Multilabel Attribute Selection

naive merging

Sebastian Lehnerer
lehnerer@in.tum.de

Technische Universität München

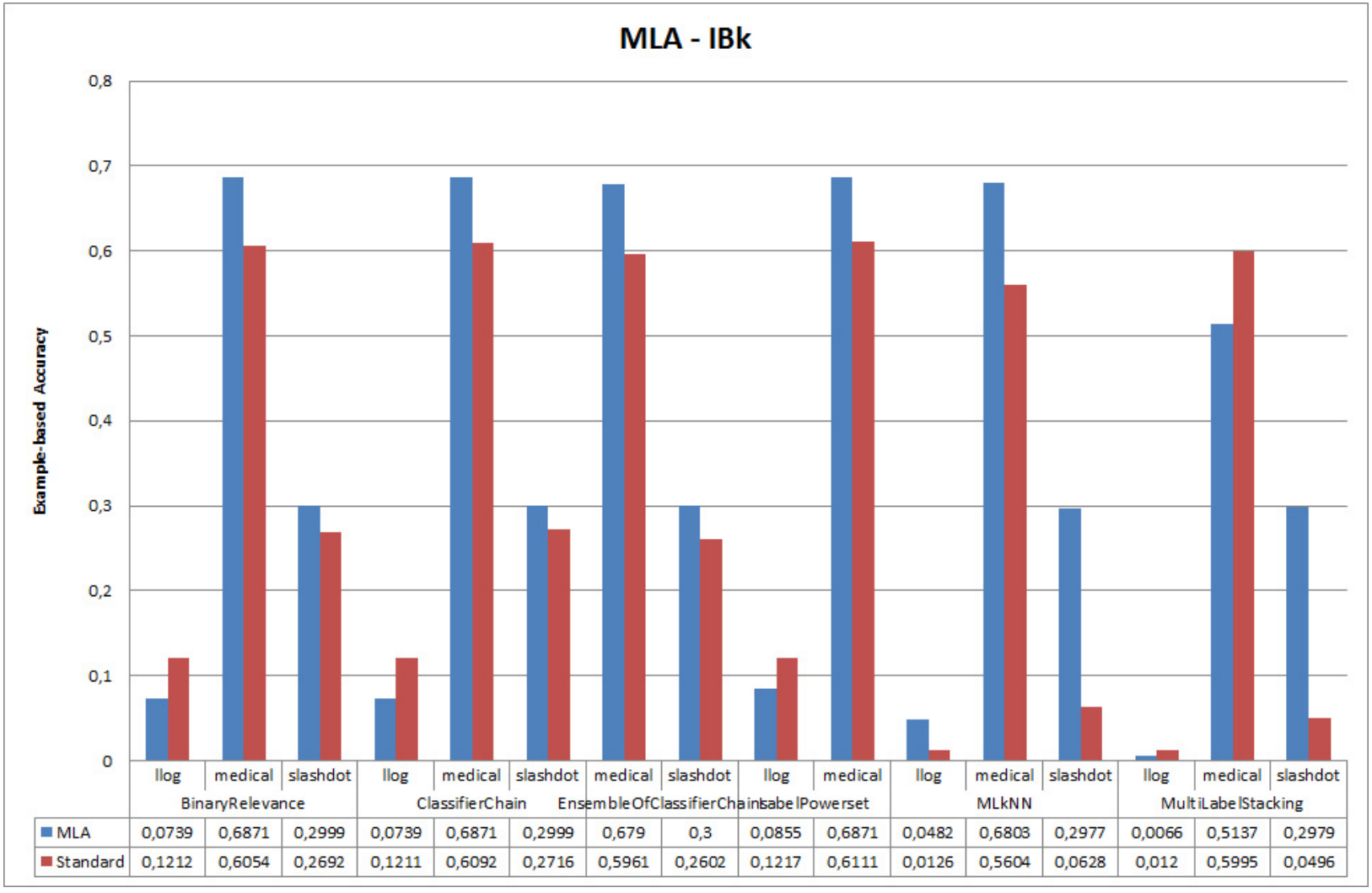
June 14, 2011

- feature selection for every label, where other labels are treated as normal features

