WiSeNet: Building a Wikipedia-based Semantic Network with Ontologized Relations

Andrea Moro
Dipartimento di Informatica
Sapienza, Università di Roma
Via Salaria, 113
Roma, Italia 00198
moro@di.uniroma1.it

Roberto Navigli
Dipartimento di Informatica
Sapienza, Università di Roma
Via Salaria, 113
Roma, Italia 00198
navigli@di.uniroma1.it

ABSTRACT

In this paper we present an approach for building a Wikipedia-based semantic network by integrating Open Information Extraction with Knowledge Acquisition techniques. Our algorithm extracts relation instances from Wikipedia page bodies and ontologizes them by, first, creating sets of synonymous relational phrases, called *relation synsets*, second, assigning semantic classes to the arguments of these relation synsets and, third, disambiguating the initial relation instances with relation synsets. As a result we obtain WiSeNet, a Wikipedia-based Semantic Network with Wikipedia pages as concepts and labeled, ontologized relations between them.

Categories and Subject Descriptors

I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods – *Semantic networks*; I.2.6 [**Artificial Intelligence**]: Learning – *Knowledge acquisition*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing

General Terms

Algorithms, Experimentation

Keywords

Information Extraction, Knowledge Acquisition, Relation Ontologization, Semantic Network

1. INTRODUCTION

In recent years we have witnessed an increasing popularity and availability of wide-coverage knowledge resources. Such resources, like Wikipedia and Wiktionary, are collaboratively created by exploiting the so-called "wisdom of crowds". As such, they provide a great wealth of semi-structured information in the form of hyperlinked Web pages, which has been shown to be reliable and up-to-date [7]. However, much of this knowledge is only implicitly available in textual form for human consumption and therefore cannot be immediately exploited by machines. This implicit knowledge can be automatically harvested and transformed in machine-readable format by means of automatic tasks such as Information Extraction and Knowledge Acquisition. These tasks aim at enabling one of the long standing goals of Artificial Intelligence, i.e., Machine Reading [16], which is the unsupervised understanding of knowledge extracted from unstructured text.

Information Extraction (IE) is concerned with harvesting relations between entities represented in textual form. Traditional IE techniques focus on extracting relation instances using a fixed set of pre-defined relations [1, 13]. In order to extract relations without pre-defining them, a new IE paradigm, called Open Information Extraction (OIE), has been introduced [3, 18]. The state-of-the-art OIE system is ReVerb [4], which relies only on two simple constraints: i) the lexical aspect of the relational phrase is enforced by means of a manually-defined part-of-speech-based regular expression; ii) an informative relational phrase must appear with several different arguments. However OIE techniques do not provide a formal semantic representation for the arguments and the labels of the harvested relations, which can denote different meanings due to the ambiguous nature of text. As a consequence, redundant relation instances are often produced by OIE systems. For instance, ReVerb extracts the following two synonymous relation instances:

(Natural Language Processing, is a field of, Computer science)
(Natural Language Processing, is an area of, Computer science)

In order to reduce this kind of redundancy, it is possible to cluster synonymous relational phrases [8, 19] and then consider clusters as relations. However, assuming that a relational phrase can have only one meaning limits the number of distinct relational phrases associated with a relation [19]. An alternative solution is that of ontologizing semantic relations [14]. However, the use of WordNet [9] to perform this task makes the ontologization step difficult for many domains, because of the inherent lack of coverage of specialized concepts and named entities.

Knowledge acquisition aims at building large knowledge bases containing semantic relations. Moreover it enables one to overcome the issues of manually-created knowledge bases, like WordNet, which need continuous human maintenance and have very few semantic relations. Recent automatic approaches to building ontologies and semantic networks leverage Wikipedia pages as the main source of semi-structured

CIKM'12, October 29–November 2, 2012, Maui, HI, USA.

