Conceptual Clustering: Concept Formation, Drift and Novelty Detection

Nicola Fanizzi   Claudia d'Amato   Floriana Esposito

Department of Computer Science
University of Bari

ESWC 2008 ◇ Tenerife, June 4, 2008

Contents

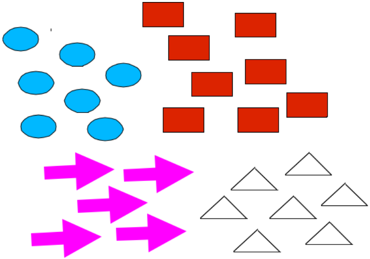
Introduction & Motivation

- Ontologies evolve over the time.
  - *New instances are asserted*
  - New concepts are defined
- **Concept Drift**
  - the change of a known concept w.r.t. the evidence provided by new annotated individuals that may be made available over time
- **Novelty Detection**
  - isolated cluster in the search space that requires to be defined through new emerging concepts to be added to the KB
- *IDEA* : **to use Conceptual clustering methods for automatically discover them**

Basics on Clustering Methods

Clustering methods: unsupervised inductive learning methods that organize a collection of unlabeled resources into meaningful clusters such that

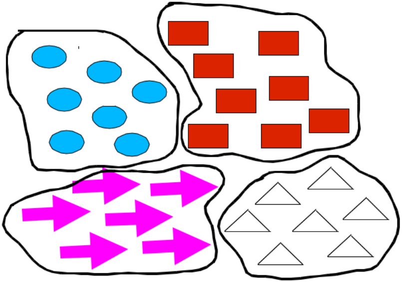
- intra-cluster *similarity* is high
- inter-cluster *similarity* is low



Basics on Clustering Methods

Clustering methods: unsupervised inductive learning methods that organize a collection of unlabeled resources into meaningful clusters such that

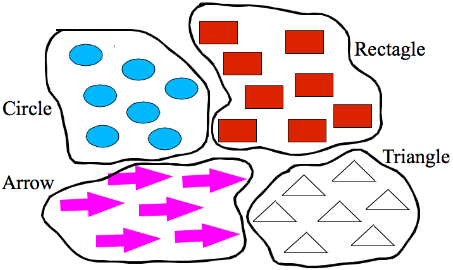
- intra-cluster *similarity* is high
- inter-cluster *similarity* is low



Basics on Clustering Methods

Clustering methods: unsupervised inductive learning methods that organize a collection of unlabeled resources into meaningful clusters such that

- intra-cluster *similarity* is high
- inter-cluster *similarity* is low



Conceptual Clustering: Related Works

- **Few** algorithms for Conceptual Clustering (CC) with multi-relational representations [Stepp & Michalski, 86]
- **Fewer** dealing with the SW standard representations and their semantics
  - KLUSTER [Kietz & Morik, 94]
  - CSKA [Fanizzi et al., 04]
    - Produce a *flat output*
    - *Suffer from noise* in the data
- **Proposal** of a new divisional hierarchical CC algorithm that
  - is **similarity-based** $\Rightarrow$ *noise tolerant*
  - produces a *hierarchy of clusters*

Reference Representation

- OWL representation founded in Description Logics (DL):
- Knowledge base: $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$
  - TBox $\mathcal{T}$: a set of DL concept definitions
  - ABox $\mathcal{A}$: assertions (facts) about the world state
  - $\mathsf{Ind}(\mathcal{A})$: set of Individuals (resources) in the ABox
- Inference service of interest from the KBMS:
  - *instance-checking*: decision procedure that assess if an individual is instance of a certain concept or not
    - Sometimes a simple lookup may be sufficient

Semi-Distance Measure: Main Idea

- **IDEA**: *on a semantic level, similar individuals should behave similarly w.r.t. the same concepts*
- Following HDD **[Sebag 1997]**: individuals can be compared on the grounds of their behavior w.r.t. a given set of hypotheses $\mathsf{F} = \{F_1, F_2, \ldots, F_m\}$, that is a collection of (primitive or defined) concept descriptions
  - F stands as a group of *discriminating features* expressed in the considered language
- As such, the new measure *totally depends on semantic* aspects of the individuals in the KB

