Graph Connectivity Measures for Unsupervised Word Sense Disambiguation

Roberto Navigli
Dipartimento di Informatica
Università di Roma "La Sapienza"
navigli@di.uniroma1.it

Mirella Lapata
School of Informatics
University of Edinburgh
mlap@inf.ed.ac.uk

Abstract

Word sense disambiguation (WSD) has been a long-standing research objective for natural language processing. In this paper we are concerned with developing graph-based unsupervised algorithms for alleviating the data requirements for large scale WSD. Under this framework, finding the right sense for a given word amounts to identifying the most "important" node among the set of graph nodes representing its senses. We propose a variety of measures that analyze the connectivity of graph structures, thereby identifying the most relevant word senses. We assess their performance on standard datasets, and show that the best measures perform comparably to state-of-the-art.

1 Introduction

Word sense disambiguation (WSD), the ability to identify the intended meanings of words (word senses) in context, is a central research topic in Natural Language Processing. Sense disambiguation is often characterized as an intermediate task, which is not an end in itself, but essential for many applications requiring broad-coverage language understanding. Examples include machine translation [Vickrey *et al.*, 2005], information retrieval [Stokoe, 2005], question answering [Ramakrishnan *et al.*, 2003], and summarisation [Barzilay and Elhadad, 1997].

Recent advances in WSD have benefited greatly from the availability of corpora annotated with word senses. Most accurate WSD systems to date exploit supervised methods which automatically learn cues useful for disambiguation from hand-labeled data. Although supervised approaches outperform their unsupervised alternatives (see Snyder and Palmer [2004] for an overview), they often require large amounts of training data to yield reliable results [Yarowsky and Florian, 2002], and their coverage is typically limited to the words for which sense labeled data exist. Unfortunately, creating sense tagged corpora manually is an expensive and labor-intensive endeavor [Ng, 1997] which must be repeated for new domains, languages, and sense inventories. Given the data requirements for supervised WSD and the current paucity of suitable data for many languages and text genres, unsupervised approaches would seem to offer near-term hope for large scale sense disambiguation.

Most unsupervised methods can be broadly divided in two categories, namely graph-based ones and similarity-based ones. Graph-based algorithms often consist of two stages [Barzilay and Elhadad, 1997; Navigli and Velardi, 2005; Mihalcea, 2005]. First, a graph is built representing all possible interpretations of the word sequence being disambiguated. Graph nodes correspond to word senses, whereas edges represent dependencies between senses (e.g., synomymy, antonymy). Next, the graph structure is assessed to determine the importance of each node. Here, sense disambiguation amounts to finding the most "important" node for each word. Similarity-based algorithms assign a sense to an ambiguous word by comparing each of its senses with those of the words in the surrounding context [Lesk, 1986; McCarthy *et al.*, 2004; Mohammad and Hirst, 2006]. The sense whose definition has the highest similarity is assumed to be the correct one. The algorithms differ in the type of similarity measure they employ and the adopted definition of context which can vary from a few words to the entire corpus. In graph-based methods word senses are determined *collectively* by exploiting dependencies across senses, whereas in similarity-based approaches each sense is determined for each word *individually* without considering the senses assigned to neighboring words. Experimental comparisons between the two algorithm types [Mihalcea, 2005; Brody *et al.*, 2006] indicate that graph-based algorithms outperform similarity-based ones, often by a significant margin.

In this paper we focus on graph-based methods for unsupervised WSD and investigate in depth the role of graph structure in determining WSD performance. Specifically, we compare and contrast various measures of graph connectivity that assess the relative importance of a node within the graph. Graph theory is abundant with such measures and evaluations have been undertaken in the context of studying the structure of a hyperlinked environment [Botafogo *et al.*, 1992] and within social network analysis [Hage and Harary, 1995]. Our experiments attempt to establish whether some of these measures are particularly appropriate for graph-based WSD. Such a comparative study is novel to our knowledge; previous work restricts itself to a single measure which is either devised specifically for WSD [Barzilay and Elhadad, 1997] or adopted from network analysis [Mihalcea, 2005; Navigli and Velardi, 2005]. Our contributions are three-fold: a general framework for graph-based WSD; an empirical comparison of a broad range of graph connectivity measures using standard evaluation datasets; and an investigation of the influence of the sense inventory on the resulting graph structure and consequently on WSD.

