Improving Robustness and Flexibility of Concept Taxonomy Learning from Text

F. Leuzzi[1], S. Ferilli[1,2], C. Taranto[1], and F. Rotella[1]

[1] Dipartimento di Informatica – Università di Bari
{fabio.leuzzi, fulvio.rotella}@uniba.it
{ferilli, claudio.taranto}@di.uniba.it
[2] Centro Interdipartimentale per la Logica e sue Applicazioni – Università di Bari

Abstract. The spread and abundance of electronic documents requires automatic techniques for extracting useful information from the text they contain. The availability of conceptual taxonomies can be of great help, but manually building them is a complex and costly task. Building on previous work, we propose a technique to automatically extract conceptual graphs from text and reason with them. Since automated learning of taxonomies needs to be robust with respect to missing or partial knowledge and flexible with respect to noise, this work proposes a way to deal with these problems. The case of poor data/sparse concepts is tackled by finding generalizations among disjoint pieces of knowledge. Noise is handled by introducing soft relationships among concepts rather than hard ones, and applying a probabilistic inferential setting. In particular, we propose to reason on the extracted graph using different kinds of relationships among concepts, where each arc/relationship is associated to a number that represents its likelihood among all *possible worlds*, and to face the problem of sparse knowledge by using generalizations among distant concepts as bridges between disjoint portions of knowledge.

1 Introduction

The spread and abundance of electronic documents requires automatic techniques for extracting useful information from the text they contain. The availability of conceptual taxonomies can be of great help, but manually building them is a complex and costly task. Obtaining automatically *Full Text Understanding* is not trivial, due to the intrinsic ambiguity of natural language and to the huge amount of common sense and linguistic/conceptual background knowledge needed to switch from a purely syntactic representation to the underlying semantics. Nevertheless, even small portions of such knowledge may significantly improve understanding performance, at least in limited domains. Although standard tools, techniques and representation formalisms are still missing, lexical and/or conceptual taxonomies can provide a useful support to many NLP tasks, allowing automatic systems to exploit different kinds of relationships that are implicit in the text but required to correctly understand it. Building on previous work, we propose a technique to automatically extract conceptual graphs from text and reason with them. Since automated learning of taxonomies needs to be robust with respect to missing or partial knowledge and flexible with respect to noise and to the need of shifting the representation, this work proposes a way to deal with these problems. Data poorness, concept sparsity and representation shift are tackled by finding generalizations among disjoint pieces of knowledge in order to build a bridge between them. Noise is handled by introducing soft relationships among concepts rather than hard ones, and applying a probabilistic inferential setting to reason on the extracted graph using different kinds of relationships, where each arc/relationship is associated to its likelihood of being true in all *possible worlds*.

This work is organized as follows: the next section describes related works; Section 3 outlines the proposed approach; then we present an evaluation of our solution; lastly we conclude with some considerations and future works.

2 Related Work

Many approaches have been attempted to build taxonomies and ontologies from text. [1] builds concept hierarchies using Formal Concept Analysis by grouping objects with their attributes, which are determined from text linking terms with verbs. A few examples of other approaches: [10, 9] build ontologies by labelling taxonomical relations only, while we label also non-taxonomical ones with actions (verbs); [12] builds a taxonomy considering only concepts that are present in a domain but do not appear in others, while we are interested in all recognized concepts independently of their being generic or domain-specific; [11] defines a language to build formal ontologies, while we are interested in the lexical level.

As regards our proposal, for the syntactic analysis of the input text we exploit the *Stanford Parser* and *Stanford Dependencies* [8, 2], two very effective tools that can identify the most likely syntactic structure of sentences (including active/passive and positive/negative forms), and specifically 'subject' or '(direct/indirect) object' components. They also normalize the words in the input text using lemmatization instead of stemming, which allows to distinguish their grammatical role and is more comfortable to read by humans.