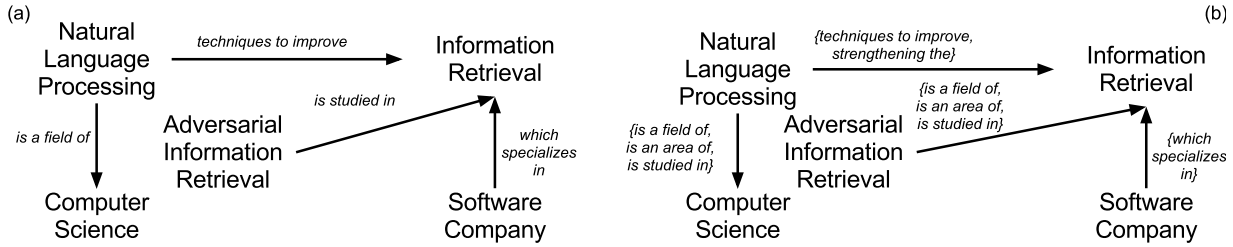

Figure 1: An excerpt of WiSeNet after the first phase (a) and after the second phase (b).

information from which concepts and relations can be extracted [2, 6, 11, 12, 15, 17]. One of the most widespread ontologies is YAGO2 [6], which exploits a set of relation-specific heuristics to extract knowledge from Wikipedia, GeoNames and WordNet. However this ontology considers only about 100 different semantic relations between its concepts. Nastase and Strube presented an automatically created concept network called WikiNet [11] that differs from YAGO2 in that it uses more general heuristics. This resource has around 500 different semantic relations, however, it heavily relies on Wikipedia categories and Infoboxes. As a consequence, WikiNet does not easily scale with the number of different relations. Finally, Babelnet [12], while providing wide coverage of lexicographic and encyclopedic senses, does not provide labels for the relations obtained from Wikipedia.

The approach presented in this paper aims at addressing the above-mentioned issues in OIE, relation ontologization and knowledge acquisition by taking the best of each technique. Large-scale shallow relation extraction is coupled with a Wikipedia category-based semantic representation of the extracted ambiguous relations, which is used to create a full-fledged Wikipedia-based semantic network.

2. A WIKIPEDIA SEMANTIC NETWORK

Our approach consists of two phases, i.e., relation extraction and relation ontologization, which will be illustrated in the following two subsections.

2.1 Relation Extraction

The first phase of our approach consists of extracting relation instances from Wikipedia pages. We use a hyperlink-based heuristic to harvest relational phrases together with their arguments.

DEFINITION 1. *A **relational phrase** is a sequence of words that comprises at least one verb.*

DEFINITION 2. *A **relation instance** is a triple (p_1, ρ, p_2) where p_1, p_2 are Wikipedia pages and ρ is a relational phrase.*

For each Wikipedia page p we mark each occurrence of the title in the body of the page p as a link to p itself. Then, for each sentence s in p, we consider each pair of hyperlinks to the respective Wikipedia pages p_1 and p_2, and if the text between the two links satisfies the definition of relational phrase, we keep the corresponding relation instance *(p_1, relational phrase, p_2)*.

For example from the following excerpt of a Wikipedia page: "[[Natural Language Processing]] is a field of [[computer science]]" we extract the following relation instance: *(Natural Language Processing, is a field of, Computer science)*.

As a result of this extraction phase on the entire Wikipedia dump, we obtain the set $T := \{(p_1, \rho_1, q_1), \ldots, (p_n, \rho_n, q_n)\}$ of all the extracted relation instances. We further denote with P the set of all the relational phrases in T, $P := \{\rho : \exists(p_1, \rho, p_2) \in T\}$.

In contrast to ReVerb our relation extraction step extracts relation instances between Wikipedia pages with a less restrictive constraint on the relational phrase.

2.2 Relation Ontologization

In the second phase we automatically provide explicit semantics for our relation instances, i.e. we ontologize them. Starting from the shallow semantic network obtained in the previous section (Figure 1a), we obtain WiSeNet, an ontologized, Wikipedia-based Semantic Network (Figure 1b).

