# Context-Aware Pattern Recognition

## Marcello Pelillo

University of Venice, Italy

Artificial Intelligence

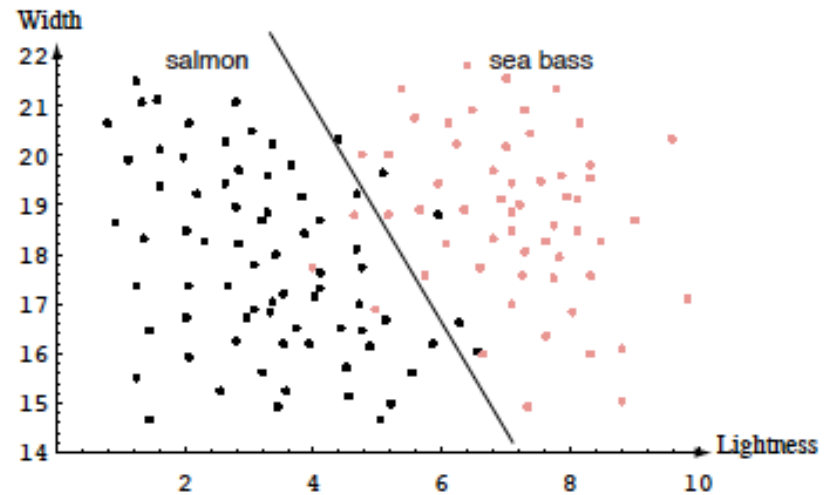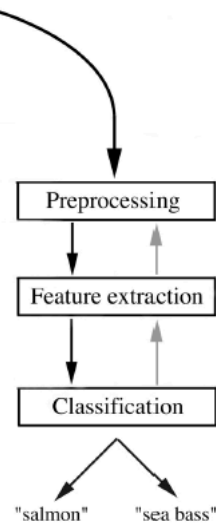*a.y. 2018/19*

Ca' Foscari University of Venice

eclt
European Centre for Living Technology

# Pattern recognition:
## The standard paradigm



From: Duda, Hart and Stork, *Pattern Classification* (2000)

# Limitations

There are cases where it's not easy to find satisfactory feature-vector representations.

**Some examples**

✓ when experts cannot define features in a straightforward way
✓ when data are high dimensional
✓ when features consist of both numerical and categorical variables,
✓ in the presence of missing or inhomogeneous data
✓ when objects are described in terms of structural properties
✓ …

# Tacit assumptions

1. Objects possess "intrinsic" (or essential) properties

2. Objects live in a vacuum

*In both cases:*
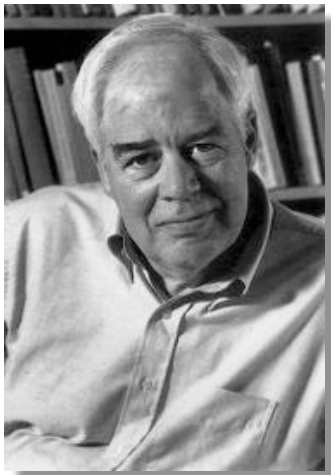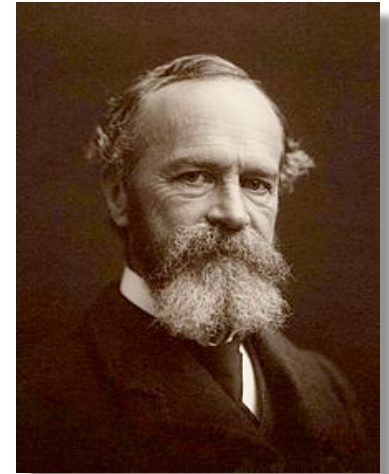
***Relations are neglected!***

# Taking it to the extreme ...

«There is no property ABSOLUTELY essential to any one thing. The same property which figures as the essence of a thing on one occasion becomes a very inessential feature upon another.»

William James
*The Principles of Psychology* (1890)

«There are, so to speak, relations all the way down, all the way up, and all the way out in every direction: you never reach something which is not just one more nexus of relations.»

Richard Rorty
*A World without Substances or Essences* (1994)

# The many types of relations

- Similarity relations between objects

- Similarity relations between categories

- Contextual relations

- …

**Application domains:** Natural language processing, computer vision, computational biology, adversarial contexts, social signal processing, medical image analysis, social network analysis, network medicine, …

# Context helps …

c   →   <u>c</u>at

   →   <u>c</u>ir<u>c</u>us

i   →   s<u>i</u>n

   →   f<u>i</u>ne

e   →   r<u>e</u>d
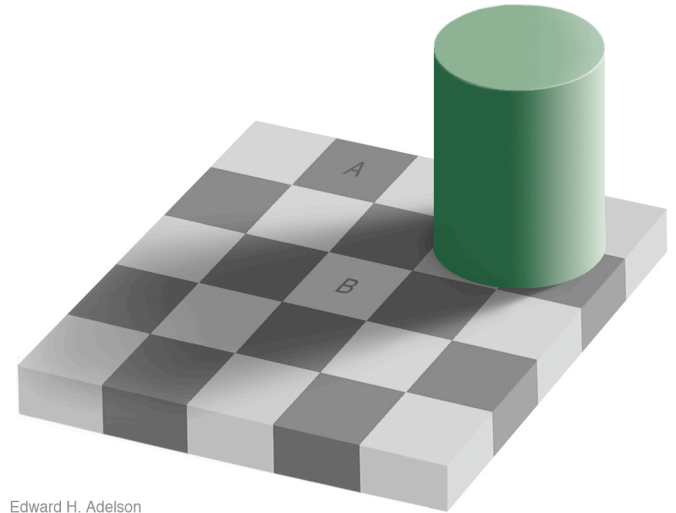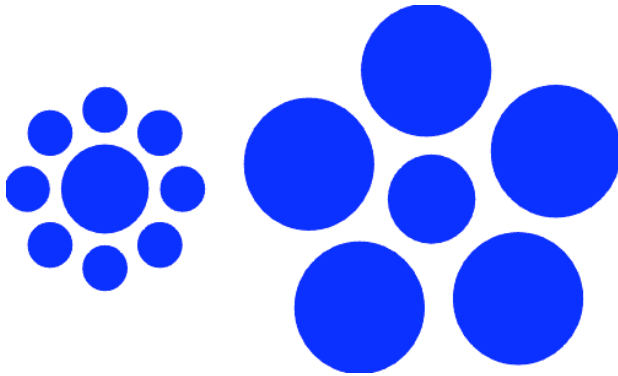
   →   r<u>e</u>ad

