- labels are joined by balanced k-means
- idea: use pairwise decomposition at the inner nodes
  - further reduce memory consumption
  - also reduce training/testing costs
  - but hopefully maintain predictive quality



Computational Costs

reduction depends on user-defined subproblem size k

- usually from 5-10

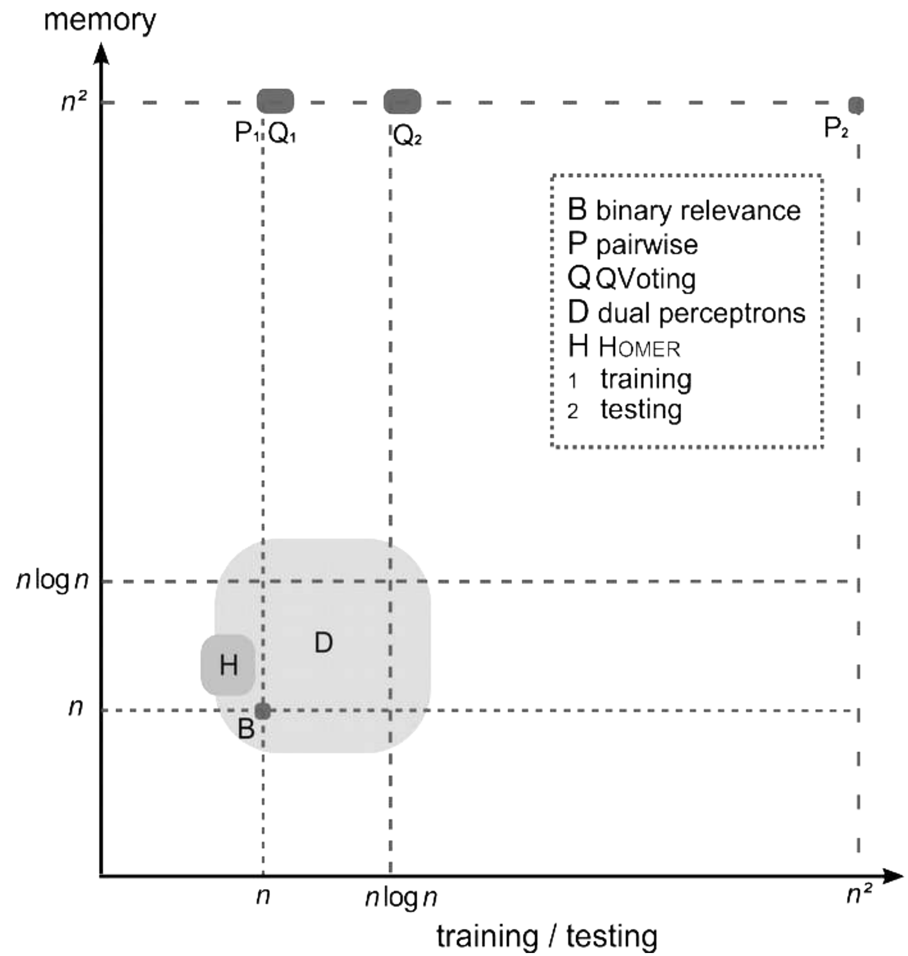
memory: from O(n²) to O(kn) base learners

training: O(log n/n) reduction in training instances used

- supra-linear
  - even less than BR

testing: O(k/n) reduction in base learner evaluations

- less than BR for large n



n: labels
d: avg. labelset size
m: training instances

Experiments

tested on four datasets with 101 to 632 labels and 16000 to 49000 instances

- formal analysis for computational costs were confirmed
- HOMER faster than BR in training and almost for testing

HOMER outbalances recall and precision

- calibration underestimates of the sizes of the labelsets for large n
  - very high precision values, but obviously low recall

HOMER with pairwise decomposition mostly outperformed all other combinations in terms of F1

- BR approaches had generally higher recall, but much lower precision

Summary

HOMER and pairwise decomposition harmonize very well

- substantially reduces computational costs
- maintains advantage of pairwise decomposition over BR
  - though number of instances in metalabels increase