We also apply probabilistic reasoning on the extracted knownledge using ProbLog [13]. It is essentially Prolog where all clauses are labelled with the probability that they are true, that in turn can be extracted from large databases by various techniques. A ProbLog program $T = \{p1 : c1, ..., pn : cn\}$ specifies a probability distribution over all its possible non-probabilistic subprograms according to the theoretical basis in [14]. The semantics of ProbLog is then defined by the success probability of a query, which corresponds to the probability that the query succeeds in a randomly sampled program. Indeed, the program can be split into a set of labelled facts $p_i :: f_i$, meaning that f_i is a fact with probability of occurrence p_i, and a Prolog program using those facts, which encodes the background knowledge (BK). Probabilistic facts correspond to mutually independent random variables (RVs), which together define a probability distribution over all ground logic programs $L \subseteq L_T$ (where L_T is the set of all f_i's):

$$P(L|T) = \prod_{f_i \in L} p_i \prod_{f_i \in L_T \setminus L} (1 - p_i)$$

In this setting we will use the term *possible world* to denote the least Herbrand model of a subprogram L together with BK, and we will denote by L both the set of sampled facts and the corresponding world.

Lastly, we need in some steps of our technique to assess the similarity among concepts in a given conceptual taxonomy. A classical, general measure, is the *Hamming distance* [6], that works on pairs of equal-lenght vectorial descriptions and counts the number of changes required to turn one into the other. Other measures, specific for conceptual taxonomies, are $mathrmsf_{Fa}$ [4] (that adopts a global approach based on the whole set of hypernyms) and sf_{WP} [15] (that focuses on a particular path between the nodes to be compared).

3 Proposed Approach

This proposal relies on a previous work [5], in which we assume that each noun in the text corresponds to an underlying *concept* (phrases can be preliminarily extracted using suitable techniques, and handled as single terms). A concept is described by a set of characterizing attributes and/or by the concepts that interact with it in the world described by the corpus. The outcome is a graph, where nodes are the concepts/nouns recognized in the text, and edges represent the relationships among these nodes, expressed by verbs in the text (whose direction denotes their role in the relationship). In this work, we handle noise by weighting the relationships among concepts, this way obtaining a semantic network with soft links between nodes.

3.1 Graph Construction

Natural language texts are processed by the Stanford Parser in order to extract triples $\langle$*subject*, *verb*, *complement*$\rangle$ that will represent the concepts (the *subject*s and *complement*s) and attributes (*verb*s) for the graph. We have adopted some representational tricks: indirect complements are treated as direct ones by embedding the corresponding preposition into the verb; sentences involving verb 'to be' or nouns with adjectives contributed in building the sub-class structure of the taxonomy (e.g., "the penguin is a bird" yields *is_a*(*penguin*,*bird*)). In this way we have two kind of edges among nodes in the graph: verbal ones, labelled with the verb linking the two concepts, and taxonomic (is_a) ones. In order to enrich the representation formalism previously defined, we analyzed the syntatic tree to seize the sentence positive or negative form based on the absence or presence (respectively) of a *negation modifer* for the verb. Moreover we decided to take into account separately the frequency of each arc between the concepts in positive and negative sentences.

This setting allowed us to give robusteness to our solution through a statistical approach. In fact, the obtained taxonomy could be inspected and used by filtering out all portions that do not pass a given level of reliability. This could be useful for the identification of relevant concept, as shown in [5], or for other applications that will be explained in the next two subsections.

3.2 Probabilistic Reasoning 'by association'

Reasoning 'by association' means finding a path of pairwise related concepts that establishes an indirect interaction between two concepts c' and c'' in the semantic network. However, since real world data are typically noisy and uncertain, there is a need for strategies that soften the classical rigid logical reasoning. In particular, when knowledge is learned from text, as in our case, we might run into the above problems, which require an inference engine that allows to perform several kinds of probabilistic queries, like getting the minimal path, or choosing the best (i.e., the most likely) path, rather than computing the exact probability of all possibile ones.

In this work we propose two reasoning stategies: the former aims at obtaining the minimal path between concepts together with all involved relations, the latter relies on ProbLog in order to allow probabilistic queries on the conceptual graph.

In the former strategy, we look for a minimal path using a *Breadth-First Search* (BFS) technique, applied to both concepts under consideration. The expansion steps of the two processes are interleaved, checking at each step whether the new set of concepts just introduced has a non-empty intersection with the set of concepts of the other process. When this happens, all the concepts in such an intersection identify one or more shortest paths connecting c' and c'', that can be retrieved by tracing back the parent nodes at each level in both directions up to the roots c' and c''. Since this path is made up of concepts only, to obtain a more sensible 'reasoning' it must be filled with the specific kind of interaction represented by the labels of edges (verbs) that connect adjacent concepts in the chain. We also provide the number of positive/negative instances, and the corresponding ratios over the total, to help understanding different gradations (such as permitted, prohibited, typical, rare, etc.) of actions between two objects.

