On the Combination of two Decomposive Multi-Label Classification Methods

Grigorios Tsoumakas[1], Eneldo Loza Mencía[2], Ioannis Katakis[1], Sang-Hyeun Park[2], and Johannes Fürnkranz[2]

[1]Aristotle University of Thessaloniki, Greece

[2]Technische Universität Darmstadt, Germany

11 September 2009

Outline

- Introduction
- Background
  - QCLR
  - HOMER
- Evaluation
- Conclusions

Multi-Label Classification

Objects are assigned to a set of labels (domains: text, biology, music etc)

The New York Times

Discovering How Greeks Computed in 100 B.C.

By JOHN NOBLE WILFORD
Published: July 31, 2008

After a closer examination of a surviving marvel of ancient Greek technology known as the Antikythera Mechanism, scientists have found that the device not only predicted solar eclipses but also organized the calendar in the four-year cycles of the Olympiad, forerunner of the modern Olympic Games.

Enlarge This Image

Antikythera Mechanism Research Project

Fragments of the Antikythera Mechanism, an ancient astronomical computer built by the Greeks around 80 B.C. It was found on a shipwreck by sponge divers in 1900, and its exact function still eludes scholars.

The new findings, reported Wednesday in the journal Nature, also suggested that the mechanism's concept originated in the colonies of Corinth, possibly Syracuse, on Sicily. The scientists said this implied a likely connection with Archimedes.

Archimedes, who lived in Syracuse and died in 212 B.C., invented a planetarium calculating motions of the Moon and the known planets and wrote a lost manuscript on astronomical mechanisms. Some evidence had previously linked the complex device of gears and dials to the island of Rhodes and the astronomer Hipparchos, who had made a study of irregularities in the Moon's orbital course.

Methods

A. Problem Adaptation

- Extend algorithms in order to handle multi-label data (e.g. ML*k*NN, BPMLL)

B. Problem Transformation

- Transform the learning task into one or more single-label classification tasks
  - e.g. Label Powerset (LP), Binary Relevance (BR)
- Decompositive Approaches: Focus on large number of labels
  - e.g. HOMER, QCLR

Methods

A. Problem Adaptation

- Extend algorithms in order to handle multi-label data (e.g. ML*k*NN, BPMLL)

B. Problem Transformation

- Transform the learning task into one or more single-label classification tasks
  - e.g. Label Powerset (LP), Binary Relevance (BR)
- Decompositive Approaches: Focus on large number of labels
  - e.g. HOMER, QCLR

Main idea of this work

Combine two state of the art decompositive methods (HOMER + QCLR) in order to confront problems with large number of labels more effectively and efficiently

QWeighted Calibrated Label Ranking (1/4)

Based on Ranking by Pairwise Comparison [Hüllermeier et al., AIJ08]

RPC - Transformation

Learns one binary model for each pair of labels

Ex #	Label set
1	{λ1, λ4}
2	{λ3, λ4}
3	{λ1}
4	{λ2, λ3, λ4}

Ex #	1vs2
1	true
3	true
4	false

Ex #	1vs3
1	true
2	false
3	true
4	false

Ex #	1vs4
2	false
3	true
4	false

Ex #	2vs3
2	false

Ex #	2vs4
1	false
2	false

Ex #	3vs4
1	false

QCLR (2/4)

RPC - Classification

new instance x

1vs2	1vs3	1vs4	2vs3	2vs4	3vs4
1	3	1	3	2	3

Label	Votes
λ1	2
λ2	1
λ3	3
λ4	0

Ranking:

λ3
λ1
λ2
λ4

QCLR (2/4)

RPC - Classification

new instance x

1vs2	1vs3	1vs4	2vs3	2vs4	3vs4
1	3	1	3	2	3

Label	Votes
λ1	2
λ2	1
λ3	3
λ4	0

Ranking:

λ3
λ1
λ2
λ4

How to obtain a bipartition?