Semantic Semi-Dinstance Measure: Definition

[Fanizzi et al. @ DL 2007] Let $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a KB and let $\mathsf{Ind}(\mathcal{A})$ be the set of the individuals in $\mathcal{A}$. Given sets of concept descriptions $\mathsf{F} = \{F_1, F_2, \ldots, F_m\}$ in $\mathcal{T}$, a *family of semi-distance functions* $d_p^{\mathsf{F}} : \mathsf{Ind}(\mathcal{A}) \times \mathsf{Ind}(\mathcal{A}) \mapsto \mathbb{R}$ is defined as follows:

$$\forall a, b \in \mathsf{Ind}(\mathcal{A}) \quad d_p^{\mathsf{F}}(a, b) := \frac{1}{m} \left[\sum_{i=1}^{m} \mid \pi_i(a) - \pi_i(b) \mid^p \right]^{1/p}$$

where $p > 0$ and $\forall i \in \{1, \ldots, m\}$ the *projection function* π_i is defined by:

$$\forall a \in \mathsf{Ind}(\mathcal{A}) \quad \pi_i(a) = \begin{cases} 1 & F_i(a) \in \mathcal{A} \quad (\mathcal{K} \models F_i(a)) \\ 0 & \neg F_i(a) \in \mathcal{A} \quad (\mathcal{K} \models \neg F_i(a)) \\ \frac{1}{2} & \mathit{otherwise} \end{cases}$$

Semi-Distance Measure: Discussion

- *More similar* the considered *individuals are*, more similar the project function values are $\Rightarrow d_p^{\mathsf{F}} \simeq 0$
- *More different* the considered *individuals are*, more different the projection values are $\Rightarrow$ the value of d_p^{F} will increase
- The measure does not depend on any specific constructor of the language $\Rightarrow$ *Language Independent Measure*
- The measure complexity mainly depends from the complexity of the *Instance Checking* operator for the chosen DL
  - $Compl(d_p^{\mathsf{F}}) = |\mathsf{F}| \cdot 2 \cdot Compl(\mathrm{IChk})$
- **Optimal discriminating feature set could be learned**

Clustering Algorithm: Characteristics

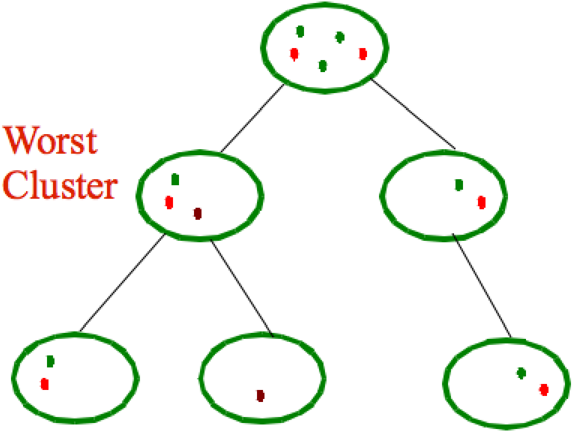
- *Hierarchical* algorithm $\Rightarrow$ returns a *hierarchy of clusters*
- Inspired to the K-Means algorithm
    - Defined for feature vectors representation where features are only numerical and the notion of the cluster *centroids* (weighted average of points in a cluster) is used for partition
- Exploits the notion of **medoid** (drawn from the PAM algorithm)
    - **central element in a group of instances**

$$m = \mathrm{medoid}(C) = \operatorname*{argmin}_{a \in C} \sum_{j=1}^{n} d(a, a_j)$$

Running the Clustering Algorithm

- *Level-wise* (number of level given in input, it is the number of clusters that we want to obtain): find the **worst cluster** on that level that has to be slip
    - *worst cluster* $\Leftrightarrow$ having the *least average inner similarity* (**cohesiveness**)
    - **select** the two **most dissimilar element** in the cluster *as medoid*
- split the cluster iterating (till convergence)
    - **distribute individuals** to either partition on the grounds of their similarity w.r.t. the medoids
    - given this bipartition, **compute the new medoids** *for either cluster*
    - **STOP when** the two generated medoids are equal to the previous ones (stable configuration) **or when** the maximum number of iteration is reached

Clustering Algorithm: Main Idea



Conceptual Clustering Step

For DLs that allow for (approximations of) the msc and lcs, (e.g. $\mathcal{ALC}$ or $\mathcal{ALE}$):

