MALEF:

A Framework for Agent-Based Distributed Machine Learning and Data Mining

Jan Tožička[1], Michael Rovatsos[2], and Michal Pěchouček[1]

[1] Agent Technology Group, Gerstner Laboratory,
Czech Technical University in Prague

[2] School of Informatics, The University of Edinburgh,
Edinburgh, United Kingdom

http://agents.felk.cvut.cz/

Introduction

Objective: To develop a generic agent-based framework for collaborative machine learning and data mining

Learners:

- autonomous
- self-directed
- individual learning goals
- private knowledge

Interaction mechanism should allow agents to:

1. **exchange knowledge**
2. **decide what knowledge to share**
3. **reason about how to use received knowledge**

Learning Problem

Learning problem:

$$D \subseteq \mathcal{D}, \mathcal{D} \leadsto h \in \mathcal{H}$$

Performance measure:

$$g : \mathcal{H} \to \mathcal{Q}$$

Clustering:

- Learning data:

$$\mathcal{D} = \times_{i=1}^{n} [A_i]$$

- Hypothesis space:

$$\mathcal{H} \subseteq \{h | h : \mathcal{D} \to \mathbb{N}, h \text{ is total with range } \{1, \ldots, k\}\}$$

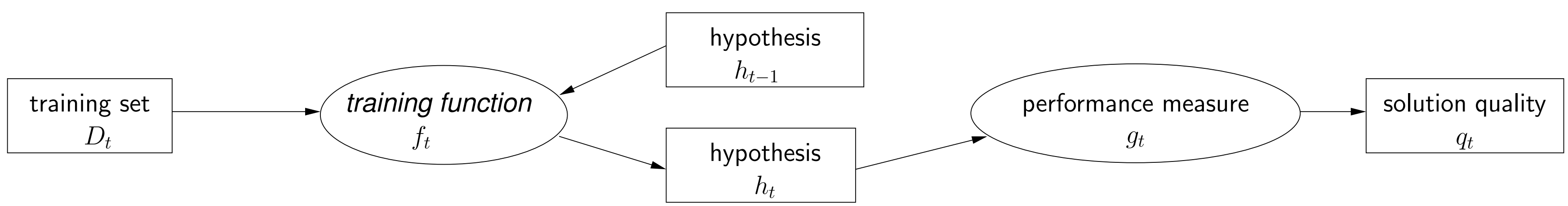
Definitions

Data set:

$$D = \langle d_1, \ldots d_k \rangle$$

Training function producing h:

$$f : \mathcal{H} \times \mathcal{D}^* \rightarrow \mathcal{H}$$



Learning step:

$$l = \langle D, H, f, g, h \rangle,$$

where $H \subseteq \mathcal{H}$, $h \in H$ and $l \in L$

Integration Matrix

Let $l_j = \langle D_j, H_j, f_j, g_j, h_j \rangle$ be the current "state" of agent j when receiving a learning process description $l_i = \langle D_i, H_i, f_i, g_i, h_i \rangle$ from agent i.

$i \backslash j$	D_j	H_j	f_j	g_j	h_j
D_i	$p_1^{D\to D}(D_i, D_j)$ $\vdots$ $p_{k_{D\to D}}^{D\to D}(D_i, D_j)$	$\ldots$	$\ldots$	n/a	$\ldots$
H_i	$\vdots$	$\ddots$		n/a	
f_i	$\vdots$		$\ddots$	n/a	
g_i	$\vdots$			n/a	$p_1^{g\to h}(g_i, h_j)$ $\vdots$ $p_{k_{g\to h}}^{g\to h}(g_i, h_j)$
h_i	$\vdots$			n/a	$\ddots$

Diagonal contains most common ways of integration including replacing c_j by c_i or ignoring c_i.

Learning Process Modifications

Modification of D_j:

- append D_i to D_j; filter out all elements from D_j which also appear in D_i; append D_i to D_j discarding all elements with attributes outside ranges which affect g_j, or those elements already correctly classified by h_j;

Modification of H_i:

- use the union/intersection of H_i and H_j; alternatively, discard elements of H_j that are inconsistent with D_j in the process of intersection or union, or filter out elements that cannot be obtained using f_j (unless f_j is modified at the same time)

Learning Process Modifications

Modification of f_j:

- modify parameters or background knowledge of f_j using information about f_i; assess their relevance by simulating previous learning steps on D_j using g_j and discard those that do not help improve own performance

Modification of h_j:

- combine h_j with h_i using (say) logical or mathematical operators; make the use of h_i contingent on a "pre-integration" assessment of its quality using own data D_j and g_j

No modification of g_j is allowed.

