An unsupervised classification process for large datasets based on web reasoning

Rafael PEIXOTO, Thomas HASSAN, Christophe CRUZ, Aurelie BERTAUX, Nuno SILVA
thomas.hassan@u-bourgogne.fr

Laboratoire LE2I – UMR CNRS 6306 – Université de Bourgogne
Outline

Context
- Global problem
- The Semantic HMC

Specific Problem
- Proposed Solution

Implementation
- Setup
- Results

Conclusion and future work
Global Problem

Value extraction from Big Data sources
Global Problem

- Why ontologies
  - Ontologies are the most accepted way to represent semantics in the Semantic Web and a good solution for intelligent computer systems that operate close to the human concept level, bridging the gap between human conceptions and computational requirements.

- Semantic HMC
  - Ontology-described predictive model
  - Learned using Big Data technologies
  - Rule-based Web Reasoning to perform classification
Proposition: «Semantic HMC»

1. Indexation
2. Vectorization
3. Hierarchization
4. Resolution
5. Realization

Predictive Model Learning

Classify Items according to the Predictive Model
Proposition: « Semantic HMC »

1. Indexation
   - Extract terms
   - Index the items

2. Vectorization
   - Calculate term frequency vectors
   - Co-occurrence matrix

3. Hierarchization
   - Label selection
   - Hierarchical relations

4. Resolution
   - Classification rules creation

5. Realization
   - Ontology population
   - Rule-based Web Reasoning to classify items
Proposition: «Semantic HMC»

1. Indexation
2. Vectorization
Data
Co-occurrence matrix
3. Hierarchization
4. Resolution
Ontology-described Knowledge Base
Label Hierarchy
Classification Rules
New Data items
5. Realization
Classified items with labels

Unsupervised ontology learning

Rule-based Classification (Web Reasoner)
Outline

Context
- Global problem
- The Semantic HMC

Specific Problem
- Proposed Solution

Implementation
- Setup
- Results

Conclusion and future work
Specific Problem

**Rule-based reasoning to perform Classification**

1. Indexation
2. Vectorization

Data → Term index → Co-occurrence matrix → 4. Resolution → 3. Hierarchization → Ontology-described Knowledge Base → Classification Rules → Label Hierarchy → 5. Realization

- New Data items
- Classified items with labels

Unsupervised ontology learning
Rule-based Classification
Specific Problem

1. Indexation
2. Vectorization
3. Hierarchization
4. Resolution
5. Realization

- Resolution: Learn classifications rules from large volumes of unstructured text
  - Distributed method that exploits the cooccurrence matrix
- Realization: classify large volumes of new data items
  - Classification using a Web Reasonner
Proposed solution: rule learning (Resolution)

Learning **Alpha** and **Beta** sets

| $P_c(i|j)$ | term₁ | term₂ | term₃ | term₄ | term₅ | term₆ | term₇ |
|-----------|-------|-------|-------|-------|-------|-------|-------|
| label₁    | 0     | 0     | 5     | 0     | 5     | 25    | 25    |
| label₂    | 0     | 75    | 0     | 0     | 0     | 75    | 5     |
| label₃    | 0     | 0     | 75    | 0     | 25    | 0     | 0     |
| label₄    | 5     | 25    | 25    | 0     | 5     | 93    | 25    |
| label₅    | 95    | 0     | 0     | 0     | 60    | 0     | 5     |
| label₆    | 0     | 60    | 0     | 95    | 0     | 0     | 90    |
| label₇    | 5     | 98    | 5     | 60    | 25    | 0     | 79    |

Cooccurrence:  

$$P_c(term_i|term_j) = \frac{cfm(term_i, term_j)}{cfm(term_j, term_j)}$$

Alpha set:  

$$\omega^t_i = \{t_j | \forall t_j \in Term: P_c(t_i|t_j) > \alpha\}$$

Beta set:  

$$\omega^t_i = \{t_j | \forall t_j \in Term: \beta \leq P_c(t_i|t_j) \leq \alpha\}$$
Proposed solution: rule learning (Resolution)

Learning **Alpha** and **Beta** sets

**Alpha set:**

\[ \omega^t_i = \{ t_j | \forall t_j \in \text{Term}: P_c(t_i|t_j) > \alpha \} \]

**Beta set:**

\[ \omega^t_i = \{ t_j | \forall t_j \in \text{Term}: \beta \leq P_c(t_i|t_j) \leq \alpha \} \]
Proposed solution: rule learning (Resolution)