Introduce a virtual label λV, that separates positive from negative labels (Calibrated Label Ranking) [Fürnkranz et al., MLJ08]

QCLR (3/4)

CLR - Transformation

Additional pairwise models are necessary

Ex #	1vsV
1	true
2	false
3	true
4	false

Ex #	2vsV
1	false
2	false
3	false
4	true

Ex #	Label set
1	{λ1, λ4}
2	{λ3, λ4}
3	{λ1}
4	{λ2, λ3, λ4}

Ex #	3vsV
1	false
2	true
3	false
4	true

Ex #	4vsV
1	true
2	true
3	false
4	true

Ex #	1vs2
1	true
3	true
4	false

Ex #	1vs3
1	true
2	false
3	true
4	false

Ex #	1vs4
2	false
3	true
4	false

Ex #	2vs3
2	false

Ex #	2vs4
1	false
2	false

Ex #	3vs4
1	false

QCLR (4/4)

CLR - Classification

new instance x

1vs2	1vs3	1vs4	2vs3	2vs4	3vs4
1	1	1	2	2	4

1vsV	2vsV	3vsV	4vsV
1	V	V	V

Label	Votes
λ1	4
λ2	2
λ3	0
λ4	1
λV	3

Ranking: λ1, λV, λ2, λ4, λ3

Limitation: Need to query quadratic number of models

Solution : Quick Weighted Voting [Loza Mencía et al., ESANN09]

- Complexity is $n + dnlog(n)$, where n is the number of labels and d is the average number of relevant labels (cardinality)

HOMER - Hierarchy Of MultiLabel ClassifiERs (1/2)

Main Idea [Tsoumakas et al., ECMLPKDD08w]

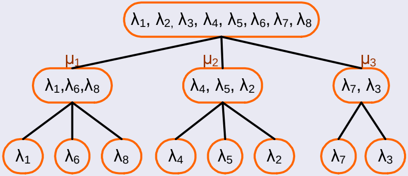
The transformation of a multi-label problem with large number of labels into many hierarchically structured simpler sub-problems

HOMER - Hierarchy Of MultiLabel ClassifiERs (1/2)

Main Idea [Tsoumakas et al., ECMLPKDD08w]

The transformation of a multi-label problem with large number of labels into many hierarchically structured simpler sub-problems

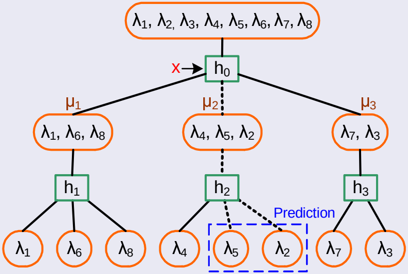
Step 1. Hierarchical Organization of Labels



- k: branching factor
- meta label μ_n : represents the union of the labels of the node

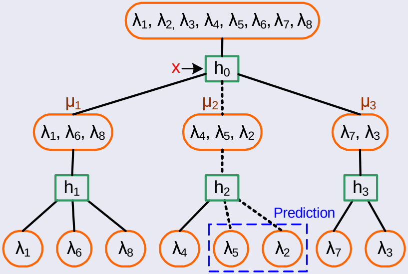
HOMER - Hierarchy Of MultiLabel ClassifiERs (2/2)

Step 2. Assign a Multilabel Classifier at each internal node



HOMER - Hierarchy Of MultiLabel ClassifiERs (2/2)

Step 2. Assign a Multilabel Classifier at each internal node



Advantages

1. Classification Time - Only invoke few classifiers of the hierarchy
2. Prediction Performance - Balanced examples for each classifier
3. Training Time - Smaller datasets at each node

Label Distribution (1/2)

Open Issue

How should we distribute labels into k children nodes (groups)?

Label Distribution (1/2)

Open Issue

How should we distribute labels into k children nodes (groups)?

Criteria

1. Labels of a group should co-occur as much as possible
   - Prediction of less meta-labels ⇒ activation of less classifiers ⇒ small classification times
2. Groups should be of equal size
   - Balanced distribution of examples for each meta-label ⇒ improved predictive performance
   - A balanced tree could lead to improved classification times

Label Distribution (2/2)

Balanced k-Means

- Extension of k-Means
- Equal sized clusters
- Maintain an ordered list of labels according to similarity with the cluster centroid
- In case a cluster overflows $\Rightarrow$ move the most distant label into the next most similar group
- Hamming distance

Motivation of Combination

Why combine HOMER with QCLR?