Knowledge Combining Operators (Examples)

Modification of D_j using f_i

- pre-process samples in f_i, e.g. to get intermediate representations that f_j can be applied to

Modification of D_j using h_i

- filter out samples from D_j that are covered by h_i and build h_j using f_j only on remaining samples

Modification of H_j using f_i

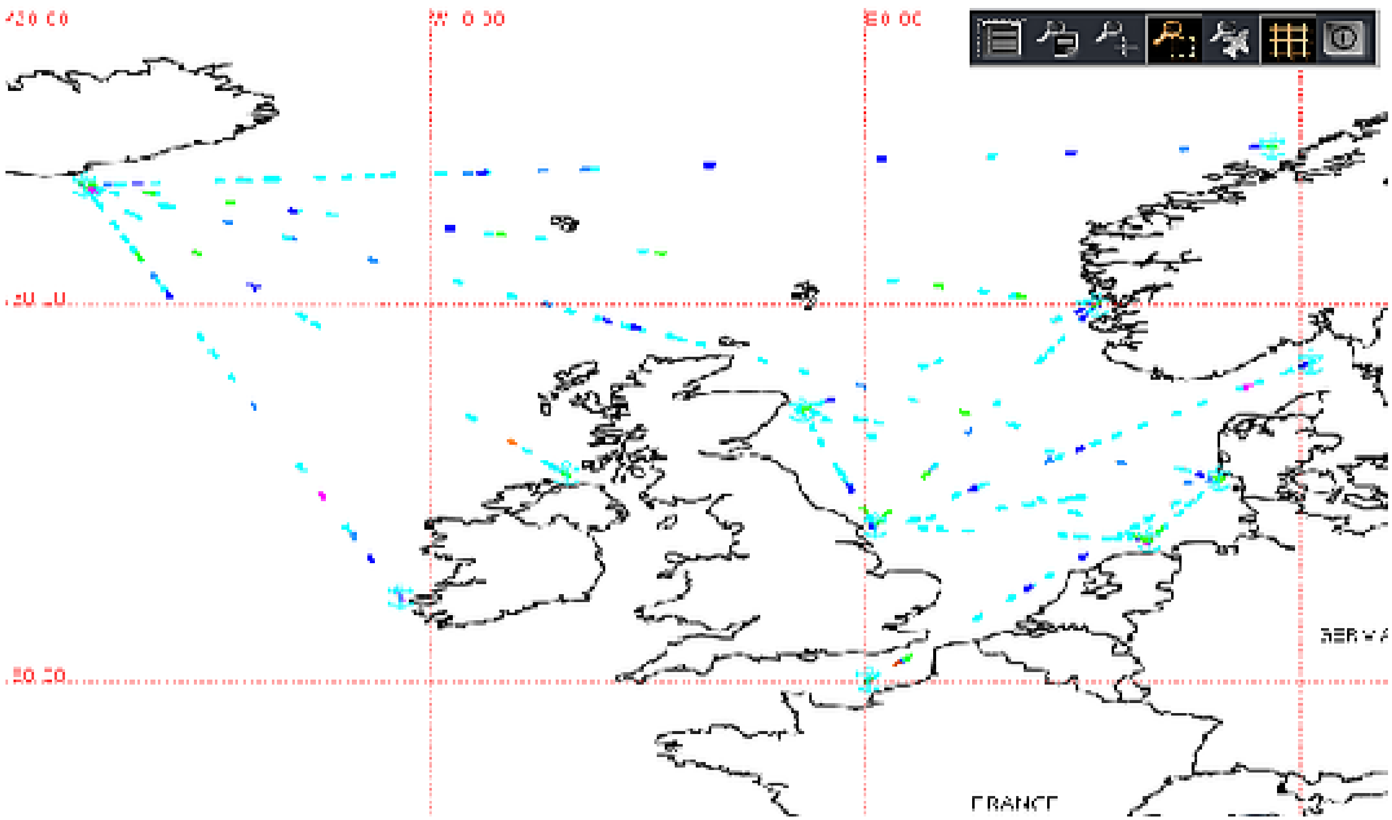
- filter out hypotheses from H_j that are not realisable using f_i

Modification of h_j using g_i

- if h_j is composed of several sub-components, filter out those sub-components that do not perform well according to g_i

AIS Domain description

Detection of unusual, potentially suspicious ships based on provided AIS data.



Agent-Based Distributed Learning System Design

To describe a concrete design for the AIS domain, we need to specify:

1. The datasets and clustering **algorithms** available to individual agents
2. The **interaction mechanism** used for exchanging descriptions of learning processes
3. The **decision mechanism** agents apply to make learning decisions

1. Available Datasets and Clustering Algorithms

Datasets:

- each agent has private dataset containing vessel descriptions: $A = \{1, \ldots, n\}$

Learning algorithms – clustering with a fixed number of k clusters:

- k-means
- k-medoids

Hypothesis space:

- $\mathcal{H} = \{\langle c_1, \ldots, c_k \rangle | c_i \in \mathbb{R}^{|A|}\}$
- For each hypothesis $h \in \mathcal{H}$ and any data point $d \in \times_{i=1}^{n} [A_i]$, the closest cluster c_i is choosen.