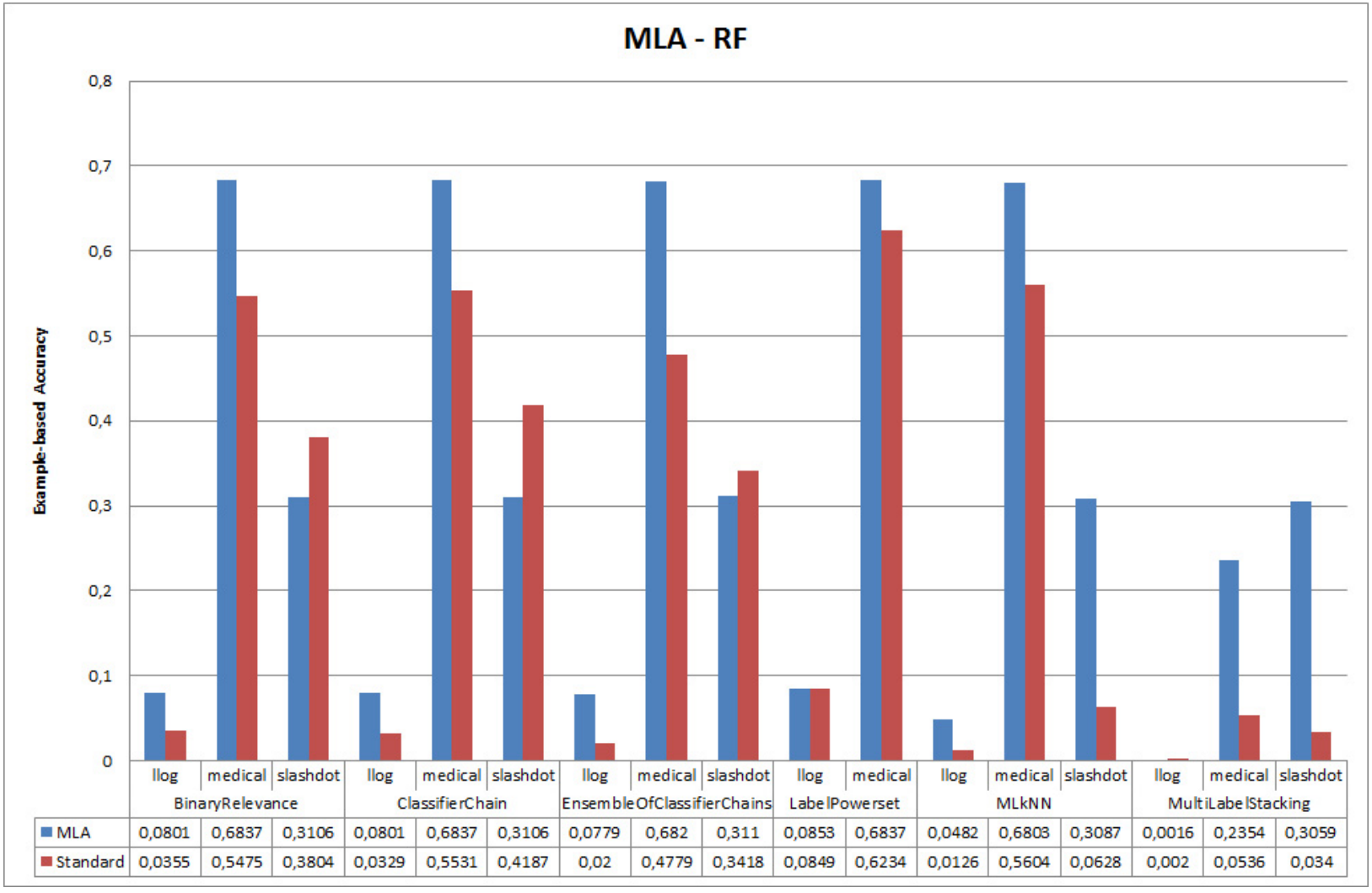
$$
\begin{aligned}
Y_1 &\leftarrow \{X_1 \ldots X_n \cup Y_2 \ldots Y_n | X_i, Y_i \in \{0,1\}\} \\
Y_2 &\leftarrow \{X_1 \ldots X_n \cup Y_1, Y_3 \ldots Y_n | X_i, Y_i \in \{0,1\}\} \\
&\vdots \\
Y_n &\leftarrow \{X_1 \ldots X_n \cup Y_1 \ldots Y_{n-1} | X_i, Y_i \in \{0,1\}\}
\end{aligned}
$$