12
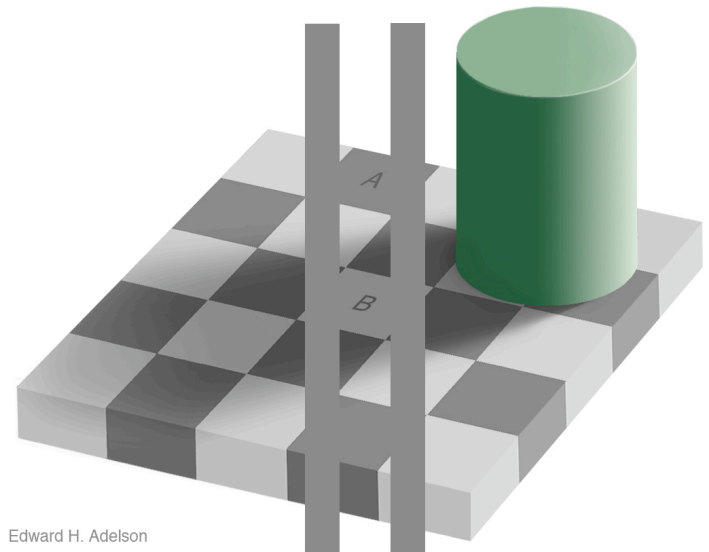A 13 C
14

festival

graphics

# … but can also deceive!



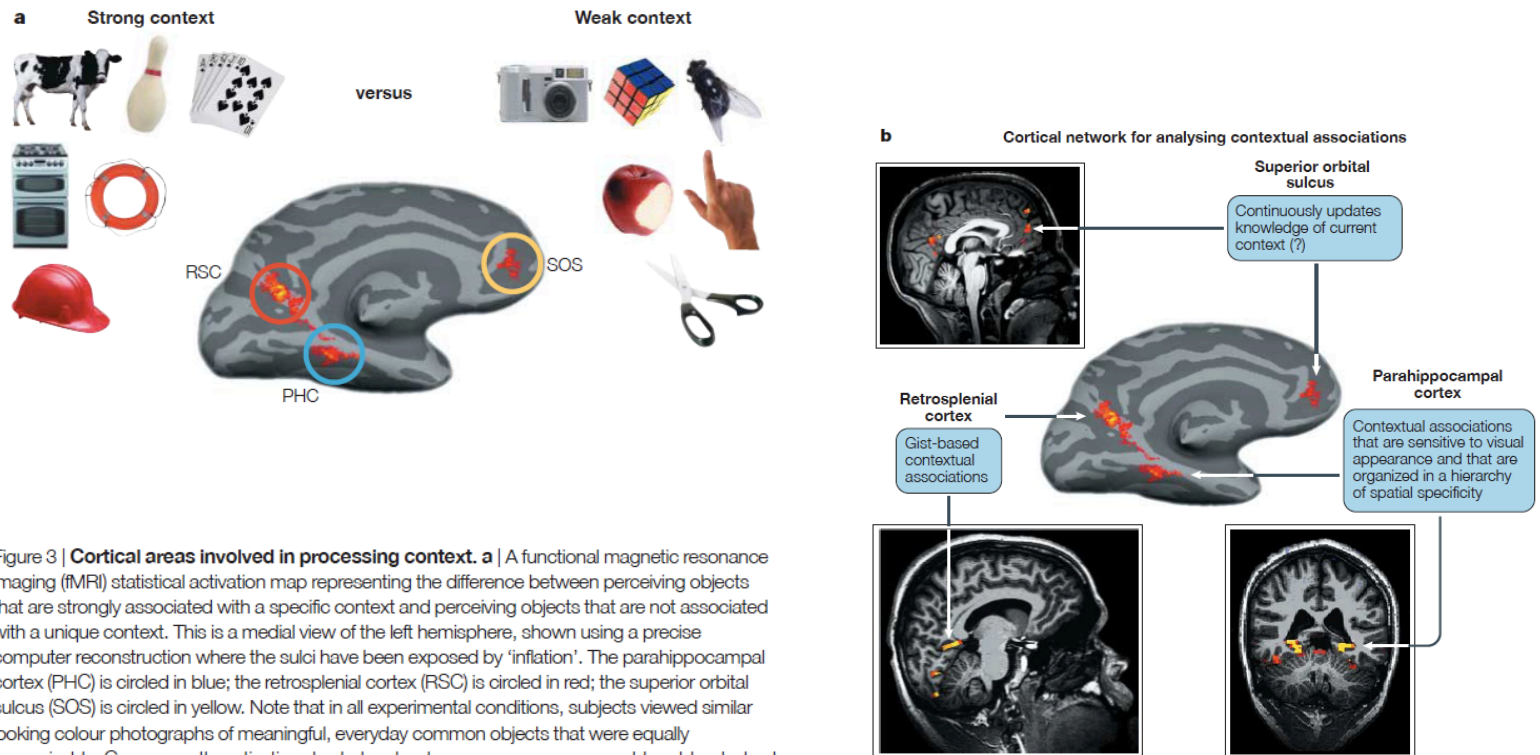Edward H. Adelson



Edward H. Adelson

# Context and the brain



Figure 3 | **Cortical areas involved in processing context. a |** A functional magnetic resonance imaging (fMRI) statistical activation map representing the difference between perceiving objects that are strongly associated with a specific context and perceiving objects that are not associated with a unique context. This is a medial view of the left hemisphere, shown using a precise computer reconstruction where the sulci have been exposed by 'inflation'. The parahippocampal cortex (PHC) is circled in blue; the retrosplenial cortex (RSC) is circled in red; the superior orbital sulcus (SOS) is circled in yellow. Note that in all experimental conditions, subjects viewed similar looking colour photographs of meaningful, everyday common objects that were equally recognizable. Consequently, activation due to low-level processes was presumably subtracted out, and the differential activation map shown here represents only processes that are related to the level of contextual association. **b |** The cortical network for contextual associations among visual objects, suggested on the basis of existing evidence. Other types of context might involve additional regions (for example, hippocampus for navigation[125] and Broca's area for language-related context). Modified, with permission, from REF. 12 © (2003) Elsevier Science.

From: M. Bar, "Visual objects in context", *Nature Reviews Neuroscience*, August 2004.

# The consistent labeling problem

A **labeling problem** involves:

✓ A set of $n$ **objects** $B = \{b_1, \ldots, b_n\}$

✓ A set of $m$ **labels** $\Lambda = \{1, \ldots, m\}$

The goal is to label each object of B with a label of $\Lambda$.

To this end, two sources of information are exploited:

✓ Local measurements, which capture the salient features of each object viewed in isolation

✓ Contextual information, expressed in terms of a real-valued $n^2 \times m^2$ matrix of **compatibility coefficients** $R = \{r_{ij}(\lambda, \mu)\}$.

The coefficient $r_{ij}(\lambda, \mu)$ measures the strenght of compatibility between the two hypotheses: "$b_i$ is labeled $\lambda$" and "$b_j$ is labeled $\mu$".

# Relaxation labeling processes

In a now classic 1976 paper, Rosenfeld, Hummel, and Zucker introduced the following update rule (assuming a non-negative compatibility matrix):

$$p_i^{(t+1)}(\lambda) = \frac{p_i^{(t)}(\lambda)q_i^{(t)}(\lambda)}{\sum_\mu p_i^{(t)}(\mu)q_i^{(t)}(\mu)}$$

where

$$q_i^{(t)}(\lambda) = \sum_j \sum_\mu r_{ij}(\lambda,\mu)p_i^{(t)}(\mu)$$

quantifies the support that context gives at time $t$ to the hypothesis "$b_i$ is labeled with label $\lambda$".

See (Pelillo, 1997) for a rigorous derivation of this rule in the context of a formal theory of consistency.