- given a cluster $node_j$,
  - $\forall a_i \in \text{node}_j$ compute $M_i := msc(a_i)$ w.r.t. the ABox $\mathcal{A}$
  - let $MSCs_j := \{M_i | a_i \in \text{node}_j\}$
- node_j *intensional description* $lcs(MSCs_j)$

Alternatively a *Supervised Learning phase* can be used

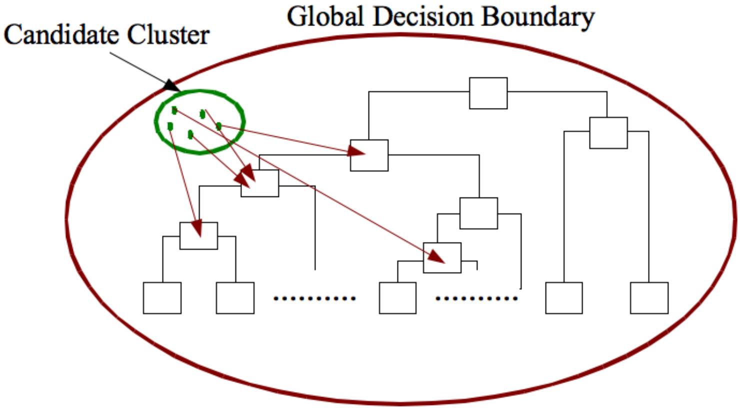
- Learn a definition for $node_j$ whose individuals represent the positive examples while the individuals in the other clusters at the same level are the negative example
- More complex algorithms for concepts learning in some DLs may be employed ([Esposito,04] [Lehmann,06])

Automated Concept Drift and Novelty Detection

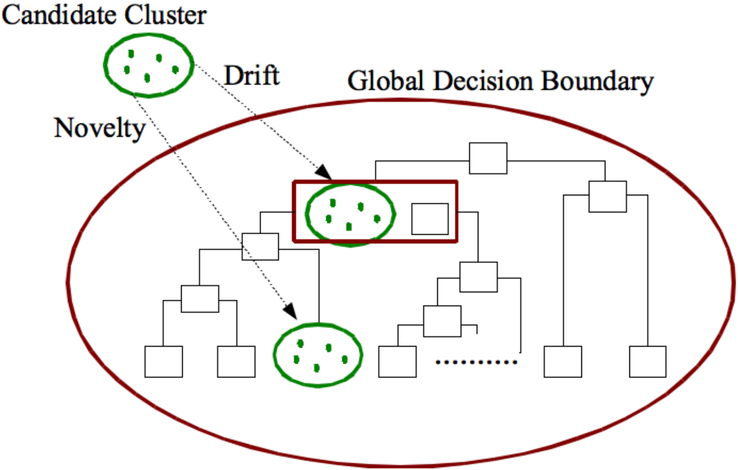
If *new annotated individuals are made available* they have to be integrated in the clustering model

1. Each individual is assigned to the closest cluster (measuring the distance w.r.t. the cluster medoids)
2. The entire clustering model is recomputed
3. The new instances are considered to be a *candidate* cluster
   - An *evaluation* of it is performed in order to assess its nature

Evaluating the Candidate Cluster: Main Idea 1/2



Evaluating the Candidate Cluster: Main Idea 2/2



Evaluating the Candidate Cluster

- Given the initial clustering model, a *global boundary* is computed for it
  - $\forall C_i \in \text{Model}$, *decision boundary cluster* $= max_{a_j \in C_i} d(a_j, m_i)$ (or the average)
  - The average of the decision boundary clusters w.r.t. all clusters represent the *decision boundary model or global boundary* $d_{overall}$
- The decision boundary for the candidate cluster CandCluster is computed $d_{candidate}$
- if $d_{candidate} \leq d_{ovevrall}$ then CandCluster is a *normal* cluster
  - *integrate* :
    $\forall a_i \in \text{CandCluster}\ a_i \rightarrow C_j\ s.t.\ d(a_i, m_j) = min_{m_j} d(a_i, m_j)$
- else CandCluster is a **Valid Candidate** for *Concept Drift* or *Novelty Detection*