The latter strategy requires the definition of a formalism based on ProbLog language. We used $p_i :: f_i$ where f_i is a ground literal of the form *link*(*subject*, *verb*, *complement*) and p_i is ratio between the sum of all examples for which f_i holds and the sum of all possibile links between *subject* and *complement*. Figure 1 shows an example where many links are present between *farmer* and *plant*, expressing different kinds of interactions between these two concepts on which the above kinds of reasoning can be applied. One might be interested in getting minimal paths, regardless of their probability, from *farmer* to *trasporter*, reporting all traversed concepts and verbs. Or, one might want to compute the *explanation probability* among all possibile path in the SLD-tree between these two nodes, obtained as the product of all traversed links together with the proof of the query (most likely explanation). One of the problems of these approaches is the tight connection between the quality of the reasoning results and that

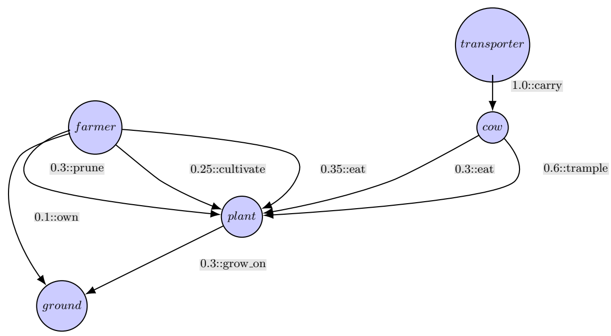

Fig. 1. Example of ProbLog Network

of the network, in turn depending on the processed texts. Indeed, if two nodes belong to disjoint regions of the graph, reasoning cannot succeed. We tackle this problem by defining a generalization operator as follows.

3.3 Generalization operator

Let us first provide a more precise account of our generalization.

Definition. (Generalization)
Given two concepts G and C, G generalizes C if anything that can be labelled as C can be labelled as G as well, but not *vice-versa*. ■

The use of generalizations provides many opportunities of enrichment and/or manipulations on the graph:

1. building taxonomic structures by (repeatedly) applying this operator, also after the addition of new text (possibly causing the presence of new nodes in the graph);
2. shifting the representation, by removing the generalized nodes from the graph and leaving just their generalization (that inherits all their relationships to other concepts);
3. extending the amount of relationships between concepts belonging to the same connected component of the graph, or building bridges between disjoint components that open new (previously impossible) reasoning paths.

In particular, as regards the third case, it can be viewed as a tool that aids reasoning 'by association' to reach its objectives, and hence as a multi-strategy setting in which induction and deduction cooperate.

More formally, we provide the following definition.

Definition. (Bridge)
Given a piece of knowledge K represented as a concept graph made up of different connected components, a *bridge* in K is a modification of the graph that connects some of such components, allowing to reach any node of either from any node of another. A *bridge* that unifies n isolated sub-graphs is named *n-way bridge*. ■

Regardless of the task, three steps can be outlined to describe the general procedure:

1. *Concept Grouping*, in which all concepts are grossly partitioned to obtain subsets of concepts: we group similar concepts if the aim is to enrich the relationships, or dissimilar ones in the bridging perspective (Algorithm 1);
2. *Word Sense Disambiguation*, that associates a single meaning to each term by solving possible ambiguities using the domain of discourse;
3. *Computation of taxonomic similarity*, in which WordNet [3] is exploited in order to further filter with an external source the groups found in step 1, and to choose an appropriate subsumer (Algorithm 2).

To generalize two or more concepts, we use as their description their direct neighbor concepts and the verbs (in positive or negative form) used to connect them. Thus, considering the set *Attributes* of all verbs in their positive and negative forms, we build a $Concepts \times (Concepts \cup Attributes)$ matrix $\mathcal{C}$ where: $\mathcal{C}_{i,j} = 1$ if j denotes a concept column and there is at least a relationship between concepts i and j; $\mathcal{C}_{i,j} = 1$ if j denotes an attribute column and there is at least a relationship between concept i and verb j; $\mathcal{C}_{i,j} = 0$ otherwise. Each row is a feature vector describing the concept, and hence two vectors can be compared according to the *Hamming distance*. Pairwise clustering under the complete link assumption is applied to these descriptions: initially, each non-null row becomes a singleton cluster; then, clusters are merged while a merging condition is fulfilled. In its standard view, complete link states that *the distance of the farthest items of the involved clusters must be less than a given threshold*.