# Applications

Since their introduction in the mid-1970's relaxation labeling algorithms have found applications in virtually all problems in computer vision and pattern recognition:

- ✓ Edge and curve detection and enhancement
- ✓ Region-based segmentation
- ✓ Stereo matching
- ✓ Shape and object recognition
- ✓ Grouping and perceptual organization
- ✓ Graph matching
- ✓ Handwriting interpretation
- ✓ …

Further, intriguing similarities exist between relaxation labeling processes and certain mechanisms in the early stages of biological visual systems (see Zucker, Dobbins and Iverson, 1989)

# Hummel and Zucker's consistency

In 1983, Hummel and Zucker developed an elegant theory of consistency in labeling problem.
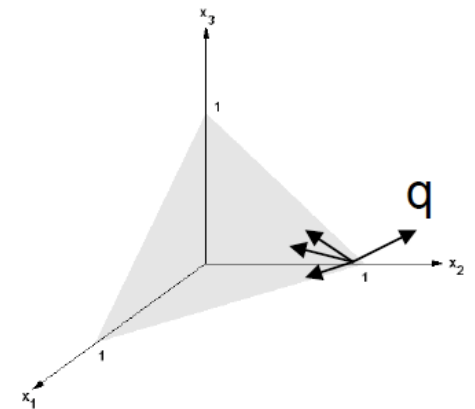
By analogy with the unambiguous case, which is easily understood, they define a weighted labeling assignment $p$ **consistent** if:

$$\sum_{\lambda} p_i(\lambda) q_i(\lambda) \geq \sum_{\lambda} v_i(\lambda) q_i(\lambda) \qquad i = 1 \ldots n$$

for all labeling assignments $v$.

**Geometrical interpretation.**

The support vector $q$ points away from all tangent vectors at $p$.



Generalization of classical (Boolean) constraint satisfaction problems!

# Relaxation labeling as a game of strategy

As observed by Miller and Zucker (1991) the consistent labeling problem is equivalent to a non-cooperative game.

Indeed, in such formulation we have:

- ✓ Objects = players
- ✓ Labels = pure strategies
- ✓ Weighted labeling assignments = mixed strategies
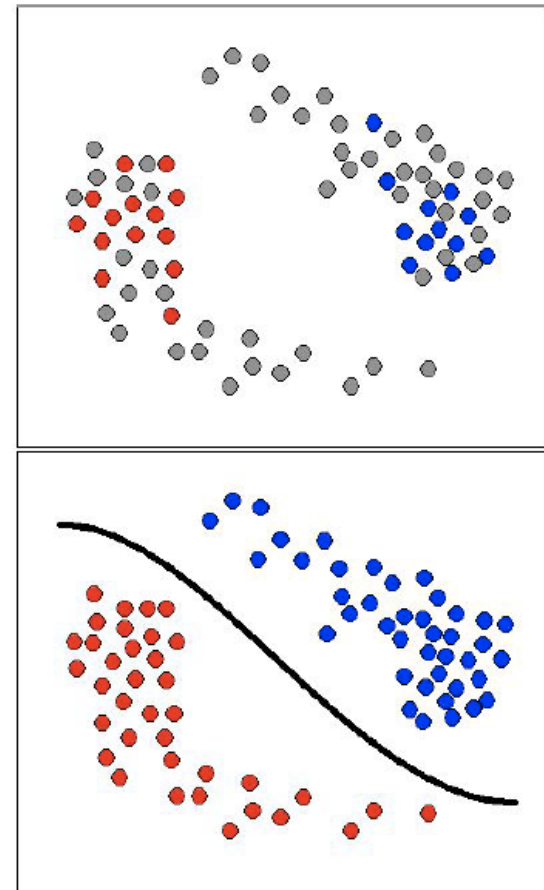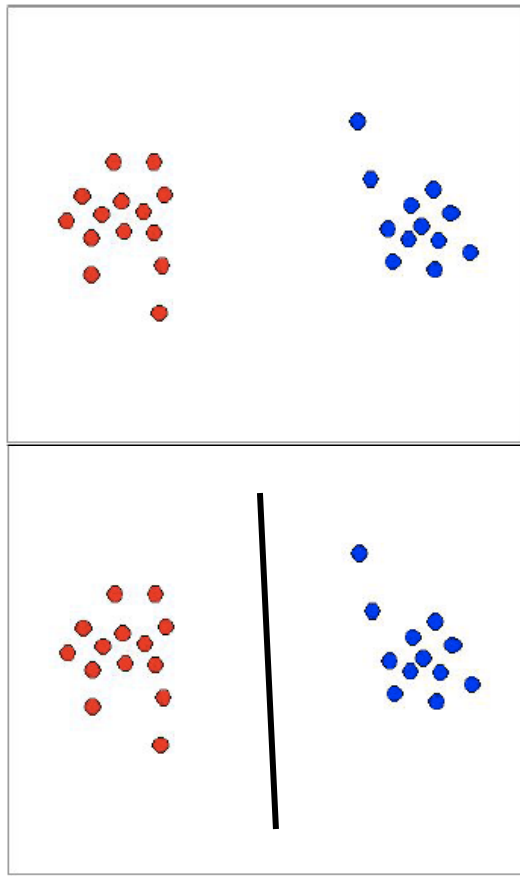- ✓ Compatibility coefficients = payoffs

and:

- ✓ **Consistent labeling = Nash equilibrium**

Further, the Rosenfeld-Hummel-Zucker update rule corresponds to discrete-time multi-population replicator dynamics.

# Application to semi-supervised learning



Adapted from: O. Duchene, J.-Y. Audibert, R. Keriven, J. Ponce, and F. Ségonne. Segmentation by transduction. *CVPR 2008*.

# Graph transduction

Given a set of data points grouped into:

- ✓ labeled data: $\{(\mathbf{x}_1, y_1), \ldots, \mathbf{x}_\ell, y_\ell)\}$
- ✓ unlabeled data: $\{\mathbf{x}_{\ell+1}, \ldots, \mathbf{x}_n\}$ $\qquad \ell \ll n$

Express data as a graph *G=(V,E)*

- ✓ *V* : nodes representing labeled and unlabeled points
- ✓ *E* : pairwise edges between nodes weighted by the similarity between the corresponding pairs of points

**Goal:** Propagate the information available at the labeled nodes to unlabeled ones in a "consistent" way.

**Cluster assumption:**

- ✓ The data form distinct clusters
- ✓ Two points in the same cluster are expected to be in the same class

# A special case

A simple case of graph transduction in which the graph $G$ is an unweighted undirected graph:

