Evaluating PageRank Methods for Structural Sense Ranking in Labeled Tree Data

Andrea Tagarelli
Dept. Electronics, Computer and Systems Sciences
University of Calabria, Italy
tagarelli@deis.unical.it

Francesco Gullo
Yahoo! Research
Barcelona, Spain
gullo@yahoo-inc.com

ABSTRACT

Link analysis methods like the popular PageRank are increasingly being applied to lexical knowledge bases to deal with a number of natural language processing problems, including unsupervised word sense ranking and disambiguation. Compared to plain-text, the topic of sense ranking in semistructured data has been however studied marginally.

This paper aims to bridge PageRank-based word sense ranking and tree-structured data. We propose PageRank-style methods for the structural sense ranking problem, which take into account tree structural relations as well as semantic relatedness in the constituents of tree data. The proposed methods are comparatively evaluated with existing PageRank methods for word sense disambiguation. Effectiveness and efficiency of PageRank methods have been assessed on various data with different domain vocabularies.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information filtering*; I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods—*Semantic Networks*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Text analysis*

General Terms

Algorithms, Experimentation, Performance

Keywords

Tree-structured data, word sense disambiguation, ranking, semantic relatedness, WordNet

1. INTRODUCTION

Labeled (rooted) trees are traditionally used as a convenient data model to enable the representation and description of real-life data objects and their structural relationships. Typically, data management and mining tasks require the format of such semistructured data is well-defined and flexible, and designed to be self-describing. Within this view, XML is well-known as a standard for the representation of labeled tree data, thanks to its ability to provide an extensible means of associating descriptive markup to semistructured data. However, in the effort of supporting the development of interoperable domain-specific lexicons, semistructured/Web data is heterogeneous by nature [23, 12]; this is clearly expected since a meta-language like XML supports the specification of a machine-readable grammar, but the grammar semantics and the semantic relations underlying the data constituents cannot be formally specified. As a consequence, different markup tags may be used to describe the same concept, and different concepts may be described using the same tags. Effectively coupling syntactic with semantic information in labeled tree data like XML is hence required to next-generation methods that are designed to identify related semantics in syntactically different data, or to discriminate among different semantics in apparently similar data syntaxes. This way, such methods will enable emerging knowledge-based applications such as, e.g., mapping and integrating conceptually related information in tree-shaped data structures, determining affinities in heterogeneous Web service descriptions, organizing semantically related documents, devising prototypes for different semantic views over document collections.

Similarly to the case of plain-text, handling the semantics in labeled tree data raises a lexical ambiguity problem, which can be expressed as: how to detect and assess semantic relationships among the concepts underlying the constituents of structural information. Word sense disambiguation (WSD) is the process of associating a given word in a text or discourse context with a semantic definition, or sense, which is distinguishable from other potential senses of that word [18]. WSD is hence essential to address the inherent ambiguity of the meanings of lexical constituents of natural language texts, which is testified by a large corpus of studies coming from different research communities.

However, regardless of the specific approach adopted to perform WSD, existing models and techniques are traditionally conceived to deal with structure-free texts, while facing WSD and sense ranking problems for semistructured data is challenging due to diverse types of structural information at different refinement levels (e.g., element/attribute labels, edges, paths, twigs) that can be used to explain such data. Focusing on tree-shaped data, a key challenge is hence how to represent the structural characteristics by coupling the syntactic hierarchical information with the semantic meanings that are associated to descriptive markup tags. To the

WIMS '12, June 13-15, 2012 Craiova, Romania

best of our knowledge, there is a relatively smaller corpus of studies that has taken somehow into account semantic aspects in XML retrieval, data management or mining tasks, including schema similarity and matching [21, 24, 9, 11, 3], feature extraction [26], classification [27], and clustering [25, 19]. Only a few of the aforementioned studies go beyond the computation of semantic similarity to determine a ranking/disambiguation of the appropriate meanings to be contextually associated to XML elements. An early study in the context of document classification [27] combines texts and structure information to generate XML structure and content features, while ontological knowledge is marginally used. [16, 26] address the problem of unsupervised structural sense disambiguation in semistructured data, which utilizes WordNet as ontological knowledge. [16] is a versatile approach as it can in principle be applied to element names in XML data as well as in Web directories, however the use of WordNet in [16] is limited to synonymies and *is-a* relations; by contrast, the approach in [26] also involves *part-of* relations and is designed to use the WordNet noun hierarchies concurrently (it indeed defined different notions of semantic relatedness and strategies of search through the WordNet hierarchies).

In recent years, there has been a growing interest in *link analysis eigenvector-based ranking*, and particularly the well-known Google's PageRank, for natural language processing applications [17, 1, 2, 7]. The underlying assumption is that in a cohesive text, related lexical concepts (word meanings) tend to occur together and form a semantic network that can be used to build a discourse understanding model. As discussed in [17], PageRank-style methods on lexical semantic networks intuitively implement the concepts of *text cohesion* and *relevance of word meanings* in a text. Effectiveness of such methods in ranking the meanings of all words in a text is ensured by the PageRank global ranking scheme: high-ranked meanings can be seen as "recommendations" by related meanings, where preferred recommendations are made by most influential meanings (which in turn are highly recommended by other related meanings). Starting from the study in [17], PageRank methods have indeed shown to improve effectiveness of knowledge-based WSD methods.

Contributions and scope of this work.

This work brings for the first time a link analysis eigenvector-based approach in the context of sense ranking for labeled tree data. We define a framework of PageRank-style methods that apply to a WordNet-based semantic network extracted from an instance of labeled tree data, with the objective of ranking the senses associated with the constituents (i.e., tags) of tree-structured data. In this framework, we focus on development of weighted formulations of PageRank as specifically conceived for taking into consideration the implicit order and structural relations of the input data. In this respect, we propose to weight the PageRank score based on the semantic relatedness between neighbor concepts of any given tag, and further refine this basic idea to extend the local context of ranking of a given concept based on its *prestige*, which is defined in terms of *support*, *influence*, or both.

For comparative evaluation, the framework also includes existing PageRank methods for WSD. We devise an experimental evaluation methodology that addresses the issues of crisp sense disambiguation and probabilistic sense ranking, and evaluate our framework on various labeled trees belonging to different application domains. Results have shown the potential of the various formulations of PageRank for structural sense ranking and have shed light on their different performances in terms of effectiveness and efficiency.

We would like to point out that PageRank approaches to sense disambiguation and ranking have received relatively less attention than other existing unsupervised knowledge-based approaches as well as supervised corpus-based approaches. Within this view, this work aims to fill a lack of knowledge on the applicability of PageRank methods to semantic networks, and particularly extends it from plain text (already studied in, e.g., [17, 1, 2, 7]) to tree-structured text data.

It should be noted that *this work is not aimed to define a new or better approach for WSD or sense ranking*, and hence a comparison with existing state-of-the-art unsupervised knowledge-based methods or supervised corpus-based methods [18] is beyond the scope of this work. Moreover, the proposed structural sense ranking framework *is not supposed to be dependent on a specific (meta)language for semistructured or Semantic Web data* (RDF/OWL formats), rather it is concerned with the simplest data model traditionally used in semistructured data management. Although in this study only WordNet-based semantic relatedness measures have been considered in our evaluation framework, the applicability of PageRank methods *is not limited to a particular semantic relatedness measure* or *to a particular lexical knowledge base* (i.e., other knowledge sources can definitely be used alternatively or in combination with WordNet).

The rest of the paper is organized as follows. Section 2 provides background notions on lexical knowledge, semantic relatedness measures, and the PageRank algorithm. The section also discusses related work, focusing on existing PageRank methods for WSD. Section 3 describes our PageRank-based sense ranking framework, and provides formal details about the construction of context graphs and the ranking methods. Section 4 presents experimental methodology and results. Section 5 concludes the paper.

2. BACKGROUND

2.1 WSD and lexical ontologies

An essential component in tasks of semantic analysis in text data is represented by knowledge sources, which span from word corpora to ontologies [18]. The particular type of knowledge source and its features are indeed central to develop the approach used for the task at hand. In WSD and sense ranking, there are two distinguished approaches, namely corpus-based and dictionary/knowledge-based. The corpus-based approach is data-driven, since involves information about the contexts of previously disambiguated words. Most methods which fall into this category usually require (semi)supervised learning from sense-tagged corpora to enable predictions on new words; therefore, these methods might rely on manually annotated corpora, which are laborious and expensive to create. Dictionary-based methods are instead knowledge-driven, since they are concerned with the textual descriptions of word definitions available from a dictionary or similar external knowledge resource (e.g., sense inventory) as a source of information about the word meanings. In this respect, *WordNet* [10] is a publicly available large-scale lexical ontology, which has been widely employed in several tasks of natural language processing. In WordNet, related concepts are grouped into equivalence classes, called *synsets* (sets of synonyms). Each synset represents one underlying lexical concept, or *sense*, and is explained by a short text, called *gloss*. A polysemous term belongs to multiple synsets, thus it is typically associated with a linearly ordered set of senses. Synsets are explicitly connected to each other in the form of ontologies through different relations, e.g., *is-a* relations (hypernymy/hyponymy) and *part-of* relations (meronymy/holonymy).

In dictionary-based WSD a major assumption is that the choice of the most plausible sense to assign to each word in an input text, or in general the ranking of its senses, is based on the *relatedness* among the selected senses. Relatedness is typically determined by means of word semantic measures, which are briefly discussed next.

