Conceptual Similarity: Why, Where, How

Michalis Sfakakis

Laboratory on Digital Libraries & Electronic Publishing, Department of Archives and Library Sciences, Ionian University, Greece

First Workshop on Digital Information Management, March 30-31, 2011
Corfu, Greece
Do we need similarity?

☐ Are the following objects similar?

■ (Similarity, SIMILARITY)

☐ As character sequences, NO!

■ How do they differ?

☐ As character sequences, but case insensitive, Yes!

☐ As English words, Yes!

■ Same word! They have the same definition, written differently
Contents

- Introduction
- Disciplines
- How we measure similarity
  - Focus on Ontology Learning evaluation
Exploring similarity... more cases

- What about the similarity of the objects?
  - (1, a)
    - The first object is the number one and the second is the first letter of the English alphabet. Therefore, as the first is a number and the second is a letter, they are different!
  - But, conceptually... When both represent an order, e.g. a chapter, or a paragraph number, they are both representing the first object of the list, the first chapter, paragraph, etc. Therefore, they could be considered as being similar!
Results for an Information Need

How similar are the Results? Which one to select?
Comparing Concepts

... again, how similar are the following objects?

(Disease, Illness)

- As English words, or as character sequences they are not similar!
  - How do they differ?

- As synonymous terms in a Thesaurus, they are both representing the same concept.
  (related with the equivalency relationship)
Comparing Hierarchies

- How similar...
  - ... is the node *car* from the left hierarchy to the node *auto* from the right hierarchy?
  - ... are the nodes *van* from both hierarchies?
  - ... is the above hierarchies?

* [Dellschaft and Staab, 2006]
... so, what similarity is?

- Similarity is a context dependent concept

- Merriam-Webster’s Learner’s dictionary defines similarity as*:
  - A quality that makes one person or thing like another
  - ... and similar, having characteristics in common

- Therefore, the context and the characteristics in common are required in order to specify and measure similarity

* [http://www.learnersdictionary.com/search/similarity](http://www.learnersdictionary.com/search/similarity)
Where the concept of similarity is encountered

- Similarity is a context dependent concept

- Machine learning
  - Ontology Learning
  - Schema & Ontology Matching and Mapping
  - Clustering
  - IR
  - ... in any evaluation concerning the results of a pattern recognition algorithm

- Vital part of the Semantic Web development
Precision & Recall in IR, measuring similarity between answers

- Let $C$ be the result set for a query (the retrieved documents, i.e. the *Computed* set)

- Also, we need to know the correct results for the query (all the relevant documents, the *Reference* set)
  - *Precision*: is the fraction of retrieved documents that are relevant to the search
  - *Recall*: is the fraction of the documents that are relevant to the query that are successfully retrieved

\[
\text{precision} = \frac{|\text{relevant documents} \cap \text{retrieved documents}|}{|\text{retrieved documents}|} \quad \text{recall} = \frac{|\text{relevant documents} \cap \text{retrieved documents}|}{|\text{relevant documents}|}
\]

... Precision & Recall, a way to measure similarity

- *Precision & Recall* are two widely used metrics for evaluating the correctness of a pattern recognition algorithm.

- *Recall* and *Precision* depend on the outcome (oval) of a pattern recognition algorithm and its relation to all relevant patterns (left) and the non-relevant patterns (right). The more correct results (green), the better.
  - *Precision*: horizontal arrow.
  - *Recall*: diagonal arrow.

Precision & Recall, once more

- **Precision**
  \[ P = \frac{|R \cap C|}{|R|} \]

- **Recall**
  \[ R = \frac{|R \cap C|}{|C|} \]

- **True Positive (TP)** = \( R \cap C \)
- **False Negative (FN)** = \( R - C \)
- **False Positive (FP)** = \( C - R \)
- **True Negative (TN)** = \( D - (R \cup C) \)
Overall evaluation, combining Precision & Recall

- Given Precision & Recall, F-measure could combine them for an overall evaluation

- Balanced F-measure ($P$ & $R$ are evenly weighted)
  - $F_1 = \frac{2(PR)}{P+R}$

- Weighted F-measure
  - $F_b = \frac{(1+b^2)PR}{b^2P+R}$, $b$ non-zero
  - $F_1$ ($b=2$) weights recall twice as much as precision
  - $F_{0.5}$ ($b=0.5$) weights precision twice as much as recall
A simplified definition of a core ontology*: 

The structure $O := (C, root, \leq_C)$ is called a core ontology. $C$ is a set of concept identifiers and $root$ is a designated root concept for the partial order $\leq_C$ on $C$. This partial order is called concept hierarchy or taxonomy. The equation $\forall c \in C : c \leq_C root$ holds for this concept hierarchy.

Levels of comparison

- Lexical, how terms are used to convey meanings
- Conceptual, which conceptual relations exist between terms
- ...