- ✓ An edge denotes perfect similarity between points
- ✓ The adjacency matrix of $G$ is a 0/1 matrix
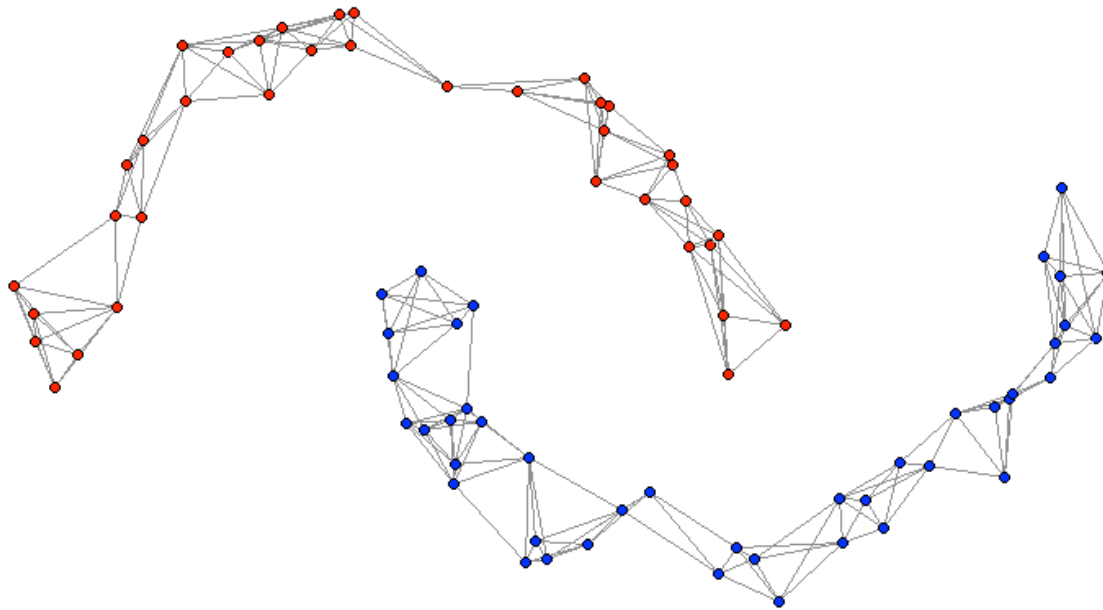


**The cluster assumption:** Each node in a connected component of the graph should have the same class label. A constraint satisfaction problem!

# The graph transduction game

Given a weighted graph G = (V, E, w), the graph trasduction game is as follow:

- ✓ Nodes = players
- ✓ Labels = pure strategies
- ✓ Weighted labeling assignments = mixed strategies
- ✓ Compatibility coefficients = payoffs

The transduction game is in fact played among the unlabeled players to choose their memberships.

- ✓ Consistent labeling = Nash equilibrium

Can be solved used standard relaxation labeling / replicator dynamics.

**Applications:** NLP (see next part), interactive image segmentation, content-based image retrieval, people tracking and re-identification, etc.

# In short …

Graph transduction can be formulated as a consistent labeling problem.

The proposed framework can cope with **symmetric, negative and asymmetric similarities** (none of the existing techniques is able to deal with all three types of similarities).

Experimental results on standard datasets show that our approach is not only more general but also competitive with standard approaches.

A. Erdem and M. Pelillo. Graph transduction as a noncooperative game. *Neural Computation* 24(3) (March 2012).

# Transductive Label Augmentation for Improved Deep Learning

A major impediment in the application of deep learning to real-world problems is the scarcity of labeled data.

Typical augmentation techniques (eg AlexNet) work well in applications such as image classification, where it is simple to design suitable transformation operators.
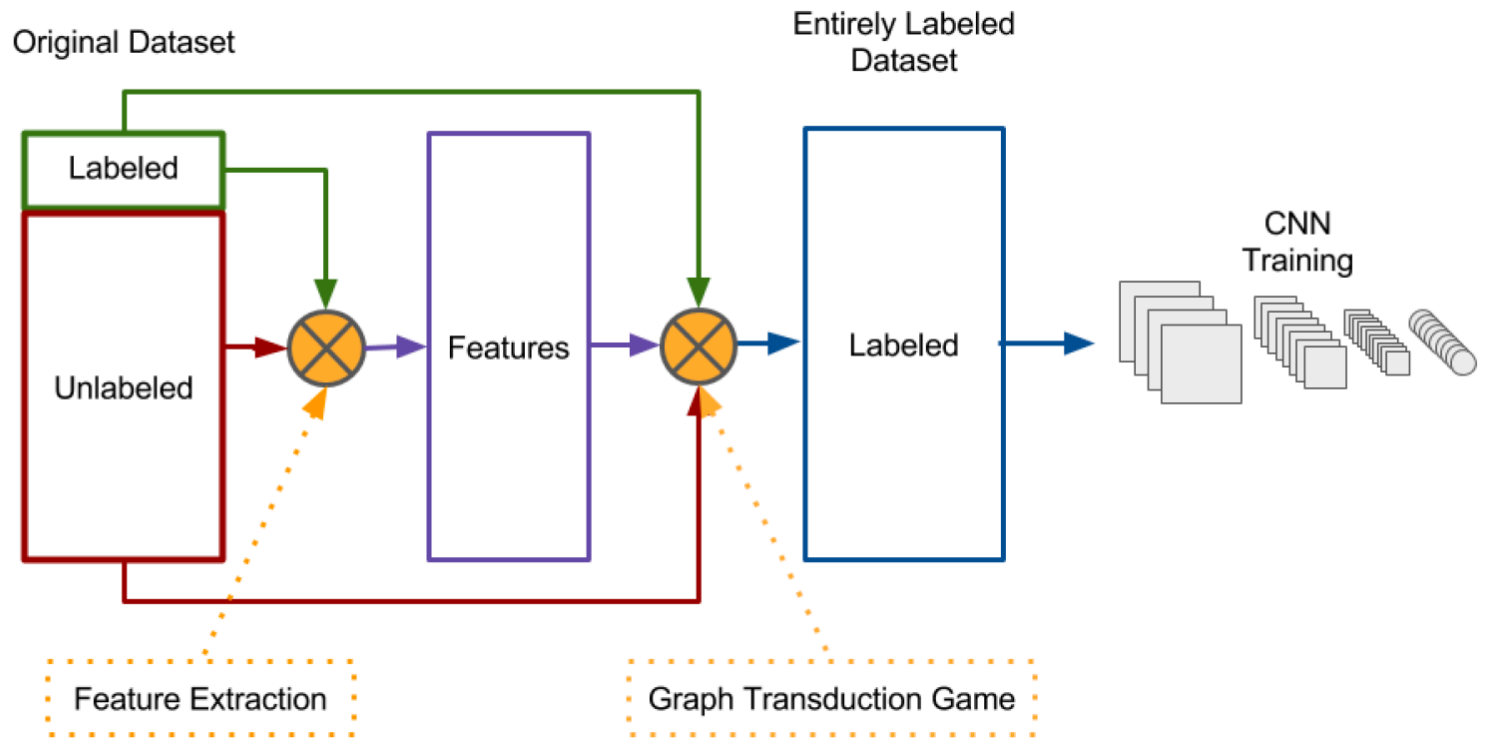
Not obvious how to apply it in more structured scenarios.

However, in virtually all application domains it is easy to obtain unlabeled data. We propose a **label augmentation** approach.