1. Available Datasets and Clustering Algorithms

Evaluation:

- **Validation set** V_i and generated **fake vessels** F_i such that $|F_i| = |V_i|$
- **Confidence value** $r(h, d)$ for ship d:

$$r(h, d) = \frac{1}{|d - c_{C(h,d)}|}$$

- a vessel in $F_i \cup V_i$ is classified as fake if its r-value is below the median of all the confidences $r(h, d)$ for $d \in F_i \cup V_i$
- **Quality**: quality $g_i(h) \in \mathbb{R}$ as the ratio between all correctly classified vessels and all vessels in $F_i \cup V_i$.

2. The Interaction Mechanism Used for LPD Exchange

We use a simple Contract-Net Protocol based on **hypothesis trading** mechanism:

1. **Initiator** of a CNP describes its own current learning state as $(*, *, *, g_i(h), *)$ and sends CfPs.
2. **Participants** may propose the quality of their own model.
3. If the bids (if any) are **accepted** by the initiator, the agents exchange their hypotheses

3. The Decision Mechanism Making Learning Decisions

Having own model h, other's model h' is **accepted** if:

$$g(h') > g(h), \text{ or with probability } P\left(g(h')/g(h)\right)$$

Model merging operators:

- $p^{h \to h}(h_i, h_j)$:
  - m-**join:** The m best clusters (in terms of coverage of D_j) from hypothesis h_i are appended to h_j.
  - m-**select:** The set of the m best clusters (in terms of coverage of D_j) from the union $h_i \cup h_j$ is chosen as a new model.
- $p^{h \to D}(h_i, D_j)$:
  - m-**filter:** The m best clusters (as above) from h_i are identified and appended to a new model formed by using those samples not covered by these clusters applying f_j.

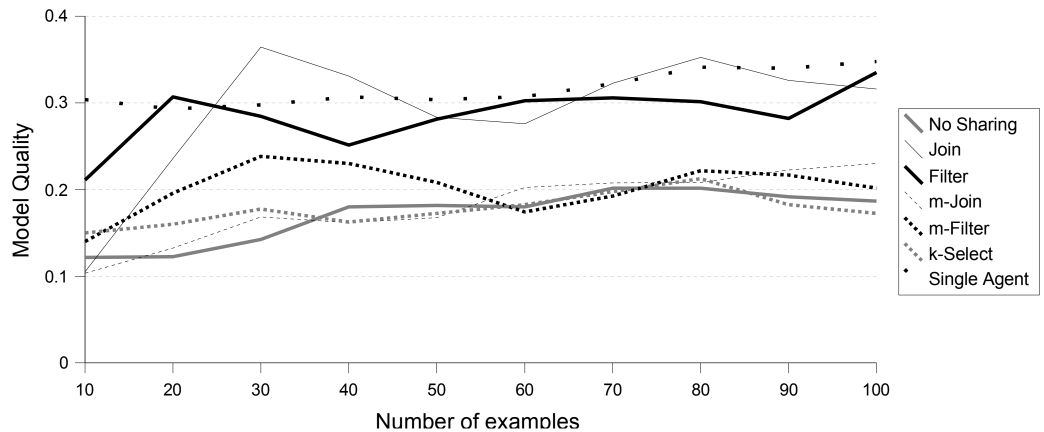
Experimental Results

Case description:

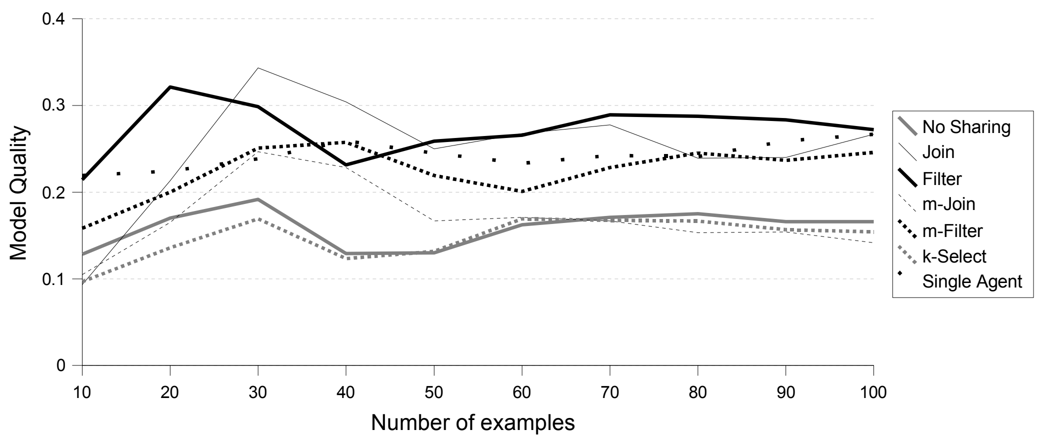
- 3 learning agents
- k-means and k-medoids learning methods
- dataset of 300 ships into three disjoint sets of 100 samples each and assign each of these to one learning agent
- **Single Agent** is learning from the whole dataset
- $k = 10$ (Davies-Bouldin index)
- $m = k$ for m-select
- $m = 3$ for m-join and m-filter
- **homogeneous** vs. **heterogeneous** learner societies