* [Dellschaft and Staab, 2006]
Gold Standard based Evaluation of Ontology Learning

- Given a pre-defined ontology
  - The so-called Gold Standard or Reference
- Compare the Learned (Computed) Ontology with the Gold Standard
Measuring Similarity - Lexical Comparison Level – LP, LR

- **Lexical Precision & Lexical Recall**
  - LP($O_C, O_R$) = $|C_C \cap C_R|/|C_C|$
  - LR($O_C, O_R$) = $|C_C \cap C_R|/|C_R|$

- The lexical precision and recall reflect how good the learned lexical terms $C_C$ cover the target domain $C_R$
- For the above example LP=4/6=0.67, LR=4/5=0.8
Measuring Similarity, Lexical Comparison Level - aSM

- **Average String Matching**, using edit distance
  - *Levenshtein distance*, the most common definition for edit distance, measures the minimum number of token insertions, deletions and substitutions required to transform one string into another.

- For example*, the *Levenshtein distance* between "kitten" and "sitting" is 3 (there is no way to do it with fewer than three edits):
  - kitten → sitten (substitution of 's' for 'k')
  - sitten → sittin (substitution of 'i' for 'e')
  - sittin → sitting (insertion of 'g' at the end).

Measuring Similarity, Lexical Comparison Level – String Matching

- **String Matching measure (SM)**, given two lexical entries $L_1$, $L_2$
  - Weights the number of the required changes against the shorter string
  - 1 stands for perfect match, 0 for bad match

- **Average SM**
  - Asymmetric, determines the extend to which $\mathcal{L}_1$ (target) is covered by $\mathcal{L}_2$ (source)

$$\text{SM}(L_i, L_j) := \max \left( 0, \frac{\min(|L_i|, |L_j|) - \text{ed}(L_i, L_j)}{\min(|L_i|, |L_j|)} \right) \in [0, 1]$$

$$\overline{\text{SM}}(\mathcal{L}_1, \mathcal{L}_2) := \frac{1}{|\mathcal{L}_1|} \sum_{L_i \in \mathcal{L}_1} \max_{L_j \in \mathcal{L}_2} \text{SM}(L_i, L_j)$$

[Maedche and Staab, 2002]
Measuring Similarity, Lexical Comparison Level - \( \text{RelHit} \)

- Relative Number of Hits
  \[
  \text{RelHit}(\mathcal{L}_1, \mathcal{L}_2) := \frac{|\mathcal{L}_1 \cap \mathcal{L}_2|}{|\mathcal{L}_1|}
  \]

- \( \text{RelHit} \) actually express Lexical Precision

- \( \text{RelHit} \) Compared to average String Matching
  - \( \text{Average SM} \) reduces the influences of string pseudo-differences (e.g. singular vs. plurals)
  - \( \text{Average SM} \) may introduce some kind of noise, e.g. “power”, “tower”
Measuring Similarity, Conceptual Comparison Level

- Conceptual level compares semantic structure of ontologies
- Conceptual structures are constituted by Hierarchies, or by Relations
- How to compare two hierarchies?
- How do the positions of concepts influence similarity of Hierarchies?
- What measures to use?
Local measures compare the positions of two concepts based on characteristics extracts from the concept hierarchies they belong to.

Some characteristic extracts:

- **Semantic Cotopy (sc)**
  
  \[ sc(c, O) = \{ c_i | c_i \in C \land (c_i \leq c \lor c \leq c_i) \} \]

- **Common Semantic Cotopy (csc)**
  
  \[ csc(c, O_1, O_2) = \{ c_i | c_i \in C_1 \cap C_2 \land (c_i \leq c \lor c \leq c_i) \} \]
Measuring Similarity, Conceptual Comparison Level – \textit{sc}

- **Semantic Cotopy**
  \[ \text{sc}(c, O) = \{c_i | c_i \in C \land (c_i \leq c \lor c \leq c_i)\} \]

- **Semantic Cotopy examples**
  - \text{sc}("root", O_R) = \{root, bike, car, van, coupé\}
  - \text{sc}("root", O_C) = \{root, bike, auto, BMX, van, coupé\}
  - \text{sc}("bike", O_R) = \{root, bike\}
  - \text{sc}("bike", O_C) = \{root, bike, BMX\}
  - \text{sc}("car", O_R) = \{root, car, van, coupé\}
  - \text{sc}("auto", O_C) = \{root, auto, van, coupé\}
Measuring Similarity, Conceptual Comparison Level – csc

- Common Semantic Cotopy
  - \( csc(c, O_1, O_2) = \{c_i | c_i \in C_1 \cap C_2 \land (c_i <_1 c \lor c <_2 c_i)\} \)