- for every label, other label attributes in the corresponding feature set will be replaced by its feature set, until no labels remain the feature set. The elimited label are stored in a labelset which results in a map between label set and feature set
- **function** FINDLABELFEATURESETS(data)
  $groups \leftarrow Map < labelSet, featureSet >$
  for $L \in labelSet(data)$ **do**
  if $L \notin keys(groups)$ **then**
  $currentLabelSet \leftarrow \{L\}$
  $currentFeatureSet \leftarrow selectFeatures(L, data)$
  while $\exists \{F \in currentFeatureSet \wedge F \in labelSet(data)\}$ **do**
  $currentLabelSet \cap \{F\}$
  $currentFeatureSet \cap \{selectFeatures(F, data)\}$
  $remove(F, currentFeatureSet)$
  return *groups*

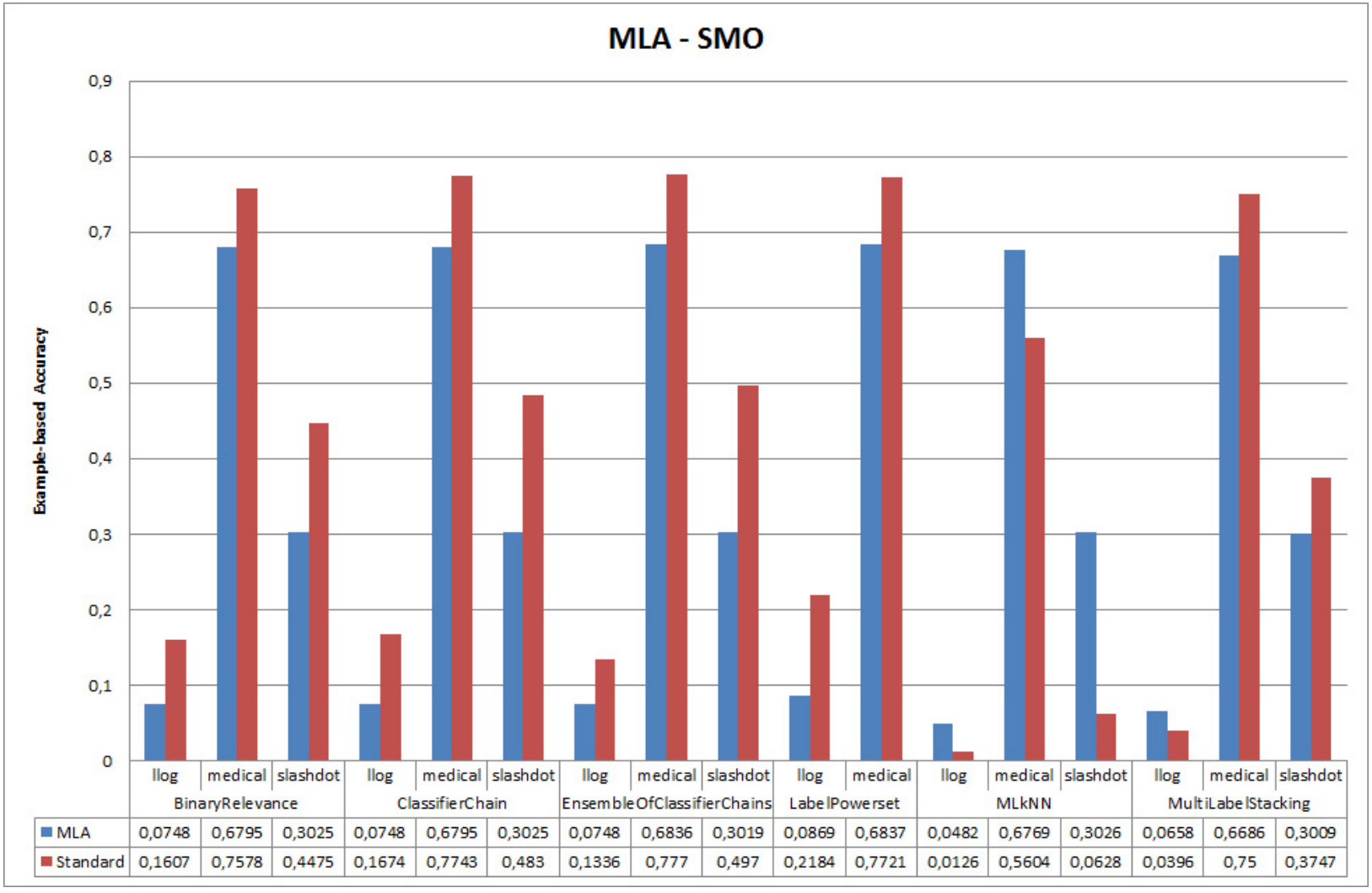
- to avoid similiar label or feature-sets a second-step merge function is applied
- **function** BACKTRACE(groups)
  for *labelSet* ∈ *keys*(*groups*) **do**
  for *labelSet*2 ∈ *keys*(*groups*) **do**
  if *labelSet*2 ⊂ *labelSet* **then**
  featureSet ← *groups*(*labelSet*)
  *featureSet*2 ← *groups*(*labelSet*2)
  groups(*labelSet*) ← *featureSet* ∩ *featureSet*2
  remove(*labelSet*2, *groups*)
  else if *labelSet* ⊂ *labelSet*2 **then**
  featureSet ← *groups*(*labelSet*)
  *featureSet*2 ← *groups*(*labelSet*2)
  groups(*labelSet*2) ← *featureSet* ∩ *featureSet*2
  remove(*labelSet*, *groups*)
  else if *groups*(*labelSet*) ⊂ *groups*(*labelSet*2) **then**
  groups(*labelSet* ∩ *labelSet*2) ← *groups*(*labelSet*2)
  remove(*labelSet*, *groups*)
  remove(*labelSet*2, *groups*)
  else if *groups*(*labelSet*2) ⊂ *groups*(*labelSet*) **then**
  groups(*labelSet* ∩ *labelSet*2) ← *groups*(*labelSet*)
  remove(*labelSet*, *groups*)
  remove(*labelSet*2, *groups*)
  return *groups*