2.2 Semantic relatedness measures

Semantic relatedness measures have been successfully applied to a variety of natural language problems; particularly, the overwhelming attention attracted from dictionary-based WSD methods in the past years has been testified by a growth of semantic relatedness measures [6, 29]. Dictionary-based semantic relatedness measures can be divided in two broad categories depending on whether they use either explicitly modeled knowledge (i.e., relations in a semantic graph structure) or information available from the concept descriptions. We will briefly overview some of the most representative measures as they will be used in our experimental evaluation.

Gloss-based measures focus on the content affinity of the glosses which are regarded as concept descriptions. According to the Lesk method [15], the word overlap between two concepts' glosses determines the relatedness of concepts—the larger the overlap, the higher the relatedness. Gloss overlaps are proven to be an effective way to find even implicit relationships between concepts, as the shared content words may hint at their relatedness. However, glosses are by definition very short texts and may hence not provide enough information about the concepts' descriptions. Banerjee and Pedersen [4] extended the basic gloss overlap notion and proposed a gloss overlap scoring function that has the merit of considering phrasal matches and weighting them more heavily than single word matches:

$$go\text{-}rel(c_1, c_2) = \sum_{go \in GO(g_1, g_2)} |go|^2 \qquad (1)$$

where g_1, g_2 denote the glosses of concepts c_1, c_2, respectively, $GO(g_1, g_2)$ denotes the set of disjoint, maximal word-sequences shared between g_1 and g_2 (overlaps), and $|go|$ indicates the number of words in the overlap *go*. In [20, 4], a notion of extended gloss overlap was also proposed to consider the glosses of concepts that are directly connected to a given concept by a certain relation.

Path-based measures are defined as functions of the location of nodes representing concepts in a semantic graph structure (e.g., the lexical ontology provided by WordNet) [6, 29]. The shorter the path connecting two concept nodes, the higher the relatedness between the two concepts. Simple path length methods however discard the taxonomic assumption that a shallower concept node corresponds to a more general concept. More refined measures also take into account the *specificity* of concepts as well as the *commonality* between concepts. The specificity of concepts is measured by their depth in the reference hierarchy, whereas the commonality of two concepts is captured by the depth of their *least common subsumer* (lcs), i.e., the most specific concept that two input concepts share on their paths to the root of the hierarchy [4, 6]. This allows for weighting more the relatedness between less general concepts compared to more abstract concepts, their path lengths being equal. All such notions are encompassed by the Wu & Palmer measure [29], which is formally defined for any two concepts c_1, c_2 as:

$$p\text{-}rel(c_1, c_2) = \frac{2 \times depth(lcs(c_1, c_2))}{depth(c_1) + depth(c_2)} \qquad (2)$$

Information-content-based measures exploit the intuition that the similarity of two concepts can be determined by the amount of information they share. Following the standard information theory, this amount of information is expressed by the *information content* of a concept: $IC(c) = -\log \Pr(c)$. $\Pr(c)$ is the probability of encountering an instance of concept c in the reference corpus, and is estimated by the relative frequency of usage of that concept in the corpus:

$$\Pr(c) = \frac{\sum_{w \in W(c)} count(w)}{N}$$

where $W(c)$ denotes the set of words (noun tokens) whose senses are subsumed by c and N is the total number of concept words in the corpus. This definition of IC relies on the availability of a word corpus; however, existing lexical ontologies like WordNet embed statistics about the usage of concepts. Moreover, since $\Pr(c)$ is monotonic when moving upward a hierarchy, for each pair of concepts c_1, c_2 where c_1 is subsumed by c_2 it holds that $\Pr(c_1) \leq \Pr(c_2)$. Analogously to the case of path-based measures, the notion of least common subsumer also applies to capture the information-content-based relatedness between concepts. The Lin measure represents a universal measure of information-content-based relatedness [6].

$$ic\text{-}rel(c_1, c_2) = \frac{2 \times IC(lcs(c_1, c_2)}{IC(c_1) + IC(c_2)} \qquad (3)$$

Note that Lin measure is the equivalent of Wu & Palmer measure in the information content approach, and that both are related to the well-known Dice similarity coefficient.

2.3 PageRank

PageRank is the renowned global ranking page scheme developed by Brin and Page [5], which essentially extends the basic citation idea by considering a notion of "importance" of the pages that point to a given page. As the definition of PageRank is recursive, the importance of a page both relies on and influences the importance of other (neighbouring) pages, based on a Markov chain model. Given a page p_i, its PageRank on a network of size N is computed as

$$rank(p_i) = \frac{1-d}{N} + d \sum_{p_j \in B(p_i)} \frac{rank(p_j)}{out(p_j)} \qquad (4)$$

where $B(p)$ denotes the set of pages that point to p (i.e., pages that can be reached through backward links of p), and $out(p)$ denotes the number of forward (outgoing) links of p. The damping factor d ($0 < d < 1$, usually set to 0.85 [5, 17]) implements the so-called random-surfer model, i.e., the random surfer is expected to discontinue the chain with probability $1-d$, and hence to randomly select a page each with relevance $1/N$. The PageRank vector computed for all pages is the dominant eigenvector of the probability transition matrix of this random walk [5].

2.4 Related work on PageRank for WSD

In recent years, PageRank has been applied to semantic networks inferred from natural language texts for WSD purposes. An early attempt is proposed in [17], where traditional PageRank is applied on a graph built over WordNet synsets (vertices) and edges are drawn using synset relations available in WordNet. Two approaches are also defined to refine the basic PageRank method or the ranking it computed: the first approach consists in using the Lesk algorithm to provide PageRank with a initial ranking of nodes, while the second approach is to combine the ranking obtained by PageRank with WordNet sense frequency information. According to the best performance results obtained on SemCor and Senseval-2, PageRank outperformed Lesk, while combining the two methods however did not bring any significant improvement over the individual methods performance when sense order is taken into account.

A different graph context for PageRank-based WSD is devised in [1]. The underlying idea is to extract an undirected subgraph of WordNet which links the synsets of words in the input text, and then again apply the basic PageRank over the subgraph. This subgraph is obtained by the union of sets of shortest paths, each set having a different synset as a source. An early proposal of personalized PageRank for WSD is proposed in [2], where the personalization vector is initialized with the synsets of the words in the input text. This personalized PageRank method utilizes the full (undirected) WordNet graph, where synsets are connected by WordNet relations, with the addition of a directed subgraph whose vertices are the input words and edges are links to the synset vertices of the WordNet graph. Inspired by the topic-sensitive PageRank approach [13], the initial probability mass is concentrated uniformly over the word vertices, which act as source nodes injecting mass into the corresponding synset vertices and spread their mass over the WordNet graph. Moreover, mutual reinforcement effect between semantically related synsets of the same word is alleviated by a variant called W2W. For each input word, W2W aims to concentrate the initial probability mass only over the synsets of the words surrounding that input word. Clearly, W2W is less efficient than personalized PageRank since it needs as many runs as the number of the input words.

To improve the efficiency in personalized PageRank, [7] proposes to exploit Latent Semantic Analysis (LSA) in order to integrate latent semantic relations between words in the initialization of the personalization vector. The input text is lexically expanded by including those words in the vocabulary that have a cosine similarity with the term frequency - inverse document frequency representation of the text in the LSA space above a certain threshold.

3. PAGERANK-BASED STRUCTURAL SENSE RANKING FRAMEWORK

A lexical ontology like WordNet can be seen as a graph $G_{WN} = (V_{WN}, E_{WN})$ whose vertices V_{WN} are concepts (represented by synsets) and edges E_{WN} correspond to semantic relations between concepts (i.e., *is-a* relations, *part-of* relations). Let $\mathcal{D}$ denote a labeled tree data instance (e.g., an XML document or portion of it) rooted in a node labeled as t_0, and let $\mathcal{T}(\mathcal{D}) = \{t_0, t_1, \ldots, t_n\}$ (for short, $\mathcal{T}$) be the set of document element labels (i.e., tag names) of $\mathcal{D}$. For each tag $t \in \mathcal{T}$, the set of senses of t available in WordNet is denoted as $\mathcal{S}(t)$. In the following, we will refer to senses and concepts interchangeably.