2.2.1 Clustering of Synonymous Relational Phrases

To cluster synonymous relational phrases in P we build vectors whose components count the occurrences of the most frequent words which occur to the left and to the right of a target relational phrase in a large corpus.

We denote with w_j^l (and w_j^r) the j-th most occurring word on the left (right) of all the extracted relational phrases. Given a relational phrase ρ, following [10], we define the j-th component $l_j(\rho)$ of our **left vector** $\vec{l}(\rho)$ as the conditional probability of w_j^l given ρ divided by the prior probability of w_j^l:

$$l_j(\rho) = \frac{p(w_j^l|\rho)}{p(w_j^l)} = \frac{\text{freq}_{w_j^l,\rho} \times \text{freq}_{\text{total}}}{\text{freq}_\rho \times \text{freq}_{w_j^l}}.$$

We define the **right vector** $\vec{r}(\rho)$ in a similar way, using the respective frequencies w_j^r (the dimension of these vectors is established in the experimental setup, cf. Section 3). Finally we define a measure of similarity between two relational phrases ρ and ρ' by calculating the harmonic mean between the cosine similarity of these vectors:

$$sim(\rho, \rho') = H(cosine(\vec{l}(\rho), \vec{l}(\rho')), cosine(\vec{r}(\rho), \vec{r}(\rho')))$$

where $H(a, b) = \frac{2ab}{a+b}$ is the harmonic mean of a and b. Then for each relational phrase $\rho \in P$ we use this similarity measure to aggregate all the relational phrases that have a similarity with ρ greater than a given threshold θ (see Algorithm 1 for details and Section 3 for the parameter settings). As a result, we obtain a set S of relation synsets for each $\rho \in P$.

DEFINITION 3. *A **relation synset** is a set of synonymous relational phrases.*

For example for the relational phrase *is a field of* we obtain the following relation synset {*is a field of, is an area of, is studied in*}.

Left Semantic Classes	Relation Synset	Right Semantic Classes
Scientific disciplines, Applied sciences, ..., Academic disciplines	is a field of, is an area of, is studied in	Scientific disciplines, Applied Science, ..., Academic disciplines
People, Academics, Students, ..., Education and training occupations	have a BSc in, hold a B.Sc. degree in, possess an undergraduate degree in	Academic disciplines, Science, ..., Scientific disciplines
People, Society, ..., Fictional organizations	assist the, aid the, help the	People, ..., Society

Table 1: Examples of relation synsets and their semantic classes.

Algorithm 1 Building relation synsets

input: P, the set of relational phrases
output: S, the set of relation synsets
function RSS(P)
  $S := \emptyset$
  for each $\rho \in P$ **do**
    $\sigma := \{\rho' \in P : sim(\rho, \rho') \geq \theta\}$
    $S := S \cup \{\sigma\}$
  return S

2.2.2 Semantic Labeling of Relation Synsets

Now that we have our relation synsets, we can introduce semantic classes to describe their arguments. We model semantic classes with Wikipedia categories, which have been shown to provide an adequate semantic representation for several domains [11, 15]. For instance, considering the following relation synset {*is a field of, is an area of, is studied in*} we can use *Subfields by academic discipline* as one of the semantic classes for its left argument and *Scientific Disciplines* as one of the semantic classes for its right argument.

Categories as multisets.

We use multisets to carry out a depth-first-search exploration of the Wikipedia category hierarchy. Given a Wikipedia category, the algorithm recursively searches, up to a fixed depth δ, the category hierarchy and counts the number of times each category is visited. This counting is considered as a relevance ranking of the super-categories of a given category. For instance, given the category *Computational Linguistics* as input and $\delta = 2$, the algorithm outputs: {(Linguistics, 3), (Language, 2), ..., (Computing, 1)}. We named this algorithm `WSC` (for Wikipedia Super Categories).

Categories of a relational phrase.