I Elezi, A. Torcinovich, S. Vascon, and M. Pelillo. Transductive label augmentation for improved deep network learning. In: *ICPR 2018*.
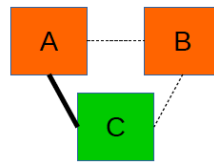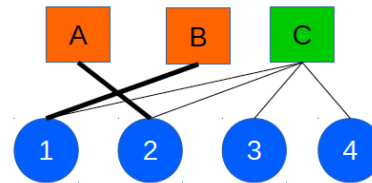
# The idea

# A toy example



Similarity Graph

Label association (t = 0)

| W | A | B | C |
|---|---|---|---|
| A | 0 | 0 | 0.9 |
| B | 0 | 0 | 0.2 |
| C | 0.9 | 0.2 | 0 |

| X(0) | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| A | 0 | 1 | 0 | 0 |
| B | 1 | 0 | 0 | 0 |
| C | 0.25 | 0.25 | 0.25 | 0.25 |

1st iteration
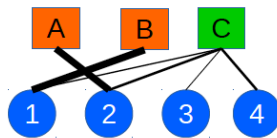
| X(1) | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| A | 0 | 1 | 0 | 0 |
| B | 1 | 0 | 0 | 0 |
| C | 0.21 | 0.47 | 0.16 | 0.16 |

2nd iteration
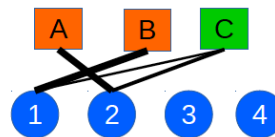
| X(2) | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| A | 0 | 1 | 0 | 0 |
| B | 1 | 0 | 0 | 0 |
| C | 0.15 | 0.85 | 0 | 0 |

3rd iteration
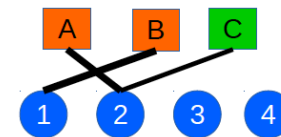
| X(3) | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| A | 0 | 1 | 0 | 0 |
| B | 1 | 0 | 0 | 0 |
| C | 0 | 1 | 0 | 0 |

# Results

| accuracy 2% labeled | caltech | | indoors | | scenenet | |
|---|---|---|---|---|---|---|
| | RN18 | DN121 | RN18 | DN121 | RN18 | DN121 |
| GTG + CNN | **0.532** | **0.620** | **0.486** | **0.538** | **0.430** | **0.495** |
| SVM + CNN | 0.473 | 0.539 | 0.434 | 0.468 | 0.370 | 0.417 |
| CNN | 0.266 | 0.235 | 0.341 | 0.323 | 0.205 | 0.178 |

| F score 2% labeled | caltech | | indoors | | scenenet | |
|---|---|---|---|---|---|---|
| | RN18 | DN121 | RN18 | DN121 | RN18 | DN121 |
| GTG + CNN | **0.468** | **0.559** | **0.357** | **0.396** | **0.399** | **0.457** |
| SVM + CNN | 0.388 | 0.455 | 0.319 | 0.327 | 0.352 | 0.377 |
| CNN | 0.181 | 0.151 | 0.187 | 0.172 | 0.191 | 0.167 |

| accuracy 5% labeled | caltech | | indoors | | scenenet | |
|---|---|---|---|---|---|---|
| | RN18 | DN121 | RN18 | DN121 | RN18 | DN121 |
| GTG + CNN | **0.625** | **0.698** | **0.568** | **0.613** | **0.563** | **0.621** |
| SVM + CNN | 0.605 | 0.675 | 0.516 | 0.580 | 0.511 | 0.601 |
| CNN | 0.457 | 0.444 | 0.456 | 0.466 | 0.408 | 0.438 |

| F score 5% labeled | caltech | | indoors | | scenenet | |
|---|---|---|---|---|---|---|
| | RN18 | DN121 | RN18 | DN121 | RN18 | DN121 |
| GTG + CNN | **0.571** | **0.653** | **0.454** | **0.508** | **0.536** | **0.608** |
| SVM + CNN | 0.542 | 0.626 | 0.426 | 0.505 | 0.501 | 0.590 |
| CNN | 0.372 | 0.358 | 0.345 | 0.306 | 0.394 | 0.419 |

| accuracy 10% labeled | caltech | | indoors | | scenenet | |
|---|---|---|---|---|---|---|
| | RN18 | DN121 | RN18 | DN121 | RN18 | DN121 |
| GTG + CNN | **0.667** | **0.727** | **0.598** | **0.645** | **0.624** | **0.686** |
| SVM + CNN | 0.658 | 0.724 | 0.576 | 0.635 | 0.622 | 0.660 |
| CNN | 0.577 | 0.598 | 0.553 | 0.567 | 0.571 | 0.584 |

| F score 10% labeled | caltech | | indoors | | scenenet | |
|---|---|---|---|---|---|---|
| | RN18 | DN121 | RN18 | DN121 | RN18 | DN121 |
| GTG + CNN | **0.622** | **0.694** | 0.509 | 0.574 | 0.609 | **0.700** |
| SVM + CNN | 0.612 | 0.686 | **0.515** | **0.579** | **0.612** | 0.650 |
| CNN | 0.519 | 0.533 | 0.478 | 0.471 | 0.565 | 0.570 |

# Results



(a)      (b)      (c)