Evaluating Concept Drift and Novelty Detection

- The *Global Cluster Medoid* is computed
  $\overline{m} := \mathrm{medoid}(\{m_j \mid C_j \in \mathsf{Model}\})$
- $d_{\max} := \max_{m_j \in \mathsf{Model}} d(\overline{m}, m_j)$
- if $d(\overline{m}, m_{CC}) \leq d_{\max}$ the CandCluster is a *Concept Drift*
  - CandCluster is **Merged** with the most similar cluster $C_j \in \mathsf{Model}$
- if $d(\overline{m}, m_{CC}) \geq d_{\max}$ the CandCluster is a *Novel Concept*
  - CandCluster is **added** to the model (at the level j where the most similar cluster is found)

Experimental Setting

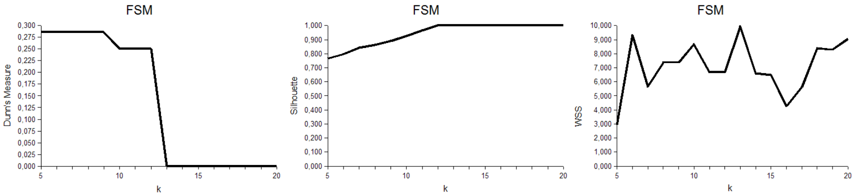
ontology	DL	#concepts	#obj. prop.	#data prop.	#individuals
FSM	$\mathcal{SOF}(D)$	20	10	7	37
S.-W.-M.	$\mathcal{ALCOF}(D)$	19	9	1	115
TRANSPORTATION	$\mathcal{ALC}$	44	7	0	250
FINANCIAL	$\mathcal{ALCIF}$	60	17	0	652
NTN	$\mathcal{SHIF}(D)$	47	27	8	676

- For each ontology, the *experiments* have been *repeated for varying numbers* k of clusters **(5 through 20)**
- For computing individual distances *all concepts* in the ontology have been used as committee of features
    - this guarantees high redundancy and thus meaningful results
- PELLET reasoner employed for computing the projections

Evaluation Methodology

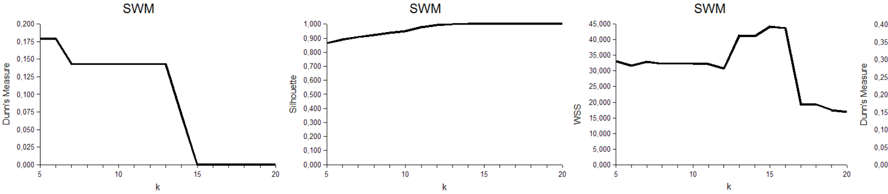
- Obtained clusters evaluated, per each value of k by the use of the standard metrics
  - **Generalized Dunn's index** $[0, +\infty[$
  - Mean Square error **WSS cohesion index** $[0, +\infty[$
    - within cluster squared sum of distances from medoid
  - **Silhouette index** $[-1, +1]$
- An overall experimentation of **16 repetitions** on a dataset took *from a few minutes to 1.5 hours* on a 2.5GhZ (512Mb RAM) Linux Machine.

Experimental Results 1/3

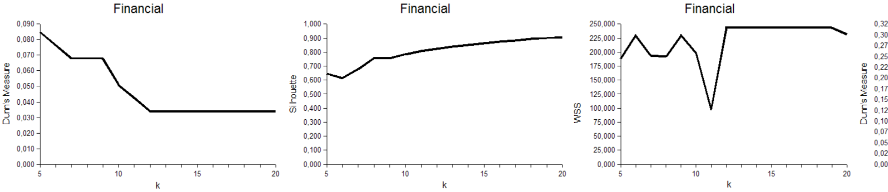


- Silhouette (most representative index)
    - Close to its max value (1)
- Dunn's + WSS:
    - knees can give a hint of optimal choice for clustering

Experimental Results 2/3



Experimental Results 3/3



Conclusions

- A hierarchical clustering algorithm for relational KBs expressed in any DL has been presented
- Based on a language independent dissimilarity measure grounded on resource semantics
  - The instance checking inference operator is exploited
- Clusters have been experimentally evaluated
  - Registered good preliminary results particularly w.r.t. Silhouette quality index

Future Works

- Grouping homogeneous individuals in the candidate cluster and evaluate each group w.r.t. the model
- Evaluating the clustering algorithm by the use of the distance optimization
- Extension to Fuzzy clustering techniques
- Conceptual Clustering Step as a Supervised learning phase with complex DL languages
- Application: Clustering Semantic WS descriptions for fast retrieval and matchmaking

The End

That's all!

Questions?