We next define the categories of the left and right arguments of a relational phrase $\rho \in P$ in the following way:

$$L_\rho = \{c : \exists\, p_1, p_2,\ \ c \in wikiCat(p_1) \wedge (p_1, \rho, p_2) \in T\}$$
$$R_\rho = \{c : \exists\, p_1, p_2,\ \ c \in wikiCat(p_2) \wedge (p_1, \rho, p_2) \in T\}$$

where $wikiCat(p)$ denotes the set of categories of a Wikipedia page p and T is the full set of extracted relation instances (cf. Section 2.1).

Extended categories of a relational phrase.

Next we define $Left_\rho$ and $Right_\rho$ to represent the extended semantics of the left and right arguments of ρ:

$$Left_\rho = \bigcup_{c \in L_\rho} \texttt{WSC}(c), \qquad Right_\rho = \bigcup_{c \in R_\rho} \texttt{WSC}(c).$$

Compared to the sets L_ρ and R_ρ, the extension consists of more varied (i.e. generalized) and consistent multisets of semantic classes for the relational phrase ρ.

Categories of a relation synset.

In order to describe the left and right arguments of a relation synset $\sigma \in S$, we merge the extended category multisets of each relational phrase in the given relation synset σ:

$$Left_\sigma = \bigcup_{\rho \in \sigma} Left_\rho, \qquad Right_\sigma = \bigcup_{\rho \in \sigma} Right_\rho$$

As a result of this step we obtain ontologized relation synsets, i.e. synsets of relational phrases whose arguments are identified by one or more Wikipedia categories. We show some examples of this ontologization step in Table 1.

2.2.3 Disambiguation of Relation Instances

At this point, on one hand we have a large set T of shallow relation instances (cf. Section 2.1), on the other hand we have a wide range of ontologized relation synsets. Our final objective is to use the latter synsets to ontologize the former, possibly ambiguous, relation instances. To do this, for each extracted relation instance $t = (p_1, \rho, p_2) \in T$, we disambiguate ρ with the most suitable relation synset σ, among those which contain ρ. As a result, we obtain a semantically labeled relation instance between two Wikipedia pages.

The algorithm takes as input a set of shallow relation instances T and the set of all the relation synsets S, and outputs a set I of ontologized relation instances.

For each relation instance $t = (p_1, \rho, p_2)$ we define the set of candidate synsets $S_\rho = \{\sigma \in S : \rho \in \sigma\}$. For example, for: t = *(Natural language processing, is a field of, Computer science)*, the set S_ρ contains the following relation synsets:

$S_\rho = \{\{$*is a field of, is cultivated with, where grows*$\}$,
$\{$*is a field of, is an area of, is studied in*$\}$,
$\{$*is a field of, is the battlefield of, was the site of the*$\}\}$.

The core of the algorithm is the computation of the intersections between the argument categories of the relation synset candidates and the Wikipedia categories of pages p_1, p_2. We normalize the cardinality of the intersections to obtain the most suitable relation synset among the candidates. The following function computes the score for each candidate σ:

$$q(p_1, \sigma, p_2) = H\Big(\frac{|C_{p_1} \cap Left_\sigma|}{|Left_\sigma|}, \frac{|C_{p_2} \cap Right_\sigma|}{|Right_\sigma|}\Big),$$

where C_p denotes the extended categories of a Wikipedia page p, $C_p := \bigcup_{c \in wikiCat(p)} \texttt{WSC}(c)$. We use the harmonic mean to guarantee a higher value for those synsets $\sigma \in S_\rho$ that have a large intersection for both the left and right argument categories of σ and the categories of p_1, p_2.

	ReVerb		YAGO2		WikiNet	
Coverage	2.6%	$(176,244/6,737,534)$	2.5%	$(233,602/9,488,985)$	0.2%	$(69,563/28,602,785)$
Extra-Coverage	159.0%	$(10,686,878/6,737,534)$	136.0%	$(13,495,622/9,488,985)$	45.7%	$(13,659,661/28,602,785)$
Novelty	94.4%	$(10,686,878/10,863,122)$	98.3%	$(13,495,622/13,729,224)$	99.5%	$(13,659,661/13,729,224)$

Table 2: Coverage, Extra-coverage, Novelty of our system against ReVerb, YAGO2 and WikiNet.