1. QCLR+HOMER will require less
   - memory
   - time for training
   - time for classification
2. HOMER+QCLR will have higher predictive performance (e.g. compared to using binary relevance at each node)

Evaluation Goals

Primary Questions

1. Can HOMER improve QCLR in terms of predictive performance, training and classification time?
2. Can HOMER+QCLR outperform HOMER+BR in terms of predictive performance?
   - And what will be the extra cost in training and classification times?

Evaluation Goals

Primary Questions

1. Can HOMER improve QCLR in terms of predictive performance, training and classification time?
2. Can HOMER+QCLR outperform HOMER+BR in terms of predictive performance?
   - And what will be the extra cost in training and classification times?

Secondary Questions

1. What is the effect of the distribution method in HOMER?
   - Clustering? Balanced Clustering? Random Distribution?
2. What is the effect of branching factor k?

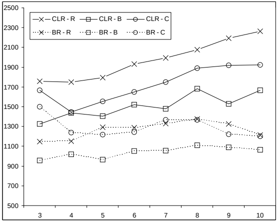
Experimental Setup

- Methods
  - Base single-label classifier: C4.5
  - Base multi-label classifiers: BR, QCLR
  - HOMER: H+BR, H+QCLR
  - Partitioning: Balanced *k*-Means (B), EM (C), Random (R)
    - Number of partitions ranging from 3 to 10
- Datasets

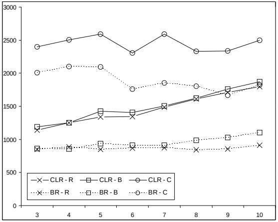
name	train	test	features	labels	cardinality	density	labelsets
HiFind	16452	16519	98	632	37.304	0.059	32734
eccv2002	42379	4686	36	374	3.525	0.009	3175
jmlr2003	48859	16503	46	153	3.071	0.020	3115
mediamill	30993	12914	120	101	4.376	0.043	6555

- Software
  - Mulan - http://sourceforge.net/projects/mulan/

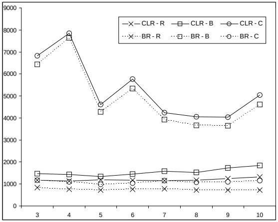
The Clustering Factor - Training Time



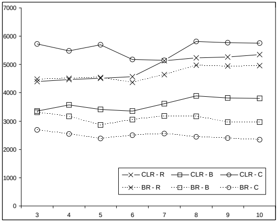
(a) mediamill



(b) jmlr2003

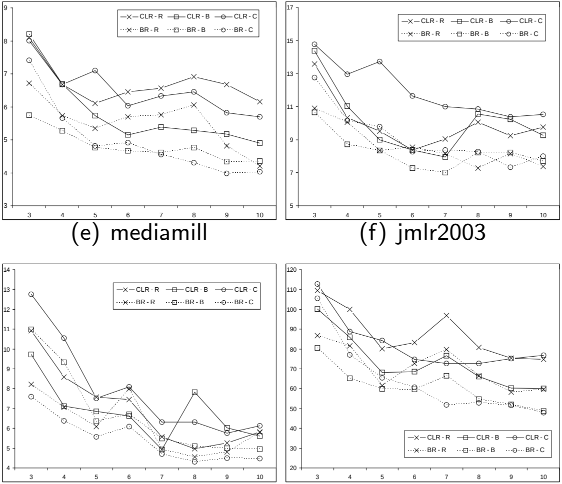


(c) eccv2002



(d) HiFind

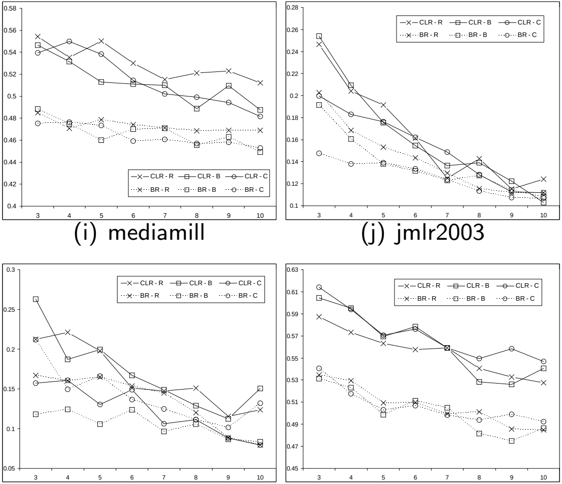
The Clustering Factor - Classification Time