- Common Semantic Cotopy examples
  - \( C_1 \cap C_2 = \{\text{root, bike, van, coupé}\} \)
  - \( csc(“\text{root}”, o_R, o_C) = \{\text{bike, van, coupé}\} \)
  - \( csc(“\text{root}”, o_C, o_R) = \{\text{bike, van, coupé}\} \)
  - \( csc(“\text{bike}”, o_R, o_C) = \{\text{root}\}, csc(“\text{bike}”, o_C, o_R) = \{\text{root}\} \)
  - \( csc(“\text{car}”, o_R, o_C) = \{\text{root, van, coupé}\}, csc(“\text{car}”, o_C, o_R) = \emptyset \)
  - \( csc(“\text{auto}”, o_C, o_R) = \{\text{root, van, coupé}\} \), \( csc(“\text{auto}”, o_R, o_C) = \emptyset \)
Measuring Similarity, Conceptual Comparison Level – local measures $tp$, $tr$

- Local taxonomic precision using characteristic extracts
  \[ tp_{ce}(c_1, c_2, O_C, O_R) = \frac{|ce(c_1, O_C) \cap ce(c_1, O_R)|}{|ce(c_1, O_C)|} \]

- Local taxonomic recall using characteristic extracts
  \[ tr_{ce}(c_1, c_2, O_C, O_R) = \frac{|ce(c_1, O_C) \cap ce(c_1, O_R)|}{|ce(c_1, O_R)|} \]
Measuring Similarity, Conceptual Comparison Level – local measures tp

- Local taxonomic precision examples using sc

- \( sc(\text{“bike”}, O_R) = \{\text{root}, \text{bike}\} \),
  \( sc(\text{“bike”}, O_C) = \{\text{root}, \text{bike}, \text{BMX}\} \)

- \( tp_{sc}(\text{“bike”}, \text{“bike”}, O_C, O_R) = |\{\text{root, bike}\}|/|\{\text{root, bike, BMX}\}|, \\
  tp_{sc}(\text{“bike”}, \text{“bike”}, O_C, O_R) = 2/3 = 0.67 \)

[Maedche and Staab, 2002]
Local taxonomic precision examples using sc

- $\text{sc}(\text{“car”}, O_R) = \{\text{root}, \text{car}, \text{van}, \text{coupé}\}$,
- $\text{sc}(\text{“auto”}, O_C) = \{\text{root}, \text{auto}, \text{van}, \text{coupé}\}$

- $tp_{sc}(\text{“car”}, \text{“auto”}, O_C, O_R) = \frac{|\{\text{root, van, coupé}\} | \cap |\{\text{root, auto, van, coupé}\}|}{|\{\text{root, van, coupé}\} | \cup |\{\text{root, auto, van, coupé}\}|} = 3/4 = 0.75$
Measuring Similarity, Conceptual Comparison Level – comparing Hierarchies

Global Taxonomic Precision (TP)

\[
TP(O_C, O_R) := \frac{1}{|C_C|} \sum_{c \in C_C} \begin{cases} 
    tp(c, c, O_C, O_R) & \text{if } c \in C_R \\
    \max_{c' \notin C_R} tp(c, c', O_C, O_R) & \text{if } c \notin C_R 
\end{cases}
\]
Measuring Similarity, Conceptual Comparison Level – Overall evaluation

- ... again F-measure, but now using **Global Taxonomic Precision (TP)** and **Global Taxonomic Recall (TR)**

- Balanced Taxonomic **F-measure** (TP & TR are evenly weighted)
  - $TF_1 = \frac{2*(TP*TR)}{(TP+TR)}$

- Weighted **TF-measure**
  - $TF_b = \frac{(1+b^2)*(TP*TR)}{(b^2*TP+TR)}$, $b$ non-zero
  - $TF_1 (b=2)$ weights recall twice as much as precision
  - $TF_{0.5} (b=0.5)$ weights precision twice as much as recall
Measuring Similarity, Conceptual Comparison Level – Taxonomic Overlap

- **Global Taxonomic Overlap... based on local taxonomic overlap (TO)**

\[
\overline{TO}(O_1, O_2) = \frac{1}{|C_1|} \sum_{c \in C_1} TO(c, O_1, O_2)
\]

\[
TO(c, O_1, O_2) = \begin{cases} 
TO'(c, O_1, O_2) & \text{if } c \in C_2 \\
TO''(c, O_1, O_2) & \text{if } c \notin C_2 
\end{cases}
\]

\[
TO'(c, O_1, O_2) := \frac{|SC(c, O_1, O_2) \cap SC(c, O_2, O_1)|}{|SC(c, O_1, O_2) \cup SC(c, O_2, O_1)|}
\]

\[
TO''(c, O_1, O_2) := \max_{c' \notin C_2} \frac{|SC(c, O_1, O_2) \cap SC(c', O_2, O_1)|}{|SC(c, O_1, O_2) \cup SC(c', O_2, O_1)|}
\]
References & Further Reading


End of tutorial!

☐ Thanks for your attention!

☐ Michalis Sfakakis