The relational phrase ρ is disambiguated by selecting the synset that maximizes the function $q(p_1, \sigma, p_2)$ over $\sigma \in S_\rho$. Following the above example, the relation instance t is disambiguated with the second synset, i.e. *(Natural language processing, {is a field of, is an area of, is studied in}, Computer science)*, because the semantic classes of the relation synset and the considered Wikipedia pages share supercategories like *Science* and *Subfields by academic disciplines*, among others, that are not shared with the other candidates.

Recall that from the relation extraction phase (Section 2.1) we obtain a shallow semantic network with Wikipedia pages as nodes and relational phrases between them (Figure 1a). Now, thanks to our relation ontologization process, we can move to a full-fledged Wikipedia-based Semantic Network, that we call WiSeNet. In this network relations have well-defined semantics and edges are explicitly associated with the most suitable relation synset (Figure 1b).

3. EXPERIMENTAL EVALUATION

Relation Extraction: Setup.

We use the 2 July 2012 dump of Wikipedia for our relation extraction phase. To determine the optimal value of the maximum relational phrase length we created a tuning set of 400 sentences containing 783 hyperlinked pairs overall. A judge evaluated each of the extracted relation instances from these sentences and we set the maximum relational phrase length to the value that maximizes the F1-score, that is 16. We extracted $16,344,622$ relation instances with $10,863,122$ distinct relational phrases which we evaluate hereafter.

Relation Extraction: Precision.

Following [4], we performed two manual evaluations regarding the precision of the extracted relational phrases and relation instances. We first built a random sample of $2,000$ distinct relational phrases. A judge was asked to label a relational phrase as correct if the phrase could be used in a sentence with a valid subject and object. For instance, *is a scientific paper by* was marked as correct while *is a scientific paper* was not. We obtained a precision of 79.8%. An error analysis has identified the following main classes of errors: i) phrases containing lists of objects; ii) phrases that do not represent a relation. We then built a random sample of $2,000$ relation instances. The judge was asked to label a relation instance as correct if it makes sense as a sentence. As a result of this evaluation we calculated a precision of 82.8%. An error analysis has identified the following classes of errors (other than those found for relational phrases): i) hyperlinks labeling modifier words, instead of the syntactic head; ii) subject and object are ordered lists.

Relation Extraction: Coverage and Novelty.

In order to study the ability of our relation extraction approach at harvesting fresh relation instances, we calculated the degree of coverage and novelty against well-known existing resources such as ReVerb, YAGO2 and WikiNet. To this end we used the following measures: $Coverage(A,B) = \frac{|A \cap B|}{|B|}$, $Novelty(A,B) = \frac{|A \setminus B|}{|A|}$ and $ExtraCoverage(A,B) = \frac{|A \setminus B|}{|B|}$, where A is either our set of relational phrases P or the Wikipedia page pairs in T, the former for B as ReVerb and the latter for B as YAGO2 or WikiNet, as detailed hereafter.

We ran ReVerb[1] on our Wikipedia dump and we considered only the relation instances output by ReVerb with a confidence score greater than 0.1, selected as a result of our tuning phase. Moreover, as ReVerb does not use Wikipedia pages as arguments of its relation instances, we restricted our comparison to relational phrases. Automatic inspection revealed that our set of relational phrases shares only 2.6% of its elements with ReVerb, while contributing 159.0% new relational phrases that ReVerb did not extract, obtaining a novelty score of 94.4%, as shown in Table 2.