In the following section, we briefly introduce the graph-based WSD algorithm considered in this paper. Then we present and motivate several measures of graph connectivity and explain how they are adapted to WSD. Next, we describe our evaluation methodology and present our experimental results. We conclude the paper by discussing future work.

2 Graph-based WSD

In order to isolate the impact of graph connectivity measures on WSD, we devised a fairly general disambiguation algorithm that has very few parameters and relies almost exclusively on graph structure for inferring word senses. In common with much current work in WSD, we are assuming that meaning distinctions are provided by a reference lexicon, which encodes for each word a discrete set of senses. Although our experiments will use the WordNet sense inventory [Fellbaum, 1998], neither our graph-based algorithm nor the proposed connectivity measures are limited to this particular lexicon. Resources with alternative sense distinctions and structure could also serve as input to our method.

We can view WordNet as a graph whose nodes are concepts (represented by *synsets* (i.e., synonym sets)) and whose edges are semantic relations between concepts (e.g., *hypernymy*, *meronymy*). For each sentence we build a graph $G = (V, E)$, which is induced from the graph of the reference lexicon. More formally, given a sentence $\sigma = w_1, w_2, \ldots, w_n$, where w_i is a word, we perform the following steps to construct G:

1. Initially, $V_\sigma := \bigcup_{i=1}^{n} Senses(w_i)$, where $Senses(w_i)$ is the set of senses of w_i in WordNet; in other words, V_σ represents all possible interpretations of sentence σ. We set $V := V_\sigma$ and $E := \emptyset$;
2. For each node $v \in V_\sigma$, we perform a depth-first search of the WordNet graph: every time we encounter a node $v' \in V_\sigma$ ($v' \neq v$) along a path $v \rightarrow v_1 \rightarrow \cdots \rightarrow v_k \rightarrow v'$, we add all intermediate nodes and edges on the path from v to v': $V := V \cup \{v_1, \ldots, v_k\}$ and $E := E \cup \{(v, v_1), \ldots, (v_k, v')\}$. For efficiency reasons, we allow paths of limited length (≤ 6 edges).

We thus obtain a subgraph of the entire lexicon which includes vertices reasonably useful for disambiguation: each vertex is at distance ≤ 3 edges from some vertex in the original set V_σ of word senses. Given a sentence σ, our aim is to select for each word $w_i \in \sigma$ the most appropriate sense $S_{w_i} \in Senses(w_i)$. The latter is determined by ranking each vertex in the graph G according to its importance. In Section 3 we discuss several measures that operationalize importance in graph-theoretic terms. Here, we will briefly note that these measures can be either *local* or *global*. Local measures capture the degree of connectivity conveyed by a single vertex in the graph towards all other vertices, whereas global measures estimate the overall degree of connectivity of the entire graph.

The choice of connectivity measure influences the selection process for the highest-ranked sense. Given a *local measure* l, and the set of vertices V_σ, we induce a ranking of the vertices $rank_l$ such that $rank_l(v) \leq rank_l(v')$ iff $l(v) \geq l(v')$. Then, for each word $w_i \in \sigma$, we select the best-ranking sense in $Senses(w_i)$ according to $rank_l$. A *global measure* g characterizes the overall graph structure G and is thus not particularly helpful in selecting a unique sense for ambiguous words – G collectively represents all interpretations of σ. We get around this problem, by applying g iteratively to each interpretation of σ and selecting the highest scoring one. An interpretation is a subgraph $G' \subseteq G$ such that G' includes one and only one sense of each word in sentence σ and all their corresponding intermediate nodes (see step (2) above). So if our sentence has five interpretations, we will measure the connectivity of the resulting subgraphs five times.