    → harder sub-problems at inner nodes
  - though many pairwise relations between labels are not considered anymore

HOMER enables to apply pairwise decomposition to potentially arbitrarily large datasets

- margin to BR reduced to a user-defined constant factor
- though, problem transformation is not equivalent anymore

Summary

Conclusions

starting point:

- pairwise classification is considered more accurate than BR
- but low efficiency and scalability

presented approaches overcome the main obstacle: the quadratic dependency on number of labels for

- predicting (QVoting)
- memory (Dual MLPP)
- training, predicting, memory (HOMER)

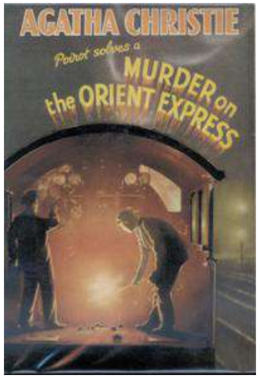
general (slightly biased) recommendation: try to use pairwise classification instead of binary relevance whenever possible

- gain predictive quality, gain even efficiency (HOMER)
- choose from approaches depending on desired needs and trade-off

Further Works

pairwise successfully applied to multitask learning (MLD 2010)

- exploit label dependencies between several domains of labels
- consider all tasks as one big joint task



Genres:

Crime, Mystery, Thriller

Subjects (LOC):

Private Investigators, Orient Express, ...

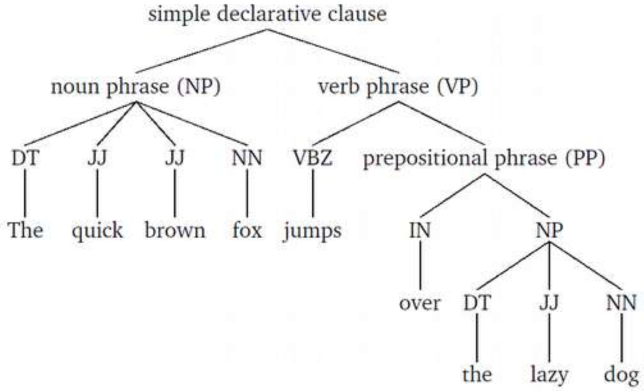
Keywords:

mystery, fiction, crime, murder, british, poirot, ...

...

syntactic parsing (LWA 2010)

- similar to multitask learning
- consider all annotation at once instead of separately



exploit exceptional label dependencies in subgroups of the data (in progress)

- further enhance capabilities of exploitation of dependencies

Perspectives

investigation of semantic hashing techniques

- mapping to reduced label output space
- promising alternative to HOMER
- participation in ECML 2012 Discovery Challenge
- 2.4 million Wikipedia documents, 325,000 categories

extension of the pairwise framework

- consideration of instances in label intersections
  → consideration of hierarchical label structures
- adapt voting to different needs and objective losses

References

(MLJ 2008) J. Fürnkranz, E. Hüllermeier, E. Loza, K. Brinker: *Multilabel Classification via Calibrated Label Ranking*. Machine Learning, vol. 73 (2): pp. 133–153,

(IJCNN 2008) E. Loza, J. Fürnkranz: *Pairwise Learning of Multilabel Classifications with Perceptrons*, Proceedings of the 2008 IEEE International Joint Conference on Neural Networks

(ECML 2008) E. Loza, J. Fürnkranz: *Efficient Pairwise Multilabel Classification for Large-Scale Problems in the Legal Domain*, Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Disocvery in Databases (ECML-PKDD-2008)

(PL 2009) G. Tsoumakas, E. Loza, I. Katakis, S.-H. Park, J. Fürnkranz: *On the Combination of Two Decompositive Multi-Label Classification Methods*. Proceedings of the ECML PKDD 2009 Workshop on Preference Learning

(NC 2010) E. Loza, S.-H. Park, J. Fürnkranz: *Efficient Voting Prediction for Pairwise Multilabel Classification*. In: Neurocomputing, vol. 73 (7-9): pp. 1164 - 1176