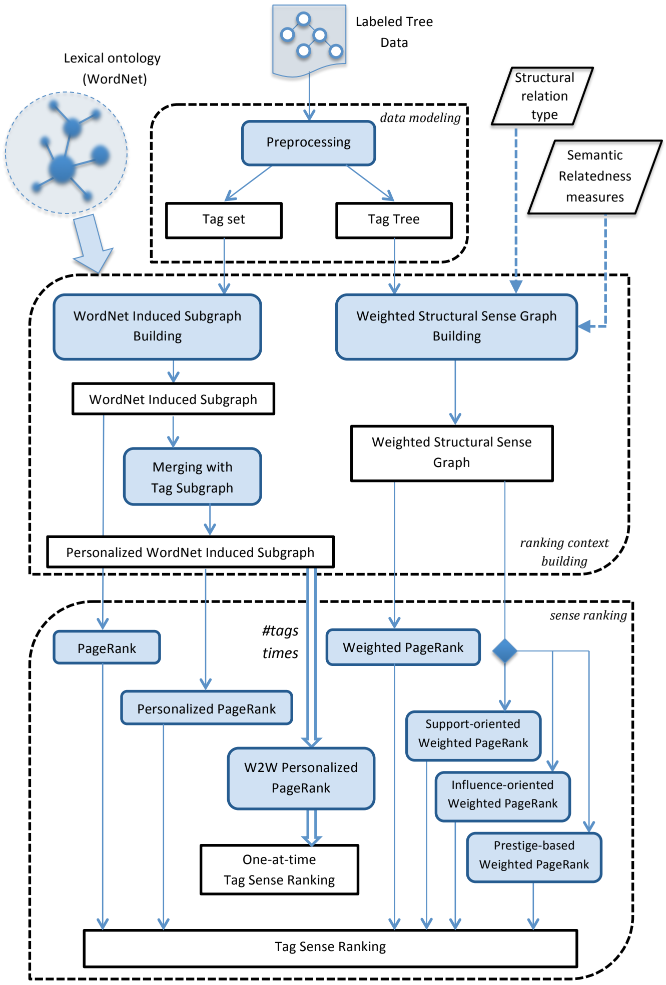
Labeled Tree Data
Lexical ontology (WordNet)
Structural relation type
Semantic Relatedness measures
data modeling
Preprocessing
Tag set
Tag Tree
WordNet Induced Subgraph Building
Weighted Structural Sense Graph Building
WordNet Induced Subgraph
Merging with Tag Subgraph
Weighted Structural Sense Graph
Personalized WordNet Induced Subgraph
ranking context building
sense ranking
PageRank
#tags times
Weighted PageRank
Personalized PageRank
Support-oriented Weighted PageRank
W2W Personalized PageRank
Influence-oriented Weighted PageRank
Prestige-based Weighted PageRank
One-at-time Tag Sense Ranking
Tag Sense Ranking

Figure 1: Overview of the proposed PageRank-based structural sense ranking framework

Our general goal for PageRank-based sense ranking in labeled tree data is as follows: Given a labeled tree data instance $\mathcal{D}$ and the availability of a lexical ontology like WordNet, compute a ranking of all concepts associated with each tag name in $\mathcal{D}$ via a PageRank-style method that is applied on a semantic network built over the tag concepts.

Figure 1 shows main modules and data flows of our PageRank-based sense ranking framework for tree-structured data. The *data modeling* module performs the step of preprocessing of the input data and yields its representation as a tree of tags or a set of tags. The *ranking context building* module is in charge of constructing a semantic graph, called *context graph*, that serves as a context for the subsequent step; such a graph can have different structural features depending on which representation of the input data is used and on to what extent the ontological knowledge and other semantic tools (i.e., relatedness measures) are involved. The *sense ranking* module executes one of the PageRank algorithms devised to finally produce a ranking of all concepts associated with each input tag name. Note that all algorithms are carried out once for each input data, except W2W Personalized PageRank which is run as many times as the number of input tags. In the following we present in detail our definitions of the second and third module in the framework.

3.1 Building the context graph

WordNet induced subgraph.

One way of building the context graph is to induce it from the WordNet graph in order to include the concepts of the input tags together with a minimal set of other concepts that serve to connect the target concepts. According to [1], the context graph can be derived by the union of the subgraphs corresponding to the shortest paths that connect all pairs of concepts of the input tag names. Formally, an undirected graph $G_I = (V_I, E_I)$ is extracted from G_{WN} such that

$$G_I = \bigcup_{t_i \in \mathcal{T}} \{shpath(c, c') \mid c \in \mathcal{S}(t_i), c' \in \mathcal{S}(t_j), t_j \neq t_i\} \quad (5)$$

where function $shpath(c, c')$ is defined to extract the subgraph corresponding to the shortest path connecting c and c'; in case of multiple shortest paths, the first one is chosen according to the linear ordering of synsets.

Personalized WordNet induced subgraph.

A more complex context graph can be obtained by the union of the WordNet induced subgraph G_I and a new graph G_T, whose vertices are the input tag names with their concepts and edges are directed links from tag vertices to concept vertices. Formally, $G_T = (V_T, E_T)$ such that $V_T = \mathcal{T} \cup \{c \mid c \in \mathcal{S}(t), t \in \mathcal{T}\}$ and $E_T = \bigcup_{t \in \mathcal{T}}(t, c),\ c \in \mathcal{S}(t)$. The resulting graph, denoted as $G_H = (V_H, E_H)$, is hence hybrid as it is comprised of a directed subgraph (G_T) and an undirected subgraph (G_I). This particular form is devised to support a personalization of PageRank, since the vertices of G_T act as source nodes for the rank propagation based on the personalization vector [2].

Weighted structural sense graph.

Induced subgraphs from the WordNet graph usually contain many concept vertices that may not be strictly relevant to the input tag names but only serve a connectivity purpose. To reduce the size of a context graph, we adopt a general methodology described as follows:

- Consider all concepts of the input tag names as vertices of the context graph; highly abstract concepts (i.e., primitive synsets in WordNet [10]) can also be added as auxiliary source vertices.

- Draw an edge between any two concepts associated to different tag names depending on a selected type of *structural relation* that may hold between the two tag names according to the tree-structured representation of the input data. Multiple occurrences of tags in different positions of the tree are not distinguished here, and this point is left as a future work.

- Compute a weight on each edge to express the *semantic relatedness* between the connected concepts. Semantic relatedness can be computed by using one of the measures discussed in Section 2.2, or other existing measures (e.g., [6, 29, 26]).

According the above methodology, the weighted structural sense graph is defined as $G_W = (V_W, E_W, w)$, where:

- $V_W = \{c \mid c \in \mathcal{S}(t), t \in \mathcal{T}\} \cup \{c \mid c \in \widehat{\mathcal{S}}(t_0), t_0 \in \mathcal{T}\}$, where $\widehat{\mathcal{S}}(t_0)$ denotes the set of primitive synsets (indirectly) connected to t_0's synsets through the *is-a* relation.

- $\widetilde{E}_W = \{(c, c') \mid c \in \mathcal{S}(t_i), c' \in \mathcal{S}(t_j), t_i, t_j \in \mathcal{T} \wedge sTree(t_i, t_j)\} \cup \{(c, c') \mid c \in \widehat{\mathcal{S}}(t_0), c' \in \mathcal{S}(t_0), t_0 \in \mathcal{T}\}$. Function *sTree* applies to a pair of tag names t_i, t_j and returns a boolean value depending on whether one of the following structural relations holds:

  - *childOf*(t_i, t_j): it holds if t_j is a child of t_i;
  - *descOf*(t_i, t_j): it holds if t_j is a descendant of t_i;
  - *desc-sibdescOf*(t_i, t_j): it holds if t_j is a descendant of t_i or a descendant of a t_i's sibling;
  - *any*(t_i, t_j): it always evaluates to true except when $t_i = t_j$ (i.e., concepts of the same tag are not connected in order to avoid undesired mutual reinforcement effects).

  Note that $\widetilde{E}_W$ is a set of directed edges, except for the case *sTree* = *any* (we can view a pair of directed edges as an undirected edge).

- $w : \widetilde{E}_W \xrightarrow{SR} \Re^*$ is a weighting function that computes a non-negative real-valued relatedness between any two connected concepts according to one of the functions in the set $SR = \{$*go-rel*(), *p-rel*(), *ic-rel*()$\}$.

- $E_W \subseteq \widetilde{E}_W$ such that $E_W = \{e = (c, c') \mid e \in \widetilde{E}_W \wedge w(e) > 0\}$.

Figure 2 illustrates a concise representation of structural sense graph in two cases, *sTree* = *descOf* and *sTree* = *desc-sibdescOf*, for an example labeled tree.

3.2 PageRank methods

The sense ranking module in Figure 1 involves a number of PageRank methods, which are listed next:

- PageRank method, which is applied to the WordNet induced subgraph [1].

- Personalized PageRank method, which is applied to either the personalized WordNet induced subgraph or its word-to-word variant [2].

- Weighted PageRank methods, which are applied to any of the variants of weighted structural sense graph. In the following we present our formulations of Weighted PageRank. The proposed methods differ from each other in the way a weight is introduced in the ranking of vertex and how neighbouring vertices are involved in the weight definition of a given vertex.

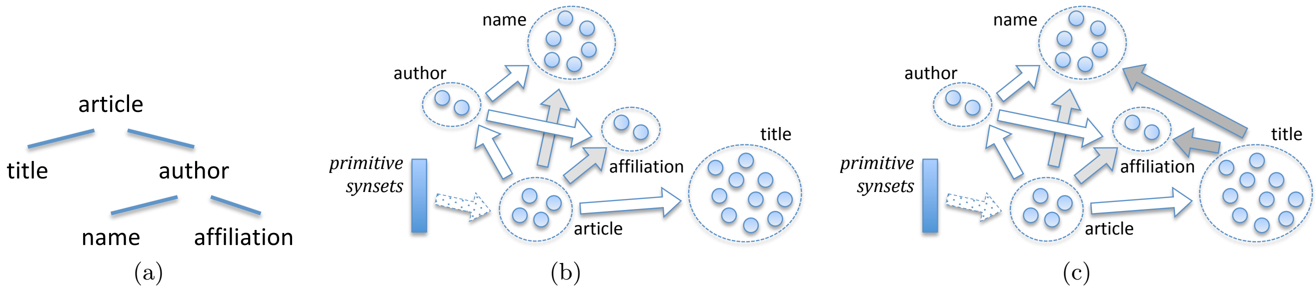

Figure 2: Structural sense graph: (a) an example labeled tree, and the corresponding context graphs with (b) *sTree* = *descOf* and (c) *sTree* = *desc-sibdescOf*

To avoid cluttering for clear presentation: concepts (synsets) of the same tag are grouped together (dashed circles), any arrow from a synset group to another represents as many directed edges as the product of the sizes of the two groups, darker arrows correspond to more indirect structural relations, edge weights are not shown.