As stated earlier, generalizations can be carried out for different objectives, affecting the way complete link is applied. In particular, if the objective is the enrichment of relationships within connected components, it is applied in the standard way, otherwise, if the objective is to build bridges, *the distance of the closer items of the involved clusters must be greater than a given threshold*. When the condition is satisfied, the average score between all pairs of items in the two clusters is saved, and only the pair of clusters corresponding to the smallest (respectively, greatest) average is merged. We define more formally the clustering of dissimilar objects as follows.

Definition. (Inverse clustering)
Given a set of objects and a distance measure, the *Inverse clustering* is obtained grouping iteratively most dissimilar objects until a given threshold of distance is satisfied. ■

Now, the clusters contain similar (resp., dissimilar) concepts that can be generalized in order to create new relationships (resp, to merge nodes) for the enrichment (resp., bridging), but this procedure alone might not be reliable, because terms that occur seldom in the corpus have few connections (which would affect their cluster assignment due to underspecification) and because the expressive power of this formalism is too low to represent complex contexts (which would affect even more important concepts). Note that the bridging setting is less affected by the underspecification problem, because it tends to group dissimilar concepts. Since underspecification corresponds to almost zero vectors, taking dissimilar vectors we tend to group the most specified distant vectors. Indeed, since there cannot be any 1 in the same positions in both descriptions (because this would mean that there exists a path between them), the more 1's overall in the two descriptions, the larger their distance, which means that the bridge is merging two hub (i.e., highly connected) nodes. This clearly improves the quality of the bridge. This solution allows to limit the shortest medium length among all possible paths built between the sub-graphs that the bridge connects. However, the support of an external resource might be desirable. We consider WordNet as a sensible candidate for this, and try to map each concept in the network to the corresponding synset (a non trivial problem due to the typical polysemy of many words) using the *one domain per discourse* assumption as a simple criterion for Word Sense Disambiguation, whose algorithm is described in [5]. Thus, WordNet allows to check and confirm/reject the similarity of concepts belonging to the same cluster, by considering all possible pairs of concepts whose similarity is above a given threshold. Let us state this more formally.

Definition. (Star)
Given a set of unordered pairs of items S and another unordered pair $P = \{A, B\}$ s.t. $P \in S$, its *Star* is a set of pairs S' s.t. $\forall P' \in S', (A \subset P' \vee B \subset P') \wedge P' \neq P$ (i.e., the set of pairs of items that contain A or B). ■

Definition. (Generalization set)
Given a cluster of concepts C, a similarity function $sf()$, a similarity threshold T, and a pair $P = \{A, B\}$ s.t. $P \in C, P$ is a *Generalization set* if it can be covered by a most specific common subsumer G, $sf(P) > T$, and $\forall P' \in C, sf(P) > sf(P')$ (i.e., has the largest similarity value among all possible pairs). Moreover, each pair P'' in the *Star* of P s.t. $P'' > T$ that can be covered from G, is used to extend the *Generalization set*. ■

Where *most specific common subsumer* is the first common node found exploring the hypernym relations in WordNet. Similarity is determined using a mix of the measures proposed in [4] and in [15], to consider both a global perspective which avoids the choice of a single path and the actual generalizing path:

$$sf(A, B) = sf_{Fa}(A, B) \cdot sf_{WP}(A, B)$$

The underlying idea is that the former general measure can be weighted with the latter real measure that represent the single path from which the subsumer is taken, and *vice-versa*.

4 Evaluation

The proposed approach was evaluated using *ad-hoc* tests that may indicate its strengths and weaknesses. Due to lack of space, only a few selected outcomes will be reported here. Although preliminary, these results seem enough to suggest that the approach is promising. We exploited a dataset made up of documents concerning *social networks* on socio-political and economic topic. It was chosen deliberately small in order to tackle problems as noise and poor knowledge.

Algorithm 1 Pair-wise clustering of all concepts in the network.