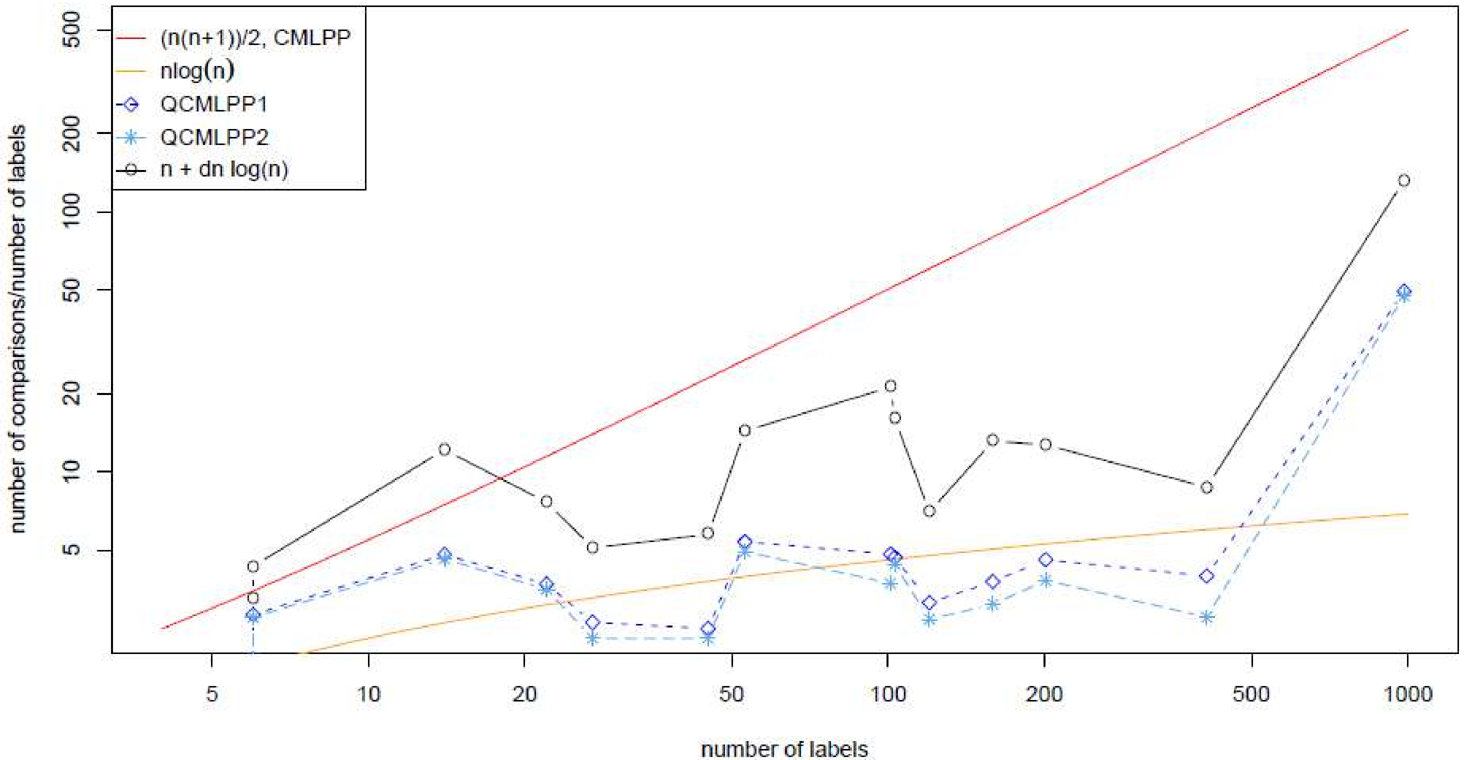
(MLD 2010) E. Loza: *Multilabel Classification in Parallel Tasks*. Working Notes of the 2nd International Workshop on Learning from Multi-Label Data at ICML/COLT 2010