(d)      (e)      (f)
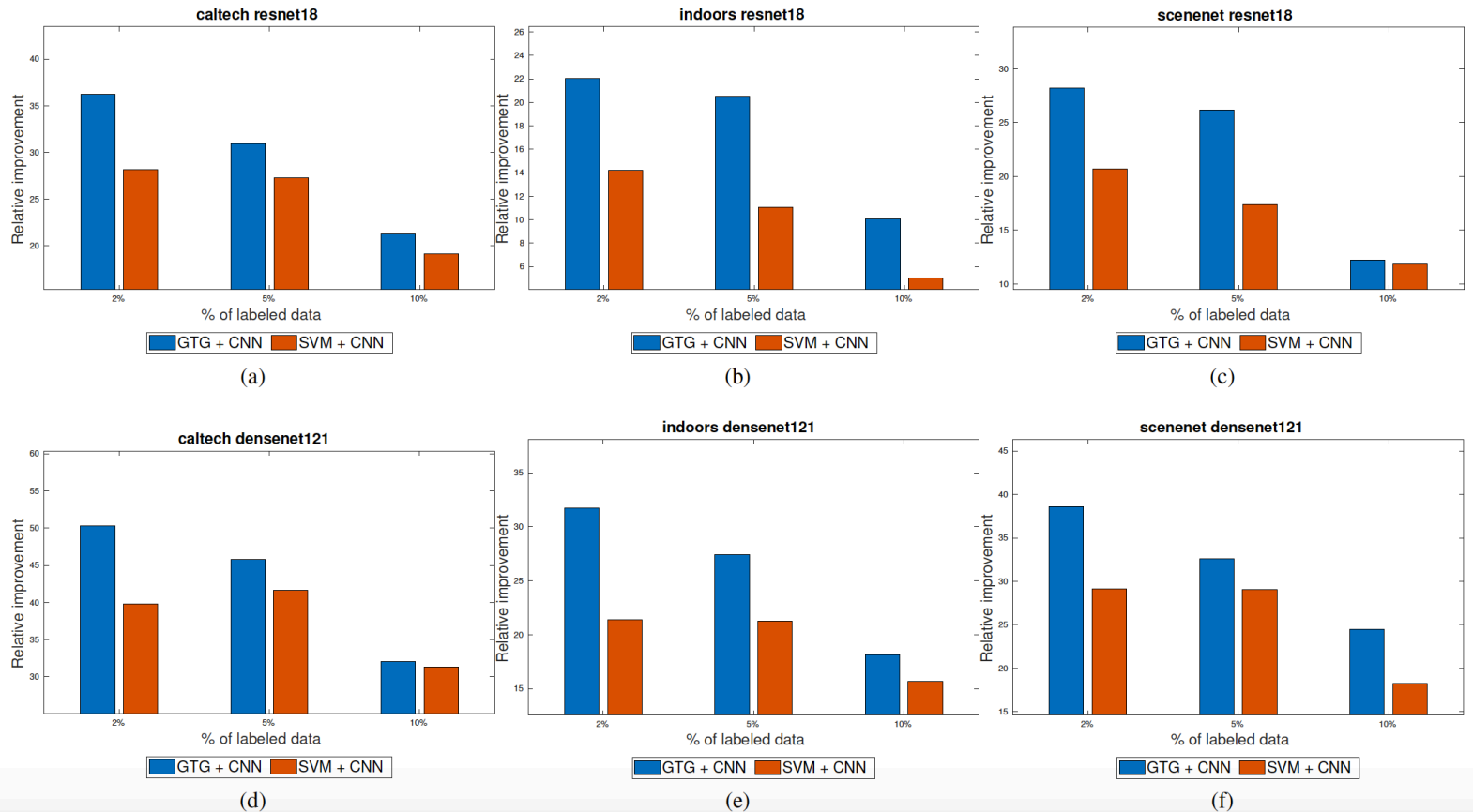
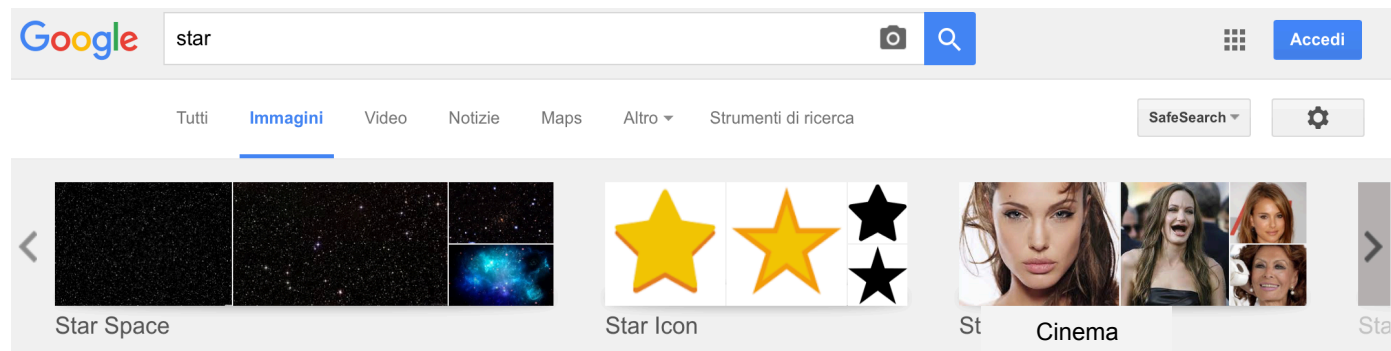Application to ancient coin classification
(Aslan, Vascon and Pelillo, 2018)

# Word sense disambiguation

WSD is the task to identify the intended meaning of a word based on the context in which it appears.

- One of the stars in the star cluster Pleiades [...]
- One of the stars in the last David Lynch film [...]



It has been studied since the beginning of NLP and also today is a central topic of this discipline.

Used in applications like text understanding, machine translation, opinion mining, sentiment analysis and information extraction.

# WSD games

The WSD problem can be formulated in game-theoretic terms modeling:

- the players of the games as the words to be disambiguated.
- the strategies of the games as the senses of each word.
- the payoff matrices of each game as a sense similarity function.
- the interactions among the players as a weighted graph.

Nash equilibria correspond to consistent word-sense assignments!

- Word-level similarities: proportional to strength of co-occurrence between words

- Sense-level similarities: computed using WordNet / BabelNet ontologies

R. Tripodi and M. Pelillo. A game-theoretic approach to word sense disambiguation. *Computational Linguistics* 43(1) (January 2017).

# An example

*There is a financial institution near the river bank.*
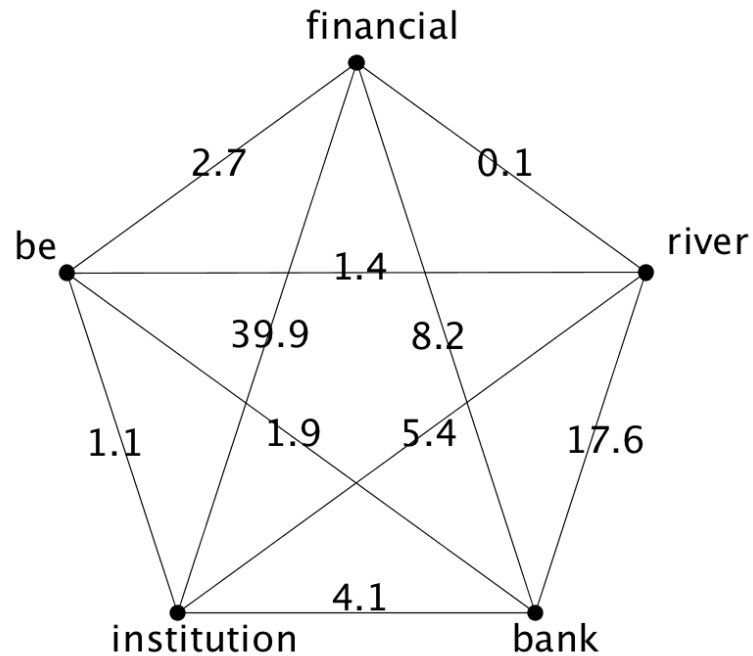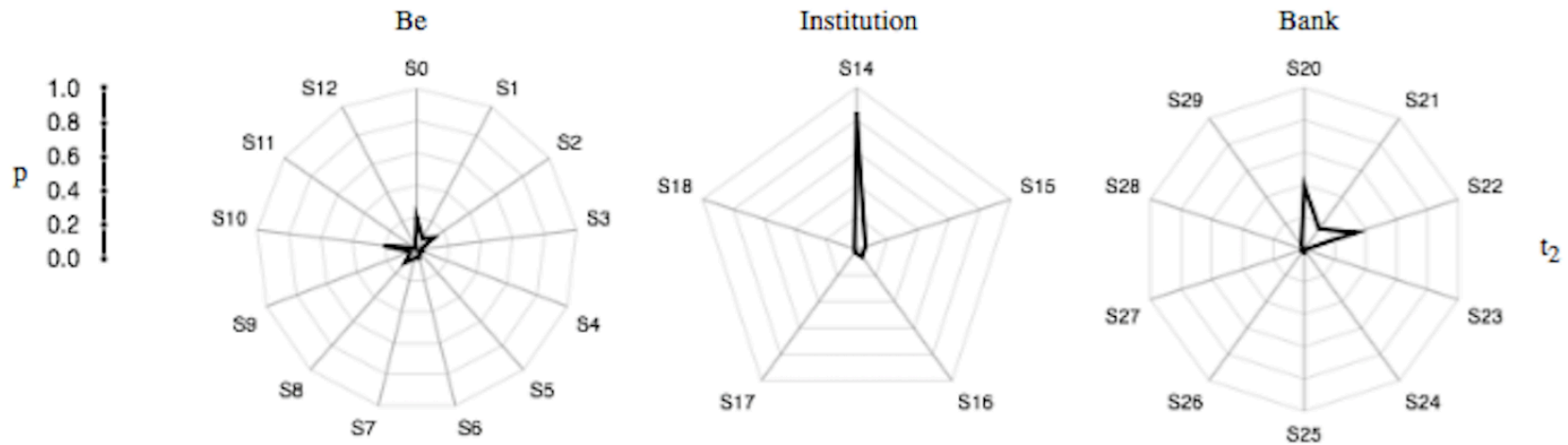
# WSD game dynamics (time = 1)