Input: $matrix\ C \times\ C{+}A$ (where C is the set of objects/concepts, and A is the set of positive and negative verbs), that for each cell has the value 1 if at least one link exist, 0 otherwise; $THRESHOLD$ for Hamming distance.
Output: set of clusters.

```
pairs ← empty
averages ← empty
for all O_i | O_i ≠ zero_vector ∧ i ∈ C do
  newCluster ← O_i
  clusters.add(newCluster)
end for
for all pair(C_k, C_z) | C ∈ clusters ∧ k, z ∈ [0, clusters.size] do
  if completeLink(C_k, C_z) then
    pairs.add(C_k, C_z)
    averages.add(getScoreAverage(C_k, C_z))
  end if
end for
pair ← getPairWithMaxMin(pairs, averages)
merge(pair)
```

completeLink $\rightarrow$ check the complete link assumption for the passed clusters.
getPairWithMaxMin $\rightarrow$ get the pair with the maximum or minimum average depending on task.

4.1 Probabilistic Reasoning 'by association'

Table 1 shows a sample of outcomes of reasoning 'by association'. In this experiment we want to investigate the minimum path (if any) between two nodes in the graph. In order to show the strength of this kind of reasoning, each verb is labelled with the frequency with which it occurs in the paths *subject, verb, complement*. Looking at case 1, we wanted to explore the relationship between *young* and *schoolwork*. This chain includes verb *look*, that occurs only in positive sentences with probability 1.0; this means that our knowledge is coherently indicating that *young* "always" *look television*. In the same case there is the relationship between *facebook* and *schoolwork*, in fact the verb *help* appears with probability 0.25 in positive and 0.75 in negative sentences. We can interpret this behaviour as *facebook* "may" *help schoolwork*, or with the specific associated probability. We need to point out that this reasoning strategy shows all possibile verbs between each pair of adjacent nodes in the path.

On the other hand, Table 2 shows a sample of probabilistic queries executed on the same paths. We have computed the exact success probability, the most likely explanation probability and an approximate probability according to the MonteCarlo method implemented in [7] keeping a 95% of confidence interval. The last type of query has been provided because exact inference can be intractable even for small networks, and so sometimes it can be reasonable to make approximate inference. The probability of each sentence was computed according to the criteria in section 3.2. For instance, in case 2, the explanation proof entails the set {*(young, look, television), (television, talk_about, facebook), (facebook, not_help, schoolwork)*}, respectively with probability {1.0, 0.75, 0.5}, whose product is 0.375.

Algorithm 2 Effective generalization research.

Input: the set of C clusters returned by pair-wise clustering; T similarity threshold; max the max number of generalizations to try to extract from a cluster.
Output: set of candidate generalizations.

```
generalizations ← empty set
for all c ∈ C do
  good_pairs ← empty set
  for all pair(O_i, O_j) | i, j ∈ c do
    if similarity_score(pair(O_i, O_j)) > T then
      good_pairs.add(pair(O_i, O_j), wordnet_hypernym(pair(O_i, O_j)))
    end if
  end for
  for all i ∈ [0, max] do
    if good_pairs ≠ empty set then
      new_set ← {good_pairs.getBestPair, good_pairs.getStar}
      generalizations.add(new_set)
      good_pairs.remove(new_set)
    end if
  end for
end for
```

good_pairs → contains a list of pairs that satisfy T, with their relative subsumer.

good_pairs.getBestPair → get the pair that has the best similarity score.

good_pairs.getStar → get the Star of the pair

good_pairs.remove → remove all pairs in the passed set.

wordnet_hypernym → get the subsumer discovered in WordNet for the passed pair.

4.2 Evaluation of generalization operator

Two toy experiments are reported for concept generalization, the former aimed to the enrichment of relationships, the latter with the bridging perspective. The maximum threshold for the *Hamming distance* was set to 0.001, while the minimum threshold for *taxonomic similarity* was fixed at 0.45 in both. The exploited dataset included 669 objects (subjects and/or complements) and 727 verbs.

In Table 3 we can see that, coherently with the enrichment-only perspective, no bridges are built. Conversely, applying *Inverse clustering* we obtain as expected also bridges among two or more disjoint graph regions. Analyzing the two conceptual similarity measures in both experimental settings, they both return very high values for almost all pairs, leading to final scores that neatly pass

Table 1. Examples of smooth reasoning 'by association' throught BFS (start and target nodes in emphasis).