Weighted PageRank.

Our definition of weighted PageRank is an adaptation of the basic PageRank that introduces a weighting factor in the rank of each of the vertices that point to a given vertex. Concepts are ranked proportionally to their semantic relatedness with respect to a given concept vertex they point to. Given a weighted structural sense graph $G_W = (V_W, E_W, w)$, for each concept vertex $c \in V_W$ the rank is computed as:

$$rank(c) = \frac{1-d}{|V_W|} + d \sum_{c_b \in B(c)} \frac{rank(c_b)\ w(c_b, c)}{\sum_{c_{b_r} \in R(c_b)} w(c_b, c_{b_r})} \quad (6)$$

where $R(c)$ denotes the set of reference vertices of c (i.e., the set of vertices that are pointed by c).

Prestige-based Weighted PageRank.

We also propose a more refined approach to the definition of weighted PageRank methods by involving a notion of *prestige* [28]. In directed network analysis, the prestige of a vertex can be seen as proportional to the number or relevance of its incoming and outgoing links. More precisely, incoming links contribute to determine the *support* of a vertex, whereas outgoing links contribute to determine the *influence* of a vertex. In our context, a concept will have high support if many, semantically related concepts refer to it; moreover, a concept will have high influence if it points to many other concepts. Note that such a general notion of prestige encompassing both influence- and support-oriented aspects is best suited to explain lexical cohesion in text: given a concept, relevant incoming and outgoing links are likely to be drawn for concepts that are semantically related to the current concept.

Given a weighted structural sense graph $G_W = (V_W, E_W, w)$, for each concept vertex $c \in V_W$, we provide the following definitions of prestige-based weighted PageRank:

Support-oriented Weighted PageRank:

$$rank(c) = \frac{1-d}{|V_W|} + d \sum_{c_b \in B(c)} rank(c_b) sup(c_b, c) \quad (7)$$

Influence-oriented Weighted PageRank:

$$rank(c) = \frac{1-d}{|V_W|} + d \sum_{c_b \in B(c)} rank(c_b) inf(c_b, c) \quad (8)$$

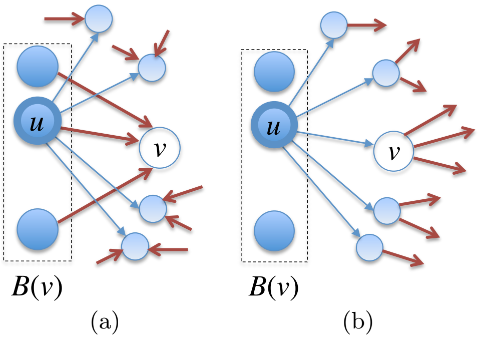

Figure 3: Prestige-based ranking of a concept-vertex in structural sense graph: (a) support- and (b) influence-oriented weights (thicker arrows correspond to the edges that contribute to the v's support or influence)

Fully Prestige-based Weighted PageRank:

$$rank(c) = \frac{1-d}{|V_W|} + d \sum_{c_b \in B(c)} rank(c_b)(sup(c_b, c) + inf(c_b, c)) \quad (9)$$

with

$$sup(u, v) = \frac{\sum_{v_b \in B(v)} w(v_b, v)}{\sum_{u_r \in R(u)} \sum_{u_{r_b} \in B(u_r)} w(u_{r_b}, u_r)} \quad (10)$$

$$inf(u, v) = \frac{\sum_{v_r \in R(v)} w(v, v_r)}{\sum_{u_r \in R(u)} \sum_{u_{r_r} \in R(u_r)} w(u_r, u_{r_r})} \quad (11)$$

Equation 10 defines the contribution a concept u gives to the support of a concept v. Such a contribution is intuitively explained as the ratio of the sum of weights on incoming links of v to the sum of weights on incoming links of vertices that are pointed by u. Analogously, Equation 11 defines the contribution a concept u gives to the influence of a concept v, which is the ratio of the sum of weights on outgoing links of v to the sum of weights on outgoing links of vertices that are pointed by u. Figure 3 illustrates such a situation.

4. EXPERIMENTAL EVALUATION

We devised an experimental evaluation of our PageRank methods to comparatively assess their effectiveness and effi-

Table 1: Labeled trees (XML documents) used in the experiments

data	*tag set size*	*average polysemy*	*average polysemy w/o monosemous tags*	*max polysemy*
Auction	22	4.18	4.34	12
Genealogy	20	4.90	5.88	16
Geography	36	3.11	4.17	8
IEEE	27	4.74	6.05	13
Lesson	19	3.52	3.82	8
Medicine	20	4.30	4.88	10
Music	31	6.45	6.45	12
Order	15	4.87	5.46	15
People	19	3.74	4.71	9
Recipe	9	4.78	5.86	11
Shakespeare	9	6.33	7.86	17
Wikipedia	14	6.36	6.77	13

ciency in structural sense ranking. In the following we first describe data and methodology used for the evaluation, then we present our main experimental results.[1]

4.1 Data and assessment methodology

Following the lead of previous works dealing with WSD in semistructured/XML data (e.g., [16, 26]), we selected different application domains, from which individual document instances were built up. Table 1 reports on statistics about tag set and polysemy for each document.[2] A short description of the data used in our experiments is provided next.

Auction contains typical information used in a process of buying and selling items, including item details, payment and shipping types, bids and bidders. *Genealogy* represents different types of genealogical information. *Geography* models geographical information about countries (morphology, hydrography, demography, religions, languages, etc.). *IEEE* represents IEEE journal structures concerning computer science literature. *Lesson* is an example of data that can be exchanged by e-learning services (classes, topics, lesson calendar, enrolled students). *Medicine* represents information about diseases (conditions, symptoms, medications, etc.) and medical procedures. *Music* refers to a revised partwise file for publishing scores in musical applications using the MusicXML format.[3] *Order* models orders, customers and market segments. *People* contains personal records, gathering individuals' profile and demographic information. *Recipe* is used to represent recipes in a cookbook. *Shakespeare* refers to a logical-structure-oriented, simplified version of the schema of the Shakespeare 2.00 collection,[4] a publicly available example of prose literature coded in XML. *Wikipedia* represents encyclopaedia articles with a Wikipedia-like structure.

As a remark on the impact of the selected structural relation (function *sTree*) on the size of the weighted structural sense graphs built on the various test data, we observed the following average, minimum and maximum increments in the number of edges that were drawn: 2.15, 1.70, 3.60 from *childOf* to *descOf*, 2.61, 1.40, 4.10 from *descOf* to *desc-sibdescOf*, and 2.33, 1.50, 3.20 from *desc-sibdescOf* to *any*.

Tag names are usually subject to a number of text processing operations in real applications (e.g., word splitting, normalization, etc.); however, for purposes of evaluation of this work we collected documents that utilize only terms (i.e., single- or multi-word tags) that match dictionary entries in WordNet. Moreover, although our proposed methods are not constrained to a particular part-of-speech, in this work we focused on noun tags, since they are much more heavily used to annotate semistructured data than verbs, adverbs, or adjectives. Note also that to enable the gloss-based semantic relatedness measure (Eq. 1), we processed the text of synset glosses by performing removal of stopwords and word stemming (based on Porter's algorithm[5]). We used WordNet version 3.0 for all experiments in this work.

We built up a human-based sense-annotation of the evaluation data, by asking 50 persons with higher-education to examine the WordNet senses for each tag name and rank them by expressing their confidence as probability values. As a result, we obtained a probabilistic ranking of the senses of each tag and for each document, as the average over 50 manually provided rankings. Note that the selected annotators had no prior experience with WordNet and no knowledge about how our methods work, so that their evaluation was not affected by any bias relating to our experiment goals.

To assess the effectiveness of the proposed methods, we performed both the conventional *accuracy* evaluation based on the top-ranked sense for each tag (i.e., proportion of correctly disambiguated tags) as well as an *information-theoretic* evaluation for the whole tag sense rankings. The latter evaluation is particularly appealing for probabilistic sense ranking tasks, and it is motivated by a common issue that arises in WSD tasks, namely the controversy about the very notion of sense: dictionaries may provide sense distinctions that are too fine or too coarse for the data at hand, and it is quite common that multiple fine-grained senses may be correct for a given noun, thus it is hard for a human annotator to decide exactly for a single sense and regard it as the appropriate one. A major consequence of this inherent uncertainty is that a reference disambiguation may be excessively biased by a crisp manual selection, and as such may result inadequate for a fair evaluation. We therefore resorted to the *cross-entropy* measure which is widely used to evaluate how well a model assigns probabilities to its predictions [22]. In our context, this maps to the calculation of the cross-entropy of two probability distributions, the first representing the human's assignment and the second representing the algorithm's assignment of the probability that each sense might be the correct one. Given a tag name t with m senses, the cross-entropy is formally defined as:

$$H(P_t^*, P_t) = -\sum_{i=1}^{m} P_t^*(i) \ \log P_t(i) \qquad (12)$$

where P_t^* (resp. P_t) represents the probabilities over the t's senses provided by a human annotator (resp. by an algorithm). For any given document, cross-entropy is computed as the average of the cross-entropies over all tags in that document. We will use symbols A and CE to denote overall accuracy (in percent) and overall cross-entropy, respectively. Note that a good word sense disambiguation/ranking should have high accuracy and low entropy.