As regards YAGO2[2] and WikiNet[3], we compared only the arguments of the extracted relation instances, as the resources do not share relational phrases. Moreover we filtered out the instances that did not consider Wikipedia pages as their arguments (that might happen with WikiNet when it uses substrings of category names and with YAGO2 when it uses terms from WordNet or GeoNames). Finally we discarded all the self-loops and multiple edges. We cover 2.5% of the pairs extracted by YAGO2, but, on the other hand, we contribute 136.0% new pairs that YAGO2 did not extract, obtaining a novelty score of 98.3%. As for WikiNet, we cover only 0.2% of the pairs, but we contribute 45.7% new pairs that WikiNet did not extract, obtaining a novelty score of 99.5% (see Table 2).

Our evaluation of the extracted relation instances shows that the nature of our relations is complementary to that of alternative resources in the literature.

Relation Ontologization: Setup.

In the second phase, we start from the set P of relational phrases extracted in the first phase. For each relational phrase $\rho \in P$, similarly to [10] we create two $2,000$-dimensional vectors, one for the top $2,000$ words occurring to the left and another one for the words occurring to the right of ρ in a 5-word window (see Section 2.2.1). We estimated such frequencies from a large corpus, that is, Gigaword [5]. We found context words for $314,210$ of our relational phrases. To set up the threshold $\theta = 0.64$, used to build relation synsets (see Algorithm 1), we built a tuning set by manually selecting 100 held-out relational phrases and aggregating them in relation synsets. We then chose the

[1]http://reverb.cs.washington.edu/reverb-latest.jar
[2]http://www.mpi-inf.mpg.de/yago-naga/yago/download/yago2/yago2core_20120109.7z
[3]http://www.h-its.org/downloads/nlp/wikinet.tar.gz

threshold value that maximizes the number of correctly classified relational phrase pairs.

Relation Ontologization: Clustering.

To evaluate the precision of the 39,577 relation synsets that we automatically built, one judge manually evaluated a random sample of 2,000 pairs of relational phrases occurring in a same relation synset. An element was marked as correct if the two relational phrases can be used to suitably represent the same semantic relation. We calculated a precision of 82.1%. An error analysis has identified the following main classes of errors (other than error classes found for relational phrases/instances): (i) relational phrases that negate each other (ii) relational phrases that share the same arguments but are not synonyms.

Relation Ontologization: Semantic Labeling.

In this section we evaluate the semantic classes associated with the arguments of our relation synsets. One judge evaluated a random sample of 2,000 instances composed of three parts: a random relational phrase of a random relation synset and the top-5 semantic classes extracted for its left and right arguments. A correct assignment of semantic classes to the relation synset arguments was marked as correct if the classes correctly describe a subset of the concepts that the relational phrase can assume on its left and right. We obtained a precision of 68.7%. Note that this is a particularly difficult task, as it involves the quality of both the relational phrases (used for building relation synsets) and relation instances (used to assign the semantic classes to the relation synsets). We show some of the evaluated instances in Table 1.

Relation Ontologization: Disambiguation.

Finally we evaluated our disambiguation procedure of our relation instances, as described in Section 2.2.3. A judge evaluated a random sample of 2,000 disambiguated relation instances (p_1, σ, p_2), where, instead of the whole relation synset σ, a randomly-chosen $\rho \in \sigma$ was presented to the annotator such that $(p_1, \rho, p_2) \notin T$ (this can happen as the relational phrases in σ can relate different pairs of Wikipedia pages in T). We required the judge to mark an element as correct if p_1 ρ p_2 makes sense as a sentence. In this way we calculated a precision of 76.7%. This evaluation indicates that, despite the difficulty of the disambiguation task and the degree of ambiguity of relational phrases (almost 5), the initial precision of our relational phrases, i.e. 82.8%, decreases by around 6% when moving from shallow to ontologized relation instances. Moreover this is an evaluation of the whole system as it takes into account all the previous steps and assesses novel relation instances that were not extracted from the first step.