#	Subject	Verb	Complement
1	*young*	look [Pos: 3/3]	television
	television	talk_about [Pos: 3/3], critic [Pos: 1/1]	facebook
	facebook	help [Pos: 1/4, Neg: 3/4], distract [Pos: 1/1]	*schoolwork*
2	*people*	be_in [Pos: 1/1]	group
	group	invite [Pos: 2/2]	facebook
	facebook	help [Pos: 1/1]	*individual*
3	everyone	want [Pos: 5/5]	*occupation*
	occupation	maintain [Pos: 1/1]	lifestyle
	lifestyle	see_in [Pos: 1/1], change_in [Pos: 1/1],	*media*

Table 2. Examples of probabilistic reasoning 'by association' throught ProbLog.

#	Query	Probability
1	problog_exact(path(*young* - *schoolwork*))	0.530
2	problog_max(path(*young* - *schoolwork*))	0.375
3	problog_approx(path(*young* - *schoolwork*))	0.555
4	problog_exact(path(*people* - *individual*))	0.167
5	problog_max(path(*people* - *individual*))	0.167
6	problog_approx(path(*people* - *individual*))	0.162
7	problog_exact(path(*occupation* - *media*))	0.750
8	problog_max(path(*occupation* - *media*))	0.500
9	problog_approx(path(*occupation* - *media*))	0.744

Table 3. Generalizations for pairwise clustering of similar concepts, and corresponding conceptual similarity scores (bottom).

#	Bridge	Subsumer	Subs. Domain	Concepts	Conc. Domain
1	No	variable [105857459]	mathematics	variable [105857459] factor [105858317]	mathematics mathematics
2	No	person [100007846]	biology, person	type [109909060] collegian [109937056]	person factotum
3	No	person [100007846]	biology, person	type [109909060] model [110324851]	person person
4	No	integer [113728499]	mathematics	nineteen [113747989] forty [113749527]	number number
5	No	person [100007846]	biology, person	scholar [110251779] job [110222949] name [110344443]	school person person

#	Pairs	Fa score	WP score	Score
1	variable, factor	0.7	0.857	0.6
2	type, collegian	0.659	0.737	0.486
3	type, model	0.692	0.778	0.538
4	nineteen, forty	0.75	0.75	0.562
5	scholar, job	0.711	0.823	0.585
	scholar, name	0.678	0.778	0.528

the 0.45 threshold. Another very interesting result is that sf_{WP} is always greater than sf_{Fa} when the generalizations are enrichments. Since the former is more related to a specific path, and hence to the goodness of the chosen subsumer, this confirms the previous outcomes (suggesting that the chosen subsumer is close to the generalized concepts). This regularity is lost in Table 4, when *Inverse clustering* was used to build bridges, validating the motivations for which sf_{Fa} is used: the identification of similar concepts when this is not so evident based only on the single subsuming path. Observing the outcome, three aspects can be emphasized: the effectiveness of the searching for bridges in case 5, in which a *three-way bridge* has been built; the overall quality of the generalizations obtained; and the opportunity to perform not only representation shifts, but also an alignment as in case 1 of Table 3. Note that, after enriching the network, we are now able to reason 'by association' also between the *new* paths opened by generalization. We have decided to assign probability 1.0 to each generalization because an oracle, in our case WordNet, has suggested the common subsumer. For example, if we search for a path between *serenity* and *food*, we have to pass through verbs {*find [Pos: 2/2], want [Pos: 3/3]*} for reaching *type* and then fol-

Table 4. Generalizations for pairwise *Inverse clustering*, and corresponding conceptual similarity scores (bottom).

#	Bridge	Subsumer	Subs. Domain	Concepts	Conc. Domain
1	Yes	teaching [100887081]	pedagogy	talk [100893243] lecture [100892861]	pedagogy pedagogy
2	Yes	person [100007846]	biology, person	scientist [110560637] youth [110804406]	person person
3	Yes	music [107020895]	music	introduction [106396930] end [106398401]	literature, music literature, music
4	Yes	figure [113741022]	number	two [113743269] pair [113743605]	number number
5	Yes	person [100007846]	biology, person	viewer [110633450] model [110324851] volunteer [110759151]	person person person
6	No	theory [105888929]	factotum	basis [105793554] theory [105888929]	factotum factotum
7	No	municipality [108626283]	administration, geography, town planning	hometown [108671644] city [108524735]	geography literature, administration, geography, town planning
8	No	territorial dominion [108552138]	administration, town planning	state [108544813] department [108548733]	geography geography
9	No	structure [104341686]	factotum	wall [104546855] level [103365991]	buildings buildings
10	No	representation [104076846]	factotum	scene [106614729] photo [103925226]	photography, racing, sociology, telecommunication photography