[1] Experiments were carried out on a Mac OS X platform with 3.06GHz, 8 GB memory.

[2] All evaluation data is available at http://uweb.deis.unical.it/tagarelli/software/ssr/

[3] http://www.recordare.com/xml.html

[4] http://metalab.unc.edu/bosak/xml/eg/shaks200.zip

[5] http://www.tartarus.org/∼martin/PorterStemmer/.

Table 2: Accuracy and cross-entropy results of baseline, PageRank and personalized PageRank methods

data	UFrequencyRank		PR		PR_isa		PPR		PPR_isa		W2W-PPR		W2W-PPR_isa	
	A	*CE*	*A*	*CE*	*A*	*CE*	*A*	*CE*	*A*	*CE*	*A*	*CE*	*A*	*CE*
Auction	50.00	0.24	40.91	0.74	31.82	0.83	45.46	0.66	**50.00**	0.71	45.46	0.66	**50.00**	0.71
Genealogy	70.00	0.20	**60.00**	0.76	50.00	0.85	50.00	0.66	40.00	0.74	50.00	0.66	40.00	0.75
Geography	75.00	0.17	72.22	0.64	50.00	0.73	**77.78**	0.53	61.11	0.70	**77.78**	0.53	61.11	0.70
IEEE	62.96	0.19	51.85	0.68	44.44	0.75	**62.96**	0.61	59.26	0.70	**62.96**	0.61	59.26	0.70
Lesson	68.42	0.24	36.84	0.67	36.84	0.71	**52.63**	0.51	47.37	0.57	**52.63**	0.51	47.37	0.58
Medicine	65.00	0.34	45.00	0.73	50.00	0.74	**55.00**	0.61	**55.00**	0.60	**55.00**	0.62	**55.00**	0.60
Music	35.48	0.25	**38.71**	0.74	25.81	0.80	32.26	0.71	29.03	0.74	32.26	0.71	29.03	0.74
Order	66.67	0.24	33.33	0.74	26.67	0.78	**46.67**	0.62	**46.67**	0.66	**46.67**	0.62	**46.67**	0.65
People	73.68	0.24	65.16	0.66	57.89	0.72	**68.42**	0.49	63.16	0.61	**68.42**	0.50	63.16	0.61
Recipe	44.44	0.47	33.33	0.33	33.33	0.34	66.67	0.61	66.67	0.59	**77.78**	0.61	66.67	0.59
Shakespeare	44.44	0.36	33.33	0.70	33.33	0.67	**44.44**	0.54	33.33	0.47	**44.44**	0.54	33.33	0.47
Wikipedia	50.00	0.25	35.71	0.70	**42.86**	0.72	**42.86**	0.57	35.71	0.62	**42.86**	0.57	35.71	0.62
total average	*55.56*	*0.27*	*45.53*	*0.67*	*40.25*	*0.72*	*53.76*	*0.59*	*48.94*	*0.64*	*54.69*	*0.59*	*48.94*	*0.64*

Bold values refer to best accuracy results obtained on each test data (baseline results excluded).

4.2 Results

The ranking methods used in our experiments are denoted as follows:

- **non-weighted methods**: PR (for PageRank), PPR (for Personalized PageRank), W2W-PPR (for word-to-word Personalized PageRank). We also denote with PR_isa, PPR_isa, and W2W-PPR_isa variants of the algorithms in which only the *is-a* relation is used to find (shortest) paths between tag concepts.
- **weighted methods**: WPR (for Weighted PageRank), S-WPR (for Support-oriented Weighted PageRank), I-WPR (for Influence-oriented Weighted PageRank), P-WPR (for Fully Prestige-based Weighted PageRank).

As a best baseline, we also performed a completely supervised annotation method, which ranked the senses of any given tag by decreasing usage frequency (i.e., the most frequent sense is ranked as first, and so on). We denote this method with UFrequencyRank.

4.2.1 Effectiveness

Table 2 compares results obtained by PageRank and personalized PageRank methods, including their *is-a* variants. We observed that the performance of PR was lower than PPR and W2W-PPR in most cases; particularly, while PR accuracy was lower on all data but *Genealogy* and *Music*, PR cross-entropy showed to be higher (hence, worse) than at least one of the other PageRank methods on all data. Overall, PPR and W2W-PPR improved over PR in terms of both accuracy (resp., +8 and +9) and cross-entropy (both -0.08). Focusing on the two best methods, W2W-PPR and PPR performed very closely to each other, with a slightly increased accuracy (on average +1) obtained by W2W-PPR but roughly identical average cross-entropy. Concerning the *is-a* variants, a degradation of performance of all methods occurred in most cases: on average, -5 A and +0.05 CE for PR, -5 A and +0.05 CE for PPR, -6 A and +0.05 CE for W2W-PPR; note also that the *is-a* restriction led to roughly identical performances of PPR and W2W-PPR. This suggests that using more semantic relationships besides *is-a* can be actually useful to improve effectiveness of sense disambiguation and ranking.

Table 2 includes results obtained by UFrequencyRank. With the exception of *Recipe*, UFrequencyRank always outperformed non-weighted PageRank methods in terms of cross-entropy (overall average improvement above 0.32) and also achieved better accuracy in six of twelve data. Note that the superiority exhibited by UFrequencyRank is clearly due to an advantage that supervised approaches have against fully unsupervised methods like non-weighted PageRank methods.

Weighted PageRank methods are compared in Table 3. Prestige-based methods generally outperformed WPR, reaching better accuracy and cross-entropy in terms of both average and best scores over *sTree* and relatedness measures per data. Over all data, P-WPR and S-WPR improved over WPR up to +6.91 A, -0.14 CE and +1.18 A, -0.02 CE, respectively, whereas I-WPR obtained a decrement of +1.65 A and an improvement of -0.04 CE. Combining support- and influence-oriented weights in P-WPR revealed to be useful to maximize accuracy performance, with a few exceptions: S-WPR was winner on *Genealogy*, whereas I-WPR prevailed on *Auction* and *IEEE*. However, as cross-entropy results are considered, the fully prestige-based weighted method produced sense rankings that are closest to the human-based reference ones, often achieving best cross-entropy below 0.15.

As concerns the semantic relatedness measures used by the weighted methods, the following remarks were drawn (results not shown due to space limits of this paper). By averaging over all data, algorithms and structural relation types, *ic-rel* and *p-rel* led generally to better accuracy than *go-rel*: 44.88 for *go-rel* (best on 4 data), 45.41 for *ic-rel* (best on 6 data), and 46.91 for *p-rel* (best on 7 data); different behaviors were exhibited in terms of average cross-entropy: 0.42 for *go-rel* (best on 9 data), 0.46 for *ic-rel* (best on 2 data), and 0.45 for *p-rel* (best on 2 data).

Focusing on the impact of the structural relation type (*sTree*) for the construction of context graphs on the PageRank performance, we observed that relations more complex than parent-child (*childOf*) produced beneficial effects in most cases, although in some cases there was no (significant) difference, or even better results were produced by *childOf* (on *IEEE* and *Medicine*). A summary of average scores over all data, algorithms and relatedness measures, is as follows: 45.75 A and 0.45 CE for *childOf*, 44.97 A and 0.45 CE for *descOf*, 43.76 A and 0.43 CE for *desc-sibdescOf*, 48.47 A and 0.44 CE for *any*. Nevertheless, we tend to believe that a relative conceptual homogeneity of the tags in the selected structural context would justify *desc-sibdescOf* or *any* as better choices than less complex contexts. In other

Table 3: Accuracy and cross-entropy results of weighted PageRank methods: average (over semantic relatedness measures) and best results by varying the type of tree structural relation