Example:

<table>
<thead>
<tr>
<th>%</th>
<th>term₁</th>
<th>term₂</th>
<th>term₃</th>
<th>term₄</th>
<th>term₅</th>
<th>term₆</th>
<th>term₇</th>
</tr>
</thead>
<tbody>
<tr>
<td>label₁</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>5</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>label₂</td>
<td>0</td>
<td>75</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>75</td>
<td>5</td>
</tr>
<tr>
<td>label₃</td>
<td>0</td>
<td>0</td>
<td>75</td>
<td>0</td>
<td>25</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>label₄</td>
<td>5</td>
<td>25</td>
<td>25</td>
<td>0</td>
<td>5</td>
<td>93</td>
<td>25</td>
</tr>
<tr>
<td>label₅</td>
<td>95</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>60</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>label₆</td>
<td>0</td>
<td>60</td>
<td>0</td>
<td>95</td>
<td>0</td>
<td>0</td>
<td>90</td>
</tr>
<tr>
<td>label₇</td>
<td>5</td>
<td>98</td>
<td>5</td>
<td>60</td>
<td>25</td>
<td>0</td>
<td>79</td>
</tr>
</tbody>
</table>

\[ \omega_{\alpha}^{t_i} = \{ t_j | \forall t_j \in \text{Term}: P_C(t_i | t_j) > \alpha \}, \alpha = 91 \]

\[ \omega_{\beta}^{t_i} = \{ t_j | \forall t_j \in \text{Term}: \beta \leq P_C(t_i | t_j) \leq \alpha \}, \beta = 70 \]
Classification at **query-time** using **backward-chaining**
<table>
<thead>
<tr>
<th>DL concepts</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Item \sqsubseteq \exists \text{hasTerm}.Term$</td>
<td>Items to classify (e.g. doc) has terms</td>
</tr>
<tr>
<td>$Term \sqsubseteq \top$</td>
<td>Terms (e.g. word) extracted from items</td>
</tr>
<tr>
<td>$Label \sqsubseteq Term$</td>
<td>Labels are terms used to classify items</td>
</tr>
<tr>
<td>$Label \sqsubseteq \forall \text{broader}.Label$</td>
<td>Broader relation between labels</td>
</tr>
<tr>
<td>$Label \sqsubseteq \forall \text{narrower}.Label$</td>
<td>Narrower relation between labels</td>
</tr>
<tr>
<td>$\text{broader} \equiv \text{narrower}^-$</td>
<td>Broader and narrower are inverse</td>
</tr>
<tr>
<td>$Item \cap Term = \emptyset$</td>
<td>Items and Terms are disjoint</td>
</tr>
<tr>
<td>$Item \sqsubseteq \forall \text{isClassified}.Label$</td>
<td>Relation that links items with labels</td>
</tr>
</tbody>
</table>
Outline

Context
- Global problem
- The Semantic HMC

Specific Problem
- Proposed Solution

Implementation
- Setup
- Results

Conclusion and future work
Implementation: rule creation

Distributed process using mapreduce:

Map

\begin{align*}
\text{label}_I \; \text{term}_J, \; P(I|J) \\
\text{label}_I \; \text{term}_M, \; P(I|M) \\
\text{label}_L \; \text{term}_K, \; P(L|K) \\
\text{label}_L \; \text{label}_M, \; P(L|M) \\
\text{label}_L \; \text{label}_N, \; P(L|N)
\end{align*}

Shuffle

\begin{align*}
\text{label}_I \; \text{term}_J, \; P(I|J) \\
\text{label}_I \; \text{term}_M, \; P(I|M) \\
\text{label}_I \; \text{label}_N, \; P(I|N) \\
\text{label}_L \; \text{term}_K, \; P(L|K) \\
\text{label}_L \; \text{label}_M, \; P(L|M)
\end{align*}

Reduce

\begin{align*}
\text{label}_I \; \text{term}_J, \; P(I|J) \\
\text{label}_I \; \text{term}_M, \; P(I|J) \\
\text{label}_I \; \text{label}_N, \; P(I|J) \\
\text{label}_L \; \text{term}_K, \; P(I|K) \\
\text{label}_L \; \text{label}_L, \; P(L|M)
\end{align*}

OWL API used to generate SWRL rules from the output

\[
Item(?it), Term(\text{term}_i), Label(\text{term}_i), hasTerm(?it, \text{term}_j) \rightarrow \\
isClassified(?it, \text{term}_i)
\]
Generated rules Exemple