#	Pairs	Fa score	WP score	Score
1	talk , lecture	0.739	0.869	0.643
2	scientist, youth	0.724	0.727	0.526
3	introduction, end	0.75	0.714	0.536
4	two, pair	0.72	0.823	0.593
5	viewer, model	0.71	0.666	0.474
	viewer, volunteer	0.71	0.666	0.474
6	basis, theory	0.694	0.842	0.584
7	hometown, city	0.738	0.842	0.621
8	state, department	0.75	0.75	0.562
9	wall, level	0.733	0.8	0.586
10	scene, photo	0.735	0.823	0.605

low the new arc to reach *model*; then, through the verbs {*eat [Pos: 1/4 Neg:3/4], buy [Pos: 2/2]*} we finally reach *food*.

5 Conclusions

This work proposes a technique to automatically extract conceptual graphs from text and reason with them. In particular a way to deal with missing or partial knowledge, noise and to the need of shifting the representation is presented, facing these problems through reasoning 'by association' and the generalization operator. Preliminary experiments confirm the results of previous works and show that this approach can be viable. Further extensions and refinements are needed to improve the effectiveness of this approach, as the definition of an Anaphora Resolution strategy including also handling the Named Entities, or a study on how to set automatically suitable thresholds for searching generalizations, or lastly a way for freeing from WordNet in the choise of the appropiate subsumer.

References

[1] P. Cimiano, A. Hotho, and S. Staab. Learning concept hierarchies from text corpora using formal concept analysis. *J. Artif. Int. Res.*, 24(1):305–339, August 2005.

[2] M.C. de Marneffe, B. MacCartney, and C. D. Manning. Generating typed dependency parses from phrase structure trees. In *LREC*, 2006.

[3] C. Fellbaum, editor. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA, 1998.

[4] S. Ferilli, M. Biba, N. Di Mauro, T.M. Basile, and F. Esposito. Plugging taxonomic similarity in first-order logic horn clauses comparison. In *Emergent Perspectives in Artificial Intelligence*, Lecture Notes in Artificial Intelligence, pages 131–140. Springer, 2009.

[5] S. Ferilli, F. Leuzzi, and F. Rotella. Cooperating techniques for extracting conceptual taxonomies from text. In *Proceedings of The Workshop on Mining Complex Patterns at AI*IA XIIth Conference*, 2011.

[6] R.W. Hamming. Error detecting and error correcting codes. *Bell System Technical Journal*, 29(2):147–160, 1950.

[7] A. Kimmig, V. S. Costa, R. Rocha, B. Demoen, and L. De Raedt. On the efficient execution of problog programs. In Maria Garcia de la Banda and Enrico Pontelli, editors, *ICLP*, volume 5366 of *Lecture Notes in Computer Science*, pages 175–189. Springer, 2008.

[8] D. Klein and C. D. Manning. Fast exact inference with a factored model for natural language parsing. In *Advances in Neural Information Processing Systems*, volume 15. MIT Press, 2003.

[9] A. Maedche and S. Staab. Mining ontologies from text. In *EKAW*, pages 189–202, 2000.

[10] A. Maedche and S. Staab. The text-to-onto ontology learning environment. In *ICCS-2000 - Eight International Conference on Conceptual Structures, Software Demonstration*, 2000.

[11] N. Ogata. A formal ontology discovery from web documents. In *Web Intelligence: Research and Development, First Asia-Pacific Conference (WI 2001)*, number 2198 in Lecture Notes on Artificial Intelligence, pages 514–519. Springer-Verlag, 2001.

[12] A. Cucchiarelli P. Velardi, R. Navigli and F. Neri. Evaluation of OntoLearn, a methodology for automatic population of domain ontologies. In *Ontology Learning from Text: Methods, Applications and Evaluation*. IOS Press, 2006.

[13] L. De Raedt, A. Kimmig, and H. Toivonen. Problog: a probabilistic prolog and its application in link discovery. In *In Proceedings of 20th IJCAI*, pages 2468–2473. AAAI Press, 2007.

[14] T. Sato. A statistical learning method for logic programs with distribution semantics. In *Proceedings of the 12th ICLP 1995*, pages 715–729. MIT Press, 1995.

[15] Z. Wu and M. Palmer. Verbs semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 133–138, Morristown, NJ, USA, 1994. Association for Computational Linguistics.