data	*sTree*	WPR				S-WPR				I-WPR				P-WPR			
		avg		*best*		*avg*		*best*		*avg*		*best*		*avg*		*best*	
		A	*CE*	*A*	*CE*	*A*	*CE*	*A*	*CE*	*A*	*CE*	*A*	*CE*	*A*	*CE*	*A*	*CE*
Auction	*childOf*	39.40	0.46	40.91	0.44	45.45	0.44	50.00	0.44	37.88	0.51	40.91	0.50	43.94	0.46	50.00	0.44
	descOf	42.42	0.48	45.45	0.46	40.91	0.51	45.45	0.50	37.88	0.51	40.91	0.50	43.94	0.52	50.00	0.50
	desc-sibdescOf	45.45	0.43	45.45	0.40	45.45	0.36	54.55	0.32	46.97	0.41	**59.09**	0.41	40.91	0.25	54.55	0.15
	any	40.91	0.52	40.91	0.50	45.45	0.45	54.55	0.40	45.45	0.45	54.55	0.40	42.42	0.33	50.00	0.28
	average	*42.05*	*0.47*	*43.18*	*0.45*	*44.32*	*0.44*	*51.14*	*0.42*	*42.05*	*0.47*	*48.87*	*0.45*	*42.80*	*0.39*	*51.14*	*0.34*
Genealogy	*childOf*	46.67	0.43	50.00	0.42	51.66	0.41	55.00	0.40	40.00	0.46	45.00	0.45	46.67	0.41	50.00	0.40
	descOf	55.00	0.42	60.00	0.42	50.00	0.43	55.00	0.40	38.33	0.46	40.00	0.45	43.33	0.44	65.00	0.40
	desc-sibdescOf	48.33	0.42	50.00	0.40	53.33	0.44	65.00	0.43	38.33	0.50	40.00	0.41	45.00	0.48	65.00	0.40
	any	41.66	0.40	55.00	0.40	43.33	0.47	55.00	0.41	43.33	0.47	55.00	0.41	46.67	0.36	**75.00**	0.12
	average	*47.92*	*0.42*	*53.75*	*0.41*	*49.58*	*0.44*	*57.50*	*0.41*	*39.99*	*0.47*	*45.00*	*0.43*	*45.42*	*0.42*	*63.75*	*0.33*
Geography	*childOf*	52.78	0.35	55.56	0.27	54.63	0.31	55.56	0.22	52.77	0.27	55.56	0.23	51.85	0.15	55.56	0.15
	descOf	55.56	0.38	58.33	0.33	57.41	0.37	58.33	0.33	52.77	0.31	55.56	0.29	56.48	0.13	66.67	0.13
	desc-sibdescOf	51.85	0.42	52.78	0.40	52.77	0.38	55.56	0.32	61.11	0.35	63.89	0.29	62.04	0.29	**80.56**	0.10
	any	63.89	0.40	75.00	0.32	62.04	0.39	66.67	0.29	62.04	0.39	66.67	0.29	63.89	0.29	**80.56**	0.06
	average	*56.02*	*0.39*	*60.42*	*0.33*	*56.71*	*0.36*	*59.03*	*0.29*	*57.17*	*0.33*	*60.42*	*0.27*	*58.57*	*0.21*	*70.84*	*0.11*
IEEE	*childOf*	48.15	0.38	55.56	0.36	46.91	0.43	48.15	0.40	60.49	0.42	**70.37**	0.40	60.49	0.40	62.96	0.36
	descOf	48.15	0.39	59.26	0.37	43.21	0.47	44.44	0.45	60.49	0.42	66.67	0.39	59.26	0.42	62.96	0.35
	desc-sibdescOf	44.44	0.44	48.15	0.43	41.98	0.44	48.15	0.41	51.86	0.39	59.26	0.37	54.32	0.38	62.96	0.36
	any	49.38	0.43	59.26	0.40	55.56	0.42	59.26	0.40	55.56	0.42	59.26	0.40	56.79	0.30	59.26	0.13
	average	*47.53*	*0.41*	*55.56*	*0.39*	*46.92*	*0.44*	*50.00*	*0.42*	*57.10*	*0.41*	*63.89*	*0.39*	*57.72*	*0.38*	*62.04*	*0.30*
Lesson	*childOf*	52.63	0.52	57.89	0.51	52.63	0.51	57.89	0.49	43.86	0.38	47.37	0.36	50.78	0.25	57.89	0.25
	descOf	52.63	0.51	57.89	0.51	50.88	0.49	52.63	0.43	42.11	0.35	42.11	0.34	49.12	0.21	57.89	0.14
	desc-sibdescOf	54.39	0.48	63.16	0.46	50.88	0.45	52.63	0.37	40.35	0.28	47.37	0.28	54.38	0.12	63.16	0.11
	any	57.89	0.44	**68.42**	0.39	52.63	0.48	63.16	0.43	52.63	0.48	63.16	0.43	52.63	0.37	63.16	0.13
	average	*54.39*	*0.49*	*61.84*	*0.47*	*51.76*	*0.48*	*56.58*	*0.43*	*44.74*	*0.37*	*50.00*	*0.35*	*51.73*	*0.24*	*60.53*	*0.16*
Medicine	*childOf*	56.67	0.56	70.00	0.53	60.00	0.56	**75.00**	0.55	36.67	0.56	40.00	0.56	56.67	0.56	70.00	0.55
	descOf	50.00	0.58	65.00	0.54	55.00	0.58	65.00	0.55	38.33	0.56	40.00	0.56	55.00	0.47	60.00	0.33
	desc-sibdescOf	38.33	0.49	45.00	0.44	40.00	0.55	50.00	0.51	38.33	0.56	50.00	0.55	36.67	0.48	55.00	0.39
	any	40.00	0.54	45.00	0.50	28.33	0.44	40.00	0.39	28.33	0.44	40.00	0.39	31.67	0.39	45.00	0.31
	average	*46.25*	*0.54*	*56.25*	*0.50*	*45.83*	*0.53*	*57.50*	*0.50*	*35.42*	*0.53*	*42.50*	*0.52*	*45.00*	*0.48*	*57.50*	*0.40*
Music	*childOf*	33.33	0.54	35.48	0.54	37.63	0.50	38.71	0.45	34.41	0.65	38.71	0.64	36.56	0.42	45.16	0.19
	descOf	25.81	0.55	29.03	0.54	37.63	0.56	45.16	0.52	33.33	0.41	35.48	0.40	40.86	0.48	54.84	0.37
	desc-sibdescOf	25.81	0.56	29.03	0.56	35.48	0.56	41.94	0.48	46.24	0.30	51.61	0.24	50.54	0.23	58.06	0.09
	any	43.01	0.53	54.84	0.45	47.31	0.47	**61.29**	0.37	47.31	0.47	**61.29**	0.37	47.31	0.36	**61.29**	0.08
	average	*31.99*	*0.55*	*37.10*	*0.52*	*39.51*	*0.52*	*46.78*	*0.46*	*40.32*	*0.46*	*46.77*	*0.41*	*43.82*	*0.37*	*54.84*	*0.18*
Order	*childOf*	51.11	0.46	53.33	0.46	48.89	0.47	53.33	0.47	46.67	0.49	46.67	0.47	48.89	0.48	53.33	0.47
	descOf	51.11	0.46	53.33	0.46	53.33	0.46	60.00	0.46	46.67	0.48	46.67	0.47	51.11	0.48	53.33	0.46
	desc-sibdescOf	48.89	0.46	53.33	0.46	48.89	0.45	53.33	0.41	37.78	0.47	40.00	0.42	37.78	0.45	46.66	0.36
	any	53.33	0.41	60.00	0.36	40.00	0.39	53.33	0.35	40.00	0.39	53.33	0.35	48.89	0.15	73.33	0.14
	average	*51.11*	*0.45*	*55.00*	*0.44*	*47.78*	*0.44*	*55.00*	*0.42*	*42.78*	*0.46*	*46.67*	*0.43*	*46.67*	*0.39*	*56.66*	*0.36*
People	*childOf*	56.14	0.47	63.16	0.45	57.89	0.45	63.16	0.41	54.39	0.47	57.89	0.46	56.14	0.42	63.16	0.40
	descOf	45.61	0.46	47.37	0.46	52.63	0.45	57.89	0.44	56.14	0.48	57.89	0.47	52.63	0.44	57.89	0.41
	desc-sibdescOf	50.88	0.48	63.16	0.48	54.39	0.48	63.16	0.46	54.39	0.49	57.89	0.48	56.14	0.47	68.42	0.45
	any	63.16	0.44	68.42	0.39	61.40	0.41	**73.68**	0.38	61.40	0.41	**73.68**	0.38	61.40	0.31	**73.68**	0.58
	average	*53.95*	*0.46*	*60.53*	*0.45*	*56.58*	*0.45*	*64.47*	*0.42*	*56.58*	*0.46*	*61.84*	*0.45*	*56.58*	*0.41*	*65.79*	*0.46*
Recipe	*childOf*	55.56	0.53	66.67	0.51	55.56	0.52	66.67	0.48	55.56	0.51	66.67	0.50	51.85	0.51	**88.89**	0.38
	descOf	55.56	0.52	66.67	0.49	55.56	0.52	66.67	0.50	51.85	0.51	55.56	0.50	51.85	0.51	66.67	0.49
	desc-sibdescOf	44.44	0.47	55.56	0.45	44.44	0.48	55.56	0.46	48.15	0.48	66.67	0.45	44.44	0.42	66.67	0.40
	any	48.15	0.52	66.67	0.49	48.15	0.46	55.56	0.42	44.44	0.46	55.56	0.42	44.44	0.44	66.67	0.36
	average	*50.93*	*0.51*	*63.89*	*0.49*	*50.93*	*0.49*	*61.12*	*0.47*	*50.00*	*0.49*	*61.12*	*0.47*	*48.15*	*0.47*	*72.23*	*0.38*
Shakespeare	*childOf*	55.56	0.47	**66.67**	0.39	55.56	0.42	**66.67**	0.34	48.15	0.45	55.56	0.41	51.85	0.41	**66.67**	0.33
	descOf	48.15	0.48	**66.67**	0.40	55.56	0.44	**66.67**	0.36	44.44	0.46	55.56	0.43	51.85	0.42	55.56	0.37
	desc-sibdescOf	51.85	0.47	**66.67**	0.42	48.15	0.46	55.56	0.45	44.44	0.47	55.56	0.45	51.85	0.44	**66.67**	0.44
	any	55.56	0.52	**66.67**	0.47	62.96	0.49	**66.67**	0.42	62.96	0.49	**66.67**	0.42	62.96	0.47	**66.67**	0.41
	average	*52.78*	*0.49*	*66.67*	*0.42*	*55.56*	*0.45*	*63.89*	*0.39*	*50.00*	*0.47*	*58.34*	*0.43*	*54.63*	*0.44*	*63.89*	*0.39*
Wikipedia	*childOf*	40.47	0.52	50.00	0.50	35.71	0.52	42.86	0.46	30.95	0.37	35.71	0.36	40.48	0.15	50.00	0.13
	descOf	30.95	0.54	35.71	0.48	30.95	0.54	35.71	0.47	35.71	0.37	42.86	0.36	45.24	0.27	57.14	0.13
	desc-sibdescOf	23.81	0.56	28.57	0.51	28.57	0.57	42.86	0.52	42.86	0.35	57.14	0.33	47.62	0.34	**64.29**	0.33
	any	40.48	0.56	50.00	0.46	47.62	0.52	**64.29**	0.43	47.62	0.52	**64.29**	0.43	47.62	0.43	**64.29**	0.15
	average	*33.93*	*0.55*	*41.07*	*0.49*	*35.71*	*0.54*	*46.43*	*0.47*	*39.29*	*0.40*	*50.00*	*0.37*	*45.24*	*0.30*	*58.93*	*0.19*
total average		*47.40*	*0.48*	*54.60*	*0.45*	*48.43*	*0.47*	*55.79*	*0.42*	*46.29*	*0.44*	*52.95*	*0.41*	*49.55*	*0.37*	*61.46*	*0.30*

Bold values refer to best accuracy results obtained on each test data.

terms, for descriptive markup covering a larger variety of topics, building the context graph over (directly) related tags (as in the case *childOf* or *descOf*) would reduce the disambiguation "noise" which might be produced by more complex structural contexts.