### Alpha rules

<table>
<thead>
<tr>
<th>Rule</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Item(? it), Term(t_1), Label(term_i), hasTerm(? it, t_1) → isClassified(? it, term_i)</code></td>
<td></td>
</tr>
<tr>
<td><code>Item(? it), Term(t_2), Label(term_i), hasTerm(? it, t_2) → isClassified(? it, term_i)</code></td>
<td></td>
</tr>
</tbody>
</table>

### Beta rules

<table>
<thead>
<tr>
<th>Rule</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Item(? it), Term(t_1), Term(t_2), Label(term_i), hasTerm(? it, t_1), hasTerm(? it, t_2) → isClassified(? it, term_i)</code></td>
<td></td>
</tr>
<tr>
<td><code>Item(? it), Term(t_1), Term(t_3), Label(term_i), hasTerm(? it, t_1), hasTerm(? it, t_3) → isClassified(? it, term_i)</code></td>
<td></td>
</tr>
<tr>
<td><code>Item(? it), Term(t_2), Term(t_3), Label(term_i), hasTerm(? it, t_2), hasTerm(? it, t_3) → isClassified(? it, term_i)</code></td>
<td></td>
</tr>
</tbody>
</table>
Implementation: Classification at query-time

Stardog used as a scalable triple-store (compatible with **backward-chaining** inference as well as **SWRL** rules inference)

Rule selection process developed in Java interacting with Stardog to optimize query performance
# Implementation: test environment

## Dataset

![Wikipedia](image)

<table>
<thead>
<tr>
<th>Sub-Dataset</th>
<th>Number of articles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wikipedia 1</td>
<td>174900</td>
</tr>
<tr>
<td>Wikipedia 2</td>
<td>407000</td>
</tr>
<tr>
<td>Wikipedia 3</td>
<td>994000</td>
</tr>
</tbody>
</table>

## Cluster

![Google Cloud Platform](image)

<table>
<thead>
<tr>
<th>Resource type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of nodes</td>
<td>4</td>
</tr>
<tr>
<td>CPU (per node)</td>
<td>Intel Xeon E5 v2</td>
</tr>
<tr>
<td>RAM (per node)</td>
<td>7.5GB</td>
</tr>
<tr>
<td>Disk (per node)</td>
<td>500GB</td>
</tr>
</tbody>
</table>
## Implementation: parameter setup

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Step</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alpha Threshold</td>
<td></td>
<td>90</td>
</tr>
<tr>
<td>Beta Threshold</td>
<td>Resolution</td>
<td>80</td>
</tr>
<tr>
<td>Term ranking (n)</td>
<td>Resolution</td>
<td>5</td>
</tr>
<tr>
<td>p</td>
<td></td>
<td>0.25</td>
</tr>
<tr>
<td>Term Threshold (γ)</td>
<td>Realization</td>
<td>2</td>
</tr>
</tbody>
</table>
Results

Number of classifications: $\text{Item} \subseteq \forall \text{isClassified}.\text{Label}$
Number of **learned rules** (Alpha + Beta)
Number of **learned rules** (Alpha + Beta)

\[ \alpha = 90 \quad \beta = 80 \]

![Graph showing the relationship between the number of items and the number of learned rules.](image-url)
Outline

Context
- Global problem
- The Semantic HMC

Specific Problem
- Proposed Solution

Implementation
- Setup
- Results

Conclusion and future work
Conclusion

• A new unsupervised process to automatically classify items
  ▪ A highly scalable rule learning method based on statistical and lexical approaches
  ▪ A novel method to classify items using a web reasoner

• The process prototype was successfully implemented in a scalable and distributed platform to process Big Data

• Preliminary results show that the items are classified automatically by the reasonner
Ongoing and Future Work

- Quality Evaluation of the process: comparison with state-of-the-art in classification
- Predictive performance evaluation based on cross-validation with large dataset
- Optimization of the process by exhaustive analysis of parameters’ impact
- Application to classification of news articles on the web
An unsupervised classification process for large datasets using web reasoning

Thank you!

Rafael PEIXOTO, Thomas HASSAN, Christophe CRUZ, Aurelie BERTAUX, Nuno SILVA
thomas.hassan@u-bourgogne.fr

Laboratoire LE2I – UMR CNRS 6306 – Université de Bourgogne