Results: Homogeneous Learners

k-means

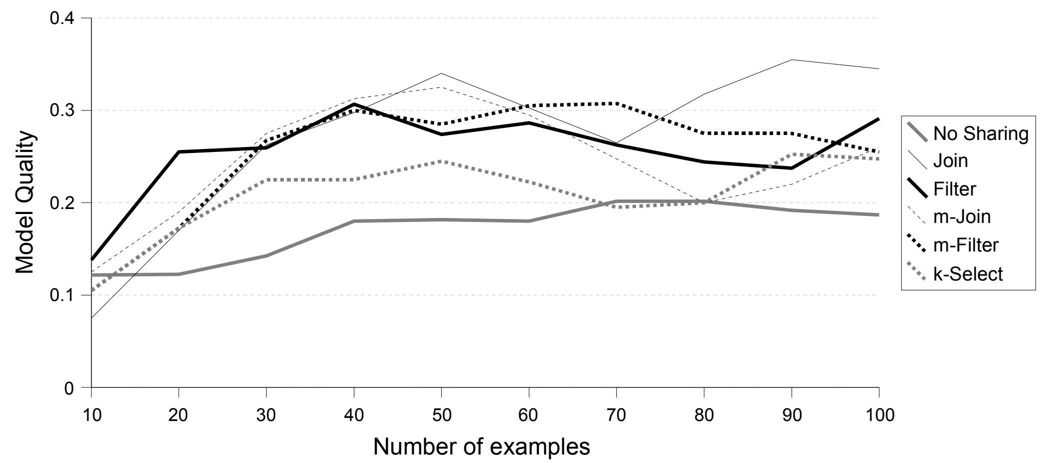


k-medoids

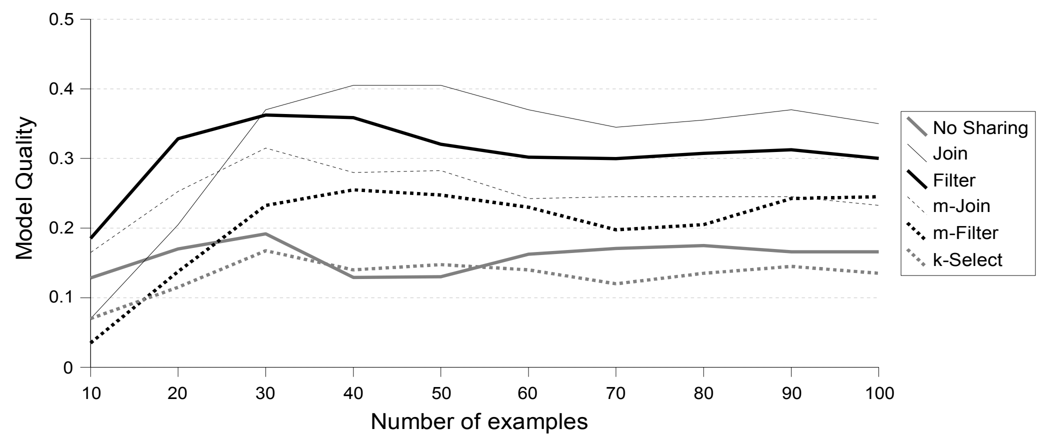


Results: Heterogeneous Learners

2 k-**means** agents and 1 k-medoids agent



2 k-**medoids** agents and 1 k-means agent



Performance

The m-**filter** operation, decreases the number of learning samples and thus can **speed up** the learning process.

	k-means	**k-medoids**
filtering	30-40 %	10-20 %
m-**filtering**	20-30 %	5-15 %

Conclusion

Even a very simplistic application of **MALEF** architecture has proven capable of **improving the performance** of individual learning agents.

Conclusion

MALEF

- abstract distributed machine learning and data mining framework
- requires very general learning architecture only
- captures complex forms of interaction between **heterogeneous** or **self-interested** learners
- allows learners to improve their learning abilities with information provided by other learners
- allows to exchange and integrate different types of knowledge

Acknowledgement

Research supported by Army Research Laboratory project N62558-03-0819 and Office for Naval Research project N00014-06-1-0232.

Thank you.