Comparing results from Table 2 and Table 3, there is evidence of an improved performance in terms of both accuracy and cross-entropy obtained by any weighted PageRank method over non-weighted PageRank methods. Over all data, the following gains in terms of average best cross-entropy were in fact achieved with respect to W2W-PPR: 0.14 by WPR, 0.17 by S-WPR, 0.18 by I-WPR, and 0.29

by P-WPR; moreover, in terms of total average best accuracy, P-WPR and S-WPR improved over W2W-PPR by +6.8 and +1.1, respectively. Note also that weighted PageRank methods obtained the same accuracy as UFrequencyRank on *Lesson* and *People*, whereas they outperformed it on the remaining ten data, with an average improvement above 15 and peaks of 44.45 on *Recipe*, 25.81 on *Music*, 22.23 on *Shakespeare*. In terms of cross-entropy, absolute best results were mostly obtained by P-WPR (e.g., 0.17 on *Music*, 0.11 on *Geography*, 0.10 on *Wikipedia* and *Order*).

4.2.2 *Efficiency*

Figure 4 shows time ratios for building the context graphs for WPR, with respect to the most complex relation type (i.e., *any*). We observed that building a *desc-sibdescOf*-based graph required 45% on average (up to 67%) of the time spent for building a *any*-based graph. Instead, the *childOf* and *descOf* relations allowed significant time savings, since they required 9 and 19%, respectively, of the *any*-based building time.

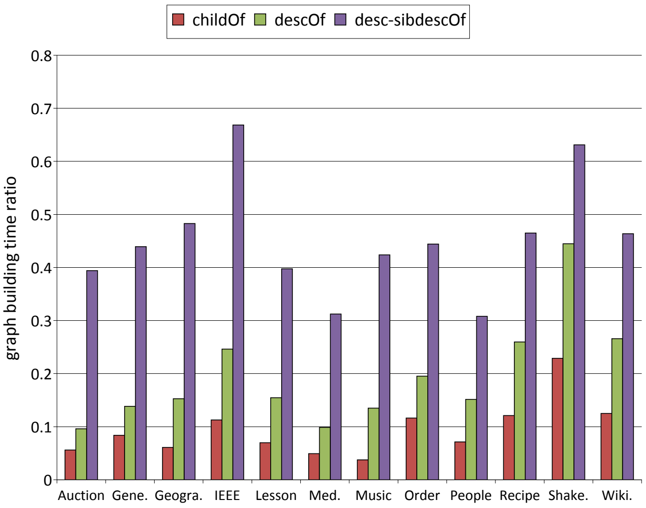

Figure 4: Building the structural sense graphs: time proportions by varying the structural relation type

We also analyzed the times for building non-weighted context graphs, i.e., WordNet induced subgraphs and their personalized versions; in this respect, however, there were no significant differences in practice, with an average gain/loss ratio of 1.5%. More importantly, we compared the processing times of weighted and non-weighted graphs (results not shown) in order to assess whether more time was generally spent for building a context graph that has a larger set of vertices but no weights on its edges, or for a context graph that has less nodes but weighted edges. For this evaluation, the observed cross-comparison between building times of WordNet induced subgraph and those of any *sTree* variant of weighted structural sense graph was consistently in favor of the weighted graphs on all documents: in fact, the time spent for building a WordNet induced subgraph was always much higher, on average 16, 58, 330, and 415 times more than building a *any*, *desc-sibdescOf*, *descOf*, and *childOf*-based graph, respectively.

Table 4 and Table 5 report time performances of the various PageRank methods in terms of number of iterations and percentage of the ranking algorithm runtime over the total runtime (i.e., context graph building time plus ranking time). Note that no maximum number of iterations was predefined, so that we let each algorithm run until a convergence tolerance of 1.0E-08 was satisfied; moreover, again for weighted methods, results were averaged over the semantic relatedness measures. Looking at non-weighted methods in Table 4, relative percentage times were usually lower than 0.5%. On average, convergence was faster for PR (0.06% time and 36 iterations) than for PPR (0.1% time and 43 iterations), and for a single-run of W2W-PPR (0.16% time and 41 iterations). A more variegated situation occurred for weighted methods (Table 5), whose performance clearly depends on the choice of structural relation used to construct the context graph. No significant difference was instead observed by varying the type of semantic relatedness measure. WPR usually required a few more iterations than the prestige-based methods: on average, 42 iterations for WPR, against 26, 40, and 25 iterations respectively for S-WPR, I-WPR, and P-WPR. By contrast, the relative percentage time spent by prestige-based methods was 21 to 33% higher than WPR, over all data. Generally, the relative percentage time of a weighted PageRank tended to get higher as the graph size (number of edges) gets larger; similarly, but much less regularly, the number of iterations required for convergence could increase with the complexity of context graph.

Table 4: Efficiency of PageRank and personalized PageRank methods: proportion of the algorithm runtime (in percent) and number of iterations

data	PR		PPR		W2W-PPR	
	% time	*#iter.*	*% time*	*#iter.*	*% time**	*#iter.**
Auction	0.03	35	0.17	43	0.07	41
Genealogy	0.01	35	0.02	44	0.02	42
Geography	0.07	34	0.13	40	0.27	37
IEEE	0.01	35	0.01	42	0.10	37
Lesson	0.19	36	0.23	43	0.47	40
Medicine	0.05	36	0.08	42	0.14	40
Music	0.02	35	0.04	41	0.09	39
Order	0.09	35	0.14	43	0.16	42
People	0.01	35	0.01	42	0.02	40
Recipe	0.22	38	0.33	45	0.43	44
Shakespeare	0.03	37	0.07	45	0.08	43
Wikipedia	0.01	36	0.01	42	0.02	41

* W2W-PPR times and iterations refer to averages over the multiple runs of the algorithm that were carried out.

5. CONCLUSION

We studied the applicability of PageRank-style methods to WSD and ranking problems in the context of labeled tree data. We evaluated existing formulations of PageRank to tree-structured data, and proposed various formulations of weighted PageRank specifically for structural sense ranking. By integrating the evidence of effectiveness results with efficiency results, we found that the proposed weighted PageRank methods should be preferred to the basic and personalized PageRank methods: in fact, the weighted methods turned out to be more effective in terms of both accuracy and cross-entropy and faster than non-weighted methods. This indicates that the performance of unsupervised knowledge-driven sense ranking based on PageRank can be

Table 5: Efficiency of weighted PageRank methods: proportion of the algorithm runtime (in percent) and number of iterations