*There is a financial institution near the river bank.*
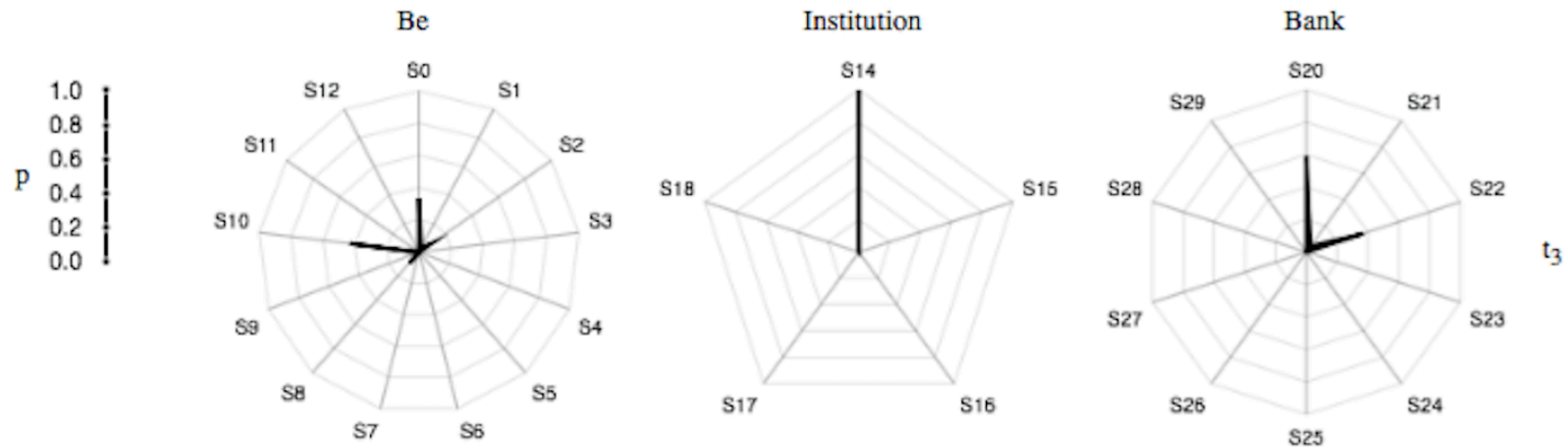
# WSD games dynamics (time = 2)

*There is a financial institution near the river bank.*
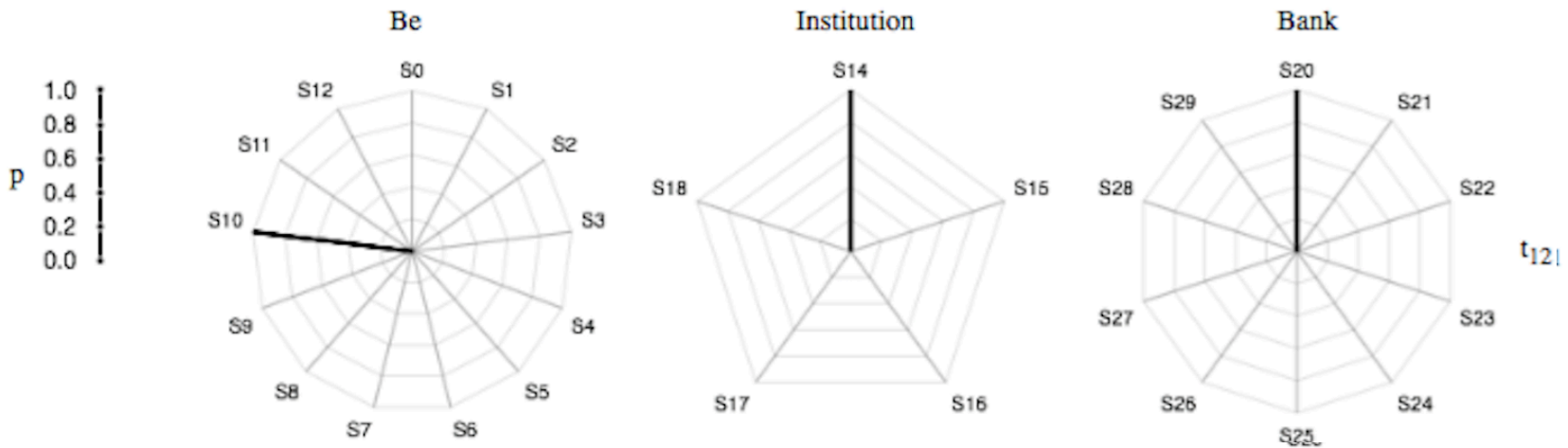
# WSD game dynamics (time = 3)

*There is a financial institution near the river bank.*

# WSD games dynamics (time = 12)

*There is a financial institution near the river bank.*

# Experimental setup

| | name | task | KB | texts | words |
|---|---|---|---|---|---|
| S7 | SemEval 2007 | fine grained | WN | 3 | 444 |
| S7CG | SemEval 2007 | coarse grained | WN | 5 | 2269 |
| S3 | Senseval 3 | fine grained | WN | 3 | 2041 |
| S2 | Senseval 2 | fine grained | WN | 3 | 2473 |
| S13 | SemEval 2013 | wsd & entity disambiguation | BN | 13 | 1931 |
| KORE | KORE50 | entity disambiguation | BN | 50 | 146 |

Evaluation measure:

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision + recall} \cdot 100$$

# Results

|  |  | S7CG | S7CG (N) | S7 | S3 | S2 |
|---|---|---|---|---|---|---|
| *unsup.* | *Nav10* | — | — | 43.1 | 52.9 | — |
|  | *PPR$_{w2w}$* | 80.1 | 83.6 | 41.7 | 57.9 | 59.7 |
|  | *WSD$_{games}$* | **80.4*** | **85.5** | **43.3** | **59.1** | **61.2** |
| *semi sup.* | *IRST-DDD-00* | — | — | — | 58.3 | — |
|  | *MFS* | 76.3 | 77.4 | 54.7 | 62.8 | 65.6* |
|  | *MRF-LP* | — | — | 50.6* | 58.6 | 60.5 |
|  | *Nav05* | **83.2** | 84.1 | — | 60.4 | — |
|  | *PPR$_{w2w}$* | 81.4 | 82.1 | 48.6 | 63.0 | 62.6 |
|  | *WSD$_{games}$* | 82.8 | **85.4** | **56.5** | **64.7*** | **66.0** |
| *sup.* | *Best* | 82.5 | 82.3* | **59.1** | 65.2 | **68.6** |
|  | *Zhong10* | **82.6** | — | 58.3 | **67.6** | 68.2 |

# The "protein function predition" game

**Motivation:** network-based methods for the automatic prediction of protein functions can greatly benefit from exploiting *both* the similarity between proteins and the similarity between functional classes.