4. CONCLUSIONS AND FUTURE WORK

We presented an automatic approach to the construction of a full-fledged semantic network by combining Open Information Extraction with knowledge acquisition techniques. Our algorithm extracts relation instances from Wikipedia pages and ontologizes them by, first, creating relation synsets, second, assigning semantic classes to the arguments of these synsets and, third, disambiguating the initial relation instances with the most suitable relation synsets.

To our knowledge, this is the first time that large-scale information extraction and relation ontologization are integrated to produce a full-fledged semantic network with Wikipedia pages as concepts and labeled, ontologized relations between them. Our evaluation shows that our resource, WiSeNet, is complementary in nature and content with existing wide-coverage resources like YAGO2 and WikiNet.

As future work, we aim at exploiting the syntactic structure of sentences to further improve the precision of our approach. The shallow semantic network, as well as WiSeNet, are available at http://lcl.uniroma1.it/wisenet.

Acknowledgments

The authors gratefully acknowledge the support of the ERC Starting Grant MultiJEDI No. 259234.

References

[1] M. Banko and O. Etzioni. The Tradeoffs Between Open and Traditional Relation Extraction. In *Proc. of ACL/HLT*, pages 28–36, 2008.
[2] Gerard de Melo and Gerhard Weikum. Menta: inducing multilingual taxonomies from wikipedia. In *Proc. of CIKM*, pages 1099–1108, 2010.
[3] O. Etzioni, M. Banko, S. Soderland, and D. S. Weld. Open information extraction from the web. *Comm. of the ACM*, 51(12):68–74, 2008.
[4] A. Fader, S. Soderland, and O. Etzioni. Identifying Relations for Open Information Extraction. In *Proc. of EMNLP*, pages 1535–1545, 2011.
[5] D. Graff and C. Cieri. English Gigaword. LDC 2003.
[6] J. Hoffart, F. M. Suchanek, K. Berberich, and G. Weikum. YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. *Journal of Artif. Intell.*, 2012.
[7] M. Hu, E. Lim, A. Sun, H. W. Lauw, and B. Vuong. Measuring article quality in wikipedia: models and evaluation. In *Proc. of CIKM*, pages 243–252, 2007.
[8] S. Kok and P. Domingos. Extracting Semantic Networks from Text Via Relational Clustering. In *Proc. of ECML/PKDD*, pages 624–639, 2008.
[9] G. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. Introduction to WordNet: An On-Line Lexical Database. *Int. J. of Lexicography*, 3(4):235–244, 1990.
[10] J. Mitchell and M. Lapata. Composition in Distributional Models of Semantics. *J. of Cog. Sc.*, 34(8):1388–1429, 2010.
[11] V. Nastase and M. Strube. Transforming Wikipedia into a large scale multilingual concept network. *Journal of Artif. Intell.*, 2012.
[12] R. Navigli and S. P. Ponzetto. BabelNet: the Automatic Construction, Evaluation and Application of a Wide-Coverage Multilingual Semantic Network. *Journal of Artif. Intell.*, 2012.
[13] F. Peng and A. McCallum. Information extraction from research papers using conditional random fields. *Journal of Inf. Process. Manage.*, 42(4):963–979, 2006.
[14] M. Pennacchiotti and P. Pantel. Ontologizing semantic relations. In *Proc. of COLING-ACL*, pages 793–800, 2006.
[15] S. P. Ponzetto and M. Strube. Taxonomy induction based on a collaboratively built knowledge repository. *Journal of Artif. Intell.*, 175(9-10):1737–1756, 2011.
[16] H. Poon and et al. Machine Reading at the University of Washington. In *Proc. of NAACL-HLT*, pages 87–95, 2010.
[17] S. Szumlanski and F. Gomez. Automatically acquiring a semantic network of related concepts. In *Proc. of CIKM*, pages 19–28, 2010.
[18] F. Wu and D. S. Weld. Open Information Extraction Using Wikipedia. In *Proc. of ACL*, pages 118–127, 2010.
[19] A. Yates and O. Etzioni. Unsupervised methods for determining object and relation synonyms on the web. *JAIR*, 34 (1):255–296, 2009.