(LWA 2010) *An Evaluation of Multilabel Classification for the Automatic Annotation of Texts.* Proceedings of the LWA 2010: Lernen, Wissen, Adaptivität, Workshop on Knowledge Discovery and Machine Learning (KDML 2010),

(in progress) W. Duivesteijn, E. Loza, J. Fürnkranz, A. Knobbe: *Multi-label LeGo – Enhancing Multi-label Classifiers with Local Patterns*

Backup Slides

QVoting Results: efficiency



- reduces testing from quadratic O(n²) to log-linear O(d n log(n)) in practice

n: labels
d: avg. labelset size
m: training instances

Dual MLPP

		1 epoch					2 epochs			5 epochs			10 epochs		
		FC	MLNB	BR	MMP	DCMLPP	BR	MMP	DCMLPP	BR	MMP	DCMLPP	BR	MMP	DCMLPP
subject matter	IsErr	99.58	99.47	65.99	55.70	51.38	58.78	51.96	44.07	53.42	42.77	38.23	50.19	40.22	36.34
	OneErr	77.83	98.68	35.71	30.58	22.78	27.13	27.09	17.29	22.69	18.38	13.49	20.64	15.97	12.55
	RankLoss	12.89	8.885	17.38	2.303	1.064	13.89	2.520	0.911	11.58	2.091	0.796	9.752	1.85	0.762
	Margin	40.16	25.04	62.31	10.11	4.316	52.28	11.22	3.757	44.77	9.366	3.337	38.45	8.177	3.214
	AvgP	22.57	11.91	59.33	74.01	78.68	66.07	76.95	82.73	70.69	82.10	85.64	73.30	83.75	86.52
	$F1_{\|P\|}$	19.61	1.88	54.79	64.61	70.07	60.79	68.54	74.56	65.64	74.33	78.18	68.45	76.15	79.34
directory code	IsErr	91.51	99.34	52.80	47.68	36.55	46.26	40.01	32.38	40.76	33.28	29.22	37.55	31.39	28.30
	OneErr	90.13	99.04	44.40	40.85	28.22	37.38	32.99	24.42	31.48	25.79	21.41	28.1	23.9	20.65
	RankLoss	14.17	7.446	19.40	2.383	0.972	15.09	2.058	0.863	11.69	1.874	0.824	9.876	1.529	0.815
	Margin	68.33	34.44	96.43	14.18	5.626	77.32	12.18	5.045	61.48	10.95	4.831	52.94	8.947	4.785
	AvgP	18.98	6.714	57.10	68.70	77.89	63.68	74.90	80.87	68.75	79.84	82.87	71.61	81.30	83.38
	$F1_{\|P\|}$	8.47	0.93	49.37	55.08	67.19	56.37	63.29	71.27	61.83	70.00	74.19	64.83	71.86	75.04
EUROVOC	IsErr	99.82	99.82	99.25	99.14	98.20	98.70	98.00	96.75	97.46	96.14		97.06	95.13	
	OneErr	93.52	99.58	53.11	78.98	34.76	44.93	56.88	28.01	36.69	39.46		33.84	34.99	
	RankLoss	12.97	22.34	39.78	3.669	2.692	35.25	4.091	2.398	30.93	4.573		28.59	4.509	
	Margin	1357.10	1623.72	3218.12	562.81	426.28	3040.01	670.65	387.51	2846.47	757.01		2716.63	740.12	
	AvgP	5.504	1.060	25.55	27.04	46.79	30.71	38.42	52.72	35.95	47.65		38.31	50.71	
	$F1_{\|P\|}$	5.646	0.427	28.67	24.76	43.07	33.64	35.16	48.04	38.61	44.24		40.81	47.07	

	1 epoch		2 epochs		10 epochs
	MMP	DCMLPP	MMP	DCMLPP	MMP
AvgP	27.04	46.79	38.42	52.72	50.71
$F1_{\|P\|}$	24.76	43.07	35.16	48.04	47.07

HOMER Experiments