(e) mediamill (f) jmlr2003

(g) eccv2002 (h) HiFind

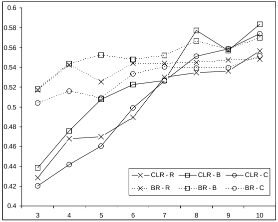
The Clustering Factor - Recall



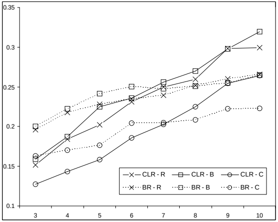
(i) mediamill (j) jmlr2003

(k) eccv2002 (l) HiFind

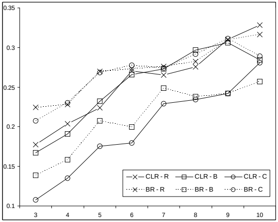
The Clustering Factor - Precision



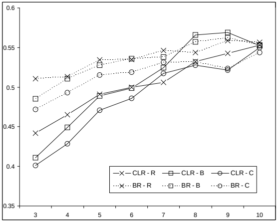
(m) mediamill



(n) jmlr2003

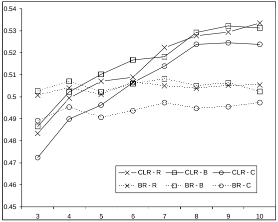


(o) eccv2002

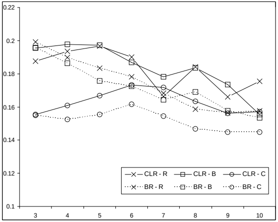


(p) HiFind

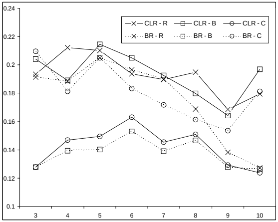
The Clustering Factor - micro F



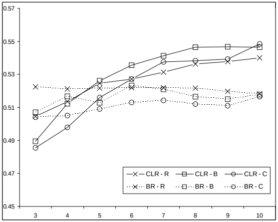
(q) mediamill



(r) jmlr2003



(s) eccv2002



(t) HiFind

The Clustering Factor - Observations

Increasing k leads to ...

- Better classification times (shorter tree of classifiers)
- Better precision
- Worse recall

Compared to random partitioning, balanced clustering takes advantage of similarity and can lead to lower (overall) training/classification time, especially for dense datasets

micro F1

Method	mediamill	jmlr2003	eccv2002	HiFind
BR	50.55 %	15.09 %	12.34 %	51.65 %
QCLR	55.04 %	8.45 %	7.21 %	–
H+BR	50.23 %	15.36 %	18.14 %	51.76 %
H+QCLR	53.13 %	15.55 %	19.70 %	54.65 %

- HOMER improves predictive performance of BR and QCLR
  - Especially in datasets with large number of labels
- HOMER+QCLR presents better predictive performance than HOMER+BR

Training Time

METHOD	MEDIAMILL	JMLR2003	ECCV2002	HIFIND
BR	2413.40	2801.17	2701.32	4179.66
QCLR	7423.19	6542.51	7460.14	–
H+BR	1065.21	1101.61	1144.47	2345.39
H+QCLR	1667.29	1871.00	1836.34	3801.53

- HOMER reduces training time for both BR and CLR

Testing Time

METHOD	MEDIAMILL	JMLR2003	ECCV2002	HIFIND
BR	3.84	6.67	5.47	50.47
QCLR	103.59	119.28	154.65	–
H+BR	4.35	7.70	4.48	48.77
H+QCLR	4.90	9.26	5.62	60.02

- HOMER significantly reduces testing time for QCLR

Conclusions & Future Work

Conclusions

A combination of decompositive methods (HOMER and QCLR)

- Builds less number of models compared to QCLR
  - Faster training
  - Faster testing
  - Less memory requirements
- Better predictive performance than QCLR
- Better predictive performance than HOMER+BR with a small expense in training and classification time

Future Work

- In depth analysis of when and why HOMER+QCLR works
- More datasets
- More base classifiers

End of presentation

Thank you for your attention!