	BinaryRelevance			ClassifierChain			EnsembleOfClassifierChains		LabelPowerset		MLkNN			MultiLabelStacking		
	llog	medical	slashdot	llog	medical	slashdot	medical	slashdot	llog	medical	llog	medical	slashdot	llog	medical	slashdot
MLA	0,0739	0,6871	0,2999	0,0739	0,6871	0,2999	0,679	0,3	0,0855	0,6871	0,0482	0,6803	0,2977	0,0066	0,5137	0,2979
Standard	0,1212	0,6054	0,2692	0,1211	0,6092	0,2716	0,5961	0,2602	0,1217	0,6111	0,0126	0,5604	0,0628	0,012	0,5995	0,0496



	llog	medical	slashdot	llog	medical	slashdot	llog	medical	slashdot	llog	medical	llog	medical	slashdot	llog	medical	slashdot
	BinaryRelevance			ClassifierChain			EnsembleOfClassifierChains			LabelPowerset		MLkNN			MultiLabelStacking		
MLA	0,0801	0,6837	0,3106	0,0801	0,6837	0,3106	0,0779	0,682	0,311	0,0853	0,6837	0,0482	0,6803	0,3087	0,0016	0,2354	0,3059
Standard	0,0355	0,5475	0,3804	0,0329	0,5531	0,4187	0,02	0,4779	0,3418	0,0849	0,6234	0,0126	0,5604	0,0628	0,002	0,0536	0,034



	llog	medical	slashdot	llog	medical	slashdot	llog	medical	slashdot	llog	medical	llog	medical	slashdot	llog	medical	slashdot
	BinaryRelevance			ClassifierChain			EnsembleOfClassifierChains			LabelPowerset		MLkNN			MultiLabelStacking		
MLA	0,0748	0,6795	0,3025	0,0748	0,6795	0,3025	0,0748	0,6836	0,3019	0,0869	0,6837	0,0482	0,6769	0,3026	0,0658	0,6686	0,3009
Standard	0,1607	0,7578	0,4475	0,1674	0,7743	0,483	0,1336	0,777	0,497	0,2184	0,7721	0,0126	0,5604	0,0628	0,0396	0,75	0,3747

example cluster characteristics (∅ over folds)

- llog
  ∅ number of clusters : 3.8
  ∅ number of cluster ($>$ 2 labels) : 3..6
  ∅ number of labels per cluster: 19,7
- medical
  ∅ number of clusters : 13.6
  ∅ number of cluster ($>$ 2 labels) : 4
  ∅ number of labels per cluster: 3.3
- slashdot
  ∅ number of clusters : 16.2
  ∅ number of cluster ($>$ 2 labels) : 0.8
  ∅ number of labels per cluster: 1.4