**Hume's principle:** *similar* proteins should have *similar* functionalities

We envisage a (non-cooperative) game where

- Players = proteins,
- Strategies = functional classes
- Payoff function = combination of protein- and function-level similarities

Nash equilibria provide consistent functional labelings of proteins.

S. Vascon, M. Frasca, R. Tripodi, G. Valentini and M. Pelillo. Protein function predition as a graph-transduction game. *Pattern Recognition Letters* (in press).

# Protein similarity

The similarity between proteins has been calculated integrating different data types.

| Database | Type of data |
|---|---|
| PRINTS | Motif fingerprints |
| PROSITE | Protein domains and families |
| Pfam | Protein domain |
| SMART | Simple Modular Architecture Research Tool (database annotations) |
| InterPro | Integrated resource of protein families, domains and functional sites |
| Protein Superfamilies | Structural and functional annotations |
| EggNOG | Evolutionary genealogy of genes: Non-supervised Orthologous Groups |
| Swissprot | Manually curated keywords describing the function of the proteins at different degrees of abstraction |

The final similarity matrix for each organism is obtained integrating the 8 sources via an unweighted mean.

# Funtion similarity

The similarities between the classes functionalities have been computed using the Gene Ontology (GO)
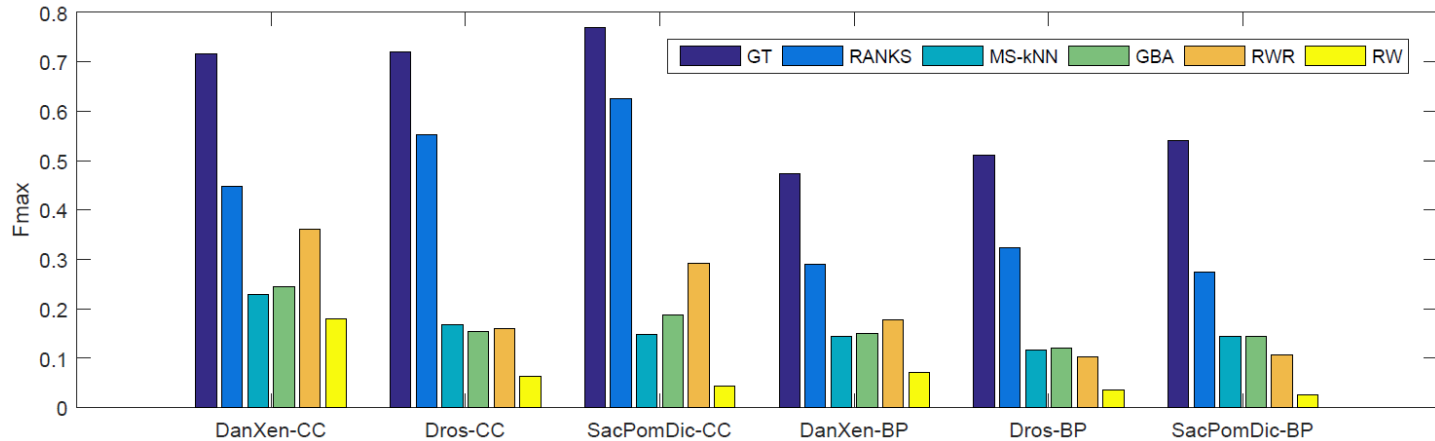
The similarity between the GO terms (classes) for each integrated network and each ontology are computed using:

- semantic similarities measures (Resnick or Lin)
- a Jaccard similarity measure between the annotations of each GO term.
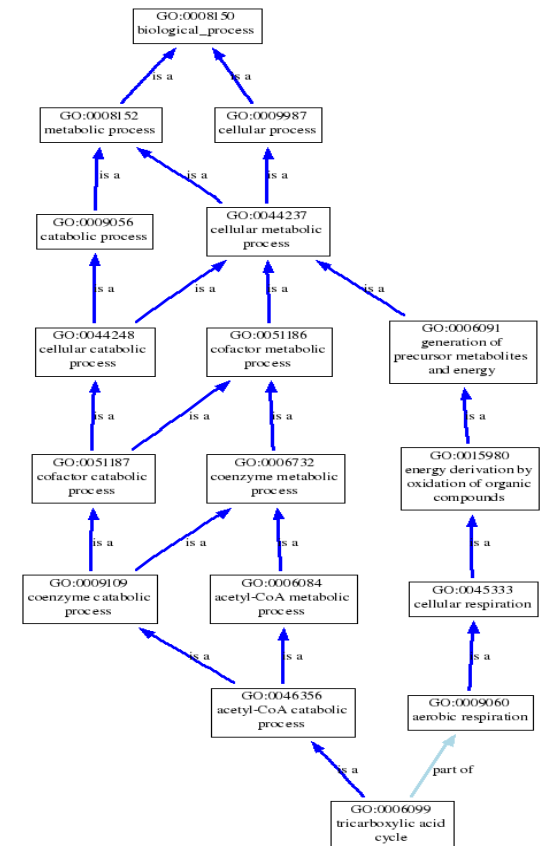
# Preliminary results



**Networks:** DanXen (includes zebrafish and frog proteins), Dros (fruit fly), SacPomDic (includes the proteins of three unicellular eukaryotes).

CC = cellular component / BP = biological processs

Number of nodes (proteins): from 3195 (Dros) to 15836 (SacPomDic)

CC terms (classes): from 184 to 919
BP terms (classes): from 2281 to 5037

**Competitors**

- Random Walk (RW)
- Random Walk with Restart (RWR)
- Funckenstein (GBA)
- Multi Source-kNN method (MS-kNN)
- RANKS

More extensive and up-to-date results in the PRL paper!

# References

R. A. Hummel and S. W. Zucker. On the foundations of relaxation labeling processes. *IEEE Trans. Pattern Anal. Machine Intell.* (1983)

M. Pelillo. The dynamics of nonlinear relaxation labeling processes. *J. Math. Imaging and Vision* (1997)

M. Pelillo and M. Refice. Learning compatibility coefficients for relaxation labeling processes. *IEEE Trans. Pattern Anal. Machine Intell.* (1994)

D. A. Miller and S. W. Zucker. Copositive-plus Lemke algorithm solves polymatrix games. *Operation Research Letters* (1991)

A. Erdem and M. Pelillo. Graph transduction as a non-cooperative game. *Neural Computation* (2012)

R. Tripodi and M. Pelillo. A game-theoretic approach to word-sense disambiguation. *Computational Linguistics* (2017)

S. Vascon, M. Frasca, R. Tripodi, G. Valentini, M. Pelillo. Protein function prediction as a graph-transduction game. *Pattern Recognition Letters* (2018)

I Elezi, A. Torcinovich, S. Vascon, and M. Pelillo. Transductive label augmentation for improved deep network learning. In: *ICPR 2018*.

S. Aslan, S. Vascon, and M. Pelillo. Ancient coin classification using graph transduction games. In: *MetroArchaeo 2018*.