

Mulan: A Java Library for Multi-Label Learning

G. Tsoumakas, E. Spyromitros, J. Vilcek, I. Vlahavas

Workshop on Machine Learning Open Source Software 2010



International
Conference on
Machine
Learning

Haifa, Israel
June 21 - 24



What is Multi-Label Learning

Training Set of News Articles

Training Attributes										Semantic Categories				
Frequent Words						Image Features				Politics	Sports	Culture	Science	Health
1	0	0	1	1	0	0.3	0.5	0.2	0.8	True	False	True	False	False
0	0	1	1	1	0	0.6	0.9	0.2	0.5	False	True	True	True	False
1	1	0	0	0	1	0.4	0.7	0.1	0.6	True	False	False	False	True

Test Article

World Cup officials asked to ban Vuvuzela noise from games



Wednesday, people around the world are asking what is a **Vuvuzela** and why does it sound so annoying? **World Cup** officials have been asked to ban the Vuvuzela noise makers from the World Cup games in **Africa**, and they have said they will not.

Applications

- ✓ **semantic annotation of images and video**
- ✓ **web page categorization**
- ✓ **direct marketing**
- ✓ **functional genomics**
- ✓ **music categorization into genres & emotions**
- ✓ **tag recommendation in Web 2.0 systems**
- ✓ **...**

Mulan at a Glance

- **It is built on top of Weka**
 - **Established code**
 - **Large user base**
- **It contains**
 - **A variety of state-of-the-art algorithms for multi-label learning**
 - **Basic dimensionality reduction methods for multi-label data**
 - **An extensive evaluation framework**
 - **Experiments package (new)**

Data Format

ARFF file

```
@relation MultiLabelExample

@attribute feature1 numeric
@attribute feature2 {0,1}
@attribute amazed {0, 1}
@attribute happy {0, 1}
@attribute relaxing {0, 1}
@attribute sad {0, 1}

@data
2.3,5.6,1.4,0,1,1,0,0
```

XML file

```
<?xml version="1.0"
  encoding="utf-8"?>
<labels xmlns="...">
  <label name="amazed" />
  <label name="happy" />
  <label name="relaxing"/>
  <label name="sad" />
</labels>
```

Hierarchies of Labels

XML file with label hierarchies

```
<?xml version="1.0" encoding="utf-8"?>
<labels
xmlns="http://mulan.sourceforge.net/labels">
  <label name="sports">
    <label name="football" />
    <label name="basketball" />
  </label>
  <label name="arts">
    <label name="sculpture" />
    <label name="photography" />
  </label>
</labels>
```

Transformation Based ML-Learners

- Package `mulan.classifier.transformation`
 - `TransformationBasedMultiLabelLearner`
 - Abstract base class for methods that convert a multi-label learning problem to one or more single-label learning problems
 - Implemented algorithms
 - `BinaryRelevance`
 - `CalibratedLabelRanking`
 - `ClassifierChains`
 - `EnsembleOfPrunedSets`
 - `IncludeLabelsClassifier`
 - `LabelPowerset`
 - `MultiLabelStacking`
 - `PPT`
 - `PrunedSets`

Algorithm Adaptation ML-Learners

- Package `mulan.classifier.lazy`
 - `MultiLabelKNN`
 - Abstract base class for kNN based methods
 - Implemented algorithms
 - `MLkNN`
 - `BRkNN`
 - `IBLR_ML`
- Package `mulan.classifier.neural`
 - `BPMLL`
 - `MMPLEarner`
 - Package `mulan.classifier.neural.model`
 - Support classes

Multi-Label Meta Learners

- Package `mulan.classifier.meta`
 - `MultiLabelMetaLearner`
 - Abstract base class for multi-label learners which use other multi-label learners
 - Implemented algorithms
 - `RAkEL`
 - `RAkELd`
 - `HOMER`
 - `HMC`
 - `ClusteringBased`
- Package `mulan.classifier.meta.thresholding`
 - `OneThreshold`

An example

```
MultiLabelInstances train, test;

train = new
MultiLabelInstances("train.arff", "format.xml");
test = new
MultiLabelInstances("test.arff", "format.xml");

Classifier base = new J48();
BinaryRelevance br = new BinaryRelevance(base);
br.build(train);

Evaluator eval = new Evaluator();
Evaluation results = eval.evaluate(br, test);
System.out.println(results.toString());
```

Sample output

Hamming Loss: 0.267

Subset Accuracy: 0.177

Example-Based Precision: 0.585

Example-Based Recall: 0.577

Example-Based F Measure: 0.581

Example-Based Accuracy: 0.456

Micro-averaged Precision: 0.600

Micro-averaged Recall: 0.582

Micro-averaged F-Measure: 0.591

Macro-averaged Precision: 0.586

Macro-averaged Recall: 0.571

Macro-averaged F-Measure: 0.577

One-Error: 0.405

Average Precision: 0.686

Is-Error: 0.729

Error Set Size: 2.392

Coverage: 2.780

Ranking Loss: 0.319

Mean Average Precision: 0.509

Micro-averaged AUC: 0.690

Macro-averaged AUC: 0.681



Thanks for your attention!

See you in the poster session!