data	*sTree*	WPR		S-WPR		I-WPR		P-WPR	
		% time	*#iter.*	*% time*	*#iter.*	*% time*	*#iter.*	*% time*	*#iter.*
Auction	*childOf*	1.51	3	7.99	3	6.15	2	4.40	3
	descOf	5.84	3	29.66	3	23.31	2	22.83	3
	desc-sibdescOf	55.07	88	87.58	35	92.00	54	62.80	21
	any	65.79	64	92.76	49	92.62	49	79.25	17
	average	*32.05*	*40*	*54.50*	*23*	*53.52*	*27*	*42.32*	*11*
Genealogy	*childOf*	9.82	67	49.76	54	45.67	54	12.48	19
	descOf	24.00	67	69.05	54	67.78	54	37.66	19
	desc-sibdescOf	16.86	95	18.13	17	36.95	46	27.28	25
	any	22.72	59	57.85	48	57.36	48	34.55	17
	average	*18.35*	*72*	*48.70*	*43*	*51.94*	*51*	*27.99*	*20*
Geography	*childOf*	21.72	32	61.06	24	83.89	50	45.75	55
	descOf	34.73	28	72.99	15	90.85	43	87.82	84
	desc-sibdescOf	50.79	91	86.54	55	86.27	56	79.25	62
	any	69.14	56	91.61	49	91.29	49	77.94	17
	average	*44.09*	*52*	*78.05*	*36*	*88.07*	*50*	*72.69*	*55*
IEEE	*childOf*	26.82	78	52.61	23	78.95	56	45.24	60
	descOf	45.23	78	60.63	21	84.10	55	73.05	51
	desc-sibdescOf	44.26	94	73.63	49	73.00	49	49.09	17
	any	58.64	63	83.80	50	83.06	50	65.52	18
	average	*43.74*	*78*	*67.67*	*36*	*79.78*	*53*	*58.23*	*37*
Lesson	*childOf*	2.27	7	25.76	6	70.48	49	49.66	17
	descOf	39.99	29	73.83	14	93.93	68	91.43	38
	desc-sibdescOf	37.18	33	70.02	12	91.20	55	70.33	51
	any	69.02	59	93.62	48	93.12	48	82.15	17
	average	*37.12*	*32*	*65.81*	*20*	*87.18*	*55*	*73.39*	*31*
Medicine	*childOf*	0.01	2	4.62	2	9.09	4	2.84	4
	descOf	17.69	20	43.24	9	55.80	17	49.89	45
	desc-sibdescOf	24.52	20	50.40	8	59.20	12	53.91	18
	any	63.28	51	92.22	47	91.51	47	78.65	16
	average	*26.38*	*23*	*47.62*	*17*	*53.90*	*20*	*46.32*	*21*
Music	*childOf*	23.91	38	79.90	40	91.92	82	49.54	54
	descOf	48.45	39	87.19	21	97.53	84	96.27	37
	desc-sibdescOf	57.70	42	77.54	11	94.14	49	91.12	44
	any	75.75	53	88.16	48	89.12	48	76.86	17
	average	*51.45*	*43*	*83.20*	*30*	*93.18*	*66*	*78.45*	*38*
Order	*childOf*	2.25	2	37.25	3	27.86	2	17.66	3
	descOf	6.81	2	50.21	3	37.06	2	35.21	3
	desc-sibdescOf	48.35	50	78.02	18	91.38	49	79.12	38
	any	63.60	48	92.58	46	92.12	46	80.88	16
	average	*30.25*	*26*	*64.52*	*18*	*62.11*	*25*	*53.22*	*15*
People	*childOf*	0.42	2	15.61	2	13.37	2	6.71	2
	descOf	6.48	2	36.88	2	37.43	2	29.29	2
	desc-sibdescOf	8.94	3	43.43	3	33.18	2	32.59	3
	any	57.19	49	93.21	47	92.95	47	71.91	16
	average	*18.26*	*14*	*47.28*	*14*	*44.23*	*13*	*35.13*	*6*
Recipe	*childOf*	7.48	2	12.22	2	4.17	2	5.57	2
	descOf	0.01	2	19.39	2	10.65	2	12.38	2
	desc-sibdescOf	44.73	93	81.36	47	80.80	47	57.36	32
	any	46.46	47	87.58	47	87.33	47	66.22	16
	average	*27.17*	*36*	*50.14*	*25*	*45.74*	*25*	*35.38*	*13*
Shakespeare	*childOf*	1.15	3	28.41	3	18.02	2	10.10	3
	descOf	3.90	3	25.17	3	20.40	2	18.44	3
	desc-sibdescOf	44.71	96	76.01	60	75.91	61	80.65	99
	any	52.11	48	85.67	48	85.59	48	71.76	17
	average	*25.47*	*38*	*53.82*	*29*	*49.98*	*28*	*45.24*	*31*
Wikipedia	*childOf*	32.13	42	64.92	14	90.24	82	76.62	48
	descOf	47.05	42	74.12	11	95.65	82	97.29	20
	desc-sibdescOf	52.83	52	72.87	14	94.93	80	93.44	16
	any	61.37	60	93.02	47	92.56	47	78.59	17
	average	*48.35*	*49*	*76.23*	*22*	*93.35*	*73*	*86.49*	*25*

actually improved by taking into account the implicit order and structural relations of the input data as well as by employing semantic relatedness measures for the tag concepts.

We plan to deepen our analysis and understanding of PageRank formulations for structural sense ranking in labeled tree data. In this respect, evaluation on large scale data with different levels of heterogeneity (both structural and semantic) is certainly needed. Moreover, we are aware of the opportunity of exploiting Web sources, including Wikipedia, as knowledge bases for computing semantic relatedness (e.g., [8, 30, 14]), in the attempt of overcoming knowledge acquisition and coverage problems typical of conventional lexical ontologies like WordNet.

6. REFERENCES

[1] E. Agirre and A. Soroa. Using the Multilingual Central Repository for Graph-Based Word Sense Disambiguation. In *Proc. Int. Conf. on Language Resources and Evaluation (LREC)*, 2008.

[2] E. Agirre and A. Soroa. Personalizing PageRank for Word Sense Disambiguation. In *Proc. 12th Conf. of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 33–41, 2009.

[3] A. Algergawy, R. Nayak, and G. Saake. Element Similarity measures in XML schema matching. *Information Sciences*, 180:4975–4998, 2010.

[4] S. Banerjee and T. Pedersen. Extended Gloss Overlaps as a Measure of Semantic Relatedness. In *Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 805–810, 2003.

[5] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1-7):107–117, 1998.

[6] A. Budanitsky and G. Hirst. Evaluating WordNet-based Measures of Lexical Semantic Relatedness. *Comput. Ling.*, 32(1):13–47, 2006.

[7] D. De Cao, R. Basili, M. Luciani, F. Mesiano, and R. Rossi. Robust and Efficient PageRank for Word Sense Disambiguation. In *Proc. Workshop on Graph-based Methods for Natural Language Processing*, 2010.

[8] R. Cilibrasi and P. M. B. Vitányi. The Google Similarity Distance. *IEEE Trans. Knowl. Data Eng.*, 19(3):370–383, 2007.

[9] P. De Meo, G. Quattrone, G. Terracina, and D. Ursino. Integration of XML Schemas at Various Severity Levels. *Information Systems*, 31(6):397–434, 2006.

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

[11] A. Formica. Similarity of XML-Schema Elements: A Structural and Information Content Approach. *Comput. J.*, 51(2):240–254, 2008.

[12] J. Gracia, M. d'Aquin, and E. Mena. Large Scale Integration of Sense for the Semantic Web. In *Proc. Conf. on World Wide Web (WWW)*, pages 611–620, 2009.

[13] T. H. Haveliwala. Topic-Sensitive PageRank: A Context-Sensitive Ranking Algorithm for Web Search. *IEEE Trans. Knowl. Data Eng.*, 15(4):784–796, 2003.

[14] A. Hawalah and M. Fasli. A graph-based approach to measuring semantic relatedness in ontologies. In *Proc. ACM Int. Conf. on Web Intelligence, Mining and Semantics (WIMS)*, 2011.

[15] M. Lesk. Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to tell a pine cone from a ice cream cone. In *Proc. ACM SIGDOC*, pages 24–26, 1986.

[16] F. Mandreoli, R. Martoglia, and E. Ronchetti. Versatile Structural Disambiguation for Semantic-aware Applications. In *Proc. ACM Conf. on Information and Knowledge Management (CIKM)*, pages 209–216, 2005.

[17] R. Mihalcea, P. Tarau, and E. Figa. PageRank on Semantic Networks, with Application to Word Sense Disambiguation. In *Proc. 20th Int. Conf. on Computational Linguistics (COLING)*, 2004.

[18] R. Navigli. Word sense disambiguation: A survey. *ACM Comput. Surv.*, 41(2), 2009.

[19] R. Nayak and W. Iryadi. XML schema clustering with semantic and hierarchical similarity measures. *Knowledge-Based Systems*, 20:336–349, 2007.

[20] S. Patwardhan, S. Banerjee, and T. Pedersen. Using Measures of Semantic Relatedness for Word Sense Disambiguation. In *Proc. Conf. on Intelligent Text Processing and Computational Linguistics (CICLing)*, pages 241–257, 2003.

[21] E. Rahm and P. A. Bernstein. A survey of approaches to automatic schema matching. *VLDB J.*, 10(4):334–350, 2001.

[22] P. Resnik and D. Yarowsky. Distinguishing Systems and Distinguishing Senses: New Evaluation Methods for Word Sense Disambiguation. *Nat. Lang. Eng.*, 5(3):113–133, 2000.

[23] I. Sanz, M. Mesiti, G. Guerrini, and R. Berlanga. Fragment-based approximate retrieval in highly heterogeneous XML collections. *Data & Knowledge Engineering*, 64:266–293, 2008.

[24] M. Smiljanic, M. van Keulen, and W. Jonker. Formalizing the XML Schema Matching Problem as a Constraint Optimization Problem. In *Proc. Conf. on Database and Expert Systems Applications (DEXA)*, pages 333–342, 2005.

[25] A. Tagarelli and S. Greco. Toward Semantic XML Clustering. In *Proc. SIAM Conf. on Data Mining*, pages 188–199, 2006.

[26] A. Tagarelli, M. Longo, and S. Greco. Word Sense Disambiguation for XML Structure Feature Generation. In *Proc. European Semantic Web Conf. (ESWC)*, pages 143–157, 2009.

[27] M. Theobald, R. Schenkel, and G. Weikum. Exploiting Structure, Annotation, and Ontological Knowledge for Automatic Classification of XML Data. In *Proc. ACM SIGMOD WebDB Workshop*, pages 1–6, 2003.

[28] W. Xing and A. Ghorbani. Weighted PageRank Algorithm. In *Proc. Conf. on Communication Networks and Services Research (CNSR)*, pages 305–314, 2004.

[29] T. Zesch and I. Gurevych. Wisdom of crowds versus wisdom of linguists – measuring the semantic relatedness of words. *Nat. Lang. Eng.*, 16(1):25–59, 2009.

[30] T. Zesch and I. Gurevych. The More the Better? Assessing the Influence of Wikipedia's Growth on Semantic Relatedness Measures. In *Proc. Int. Conf. on Language Resources and Evaluation (LREC)*, pages 1374–1380, 2010.