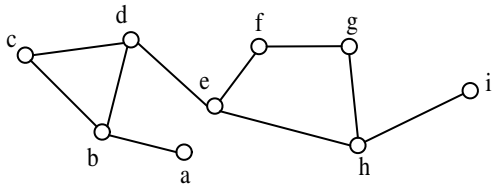

Figure 1: An example of a graph.

3 Connectivity Measures

In this section we describe the measures of graph connectivity we consider for unsupervised WSD. Although our measures can be applied to both directed and undirected graphs, for WSD purposes we are assuming that we are dealing with undirected graphs (we view an undirected edge as a pair of directed edges). This is motivated by the fact that semantic relations often have an inverse counterpart (e.g., hypernymy is the inverse relation of hyponymy).

We next introduce the distance function $d(u, v)$, which is used by some of the measures discussed below:

$$d(u,v) = \begin{cases} \text{length of shortest path} & \text{if } u \rightsquigarrow v \\ K & otherwise \end{cases}$$

where $u \rightsquigarrow v$ indicates the existence of a path from u to v, and K is a conversion constant [Botafogo *et al.*, 1992], which replaces the ∞ distance with an integer when v is not reachable from u (we choose $K = |V|$, as the length of any simple path is $< |V|$). As an example consider the graph in Figure 1 where $d(a, i) = 5$, $d(c, g) = 4$, and so on.

3.1 Local Measures

Local measures of graph connectivity determine the degree of relevance of a single vertex v in a graph G. They can thus be viewed as measures of the influence of a node over the spread of information through the network. Formally, we define a local measure l as:

$$l : V \rightarrow [0, 1]$$

A value close to 1 indicates that a vertex is relatively important, whereas a value close to 0 indicates that the vertex is peripheral.

Several local measures of graph connectivity have been proposed in the literature (see Wasserman and Faust [1994] for a comprehensive overview). A large number rely on the notion of *centrality*: a node is central if it is maximally connected to all other nodes. In the following, we consider three best-known measures of centrality, namely degree, closeness, and betweeness [Freeman, 1979], and variations thereof. We also show how graph connectivity can be computed by solving a max-flow problem.

In-degree Centrality The simplest way to measure vertex importance is by its degree, i.e., the number of edges terminating in a given vertex:

$$indeg(v) = |\{(u, v) \in E : u \in V\}|$$

A vertex is central, if it has a high degree. In-degree centrality is the degree of a vertex normalized by the maximum degree:

$$C_D(v) = \frac{indeg(v)}{|V|-1}$$

So, according to the graph in Figure 1, $C_D(a) = \frac{1}{8}$, $C_D(d) = C_D(e) = C_D(h) = \frac{3}{8}$, and $C_D(c) = \frac{2}{8}$.

Eigenvector Centrality A more sophisticated version of degree centrality is eigenvector centrality. Whereas the former gives a simple count of the number of connections a vertex has, the latter acknowledges that not all connections are equal. It assigns relative scores to all nodes in the graph based on the principle that connections to nodes having a high score contribute more to the score of the node in question [Bonacich, 1972]. PageRank [Brin and Page, 1998] and HITS [Kleinberg, 1998] are popular variants of the eigenvector centrality measure and have been almost exclusively used in graph-based WSD [Mihalcea, 2005; Navigli and Velardi, 2005].

PageRank determines the relevance of a node v recursively based on a Markov chain model. All nodes that link to v contribute towards determining its relevance. Each contribution is given by the page rank value of the respective node ($PR(u)$) divided by the number of its neighbors:

$$PR(v) = \frac{(1-\alpha)}{|V|} + \alpha \sum_{(u,v)\in E} \frac{PR(u)}{outdegree(u)}$$

The overall contribution is weighted with a damping factor α, which implements the so-called random surfer model: with probability $1-\alpha$, the random surfer is expected to discontinue the chain and select a random node (i.e., page), each with relevance $\frac{1}{|V|}$.

In contrast, HITS (Hypertext Induced Topic Selection) determines two values for each node v, the authority ($a(v)$) and the hub value ($h(v)$). These are defined in terms of one another in a mutual recursion:

$$h(v) = \sum_{u:(u,v)\in E} a(u) \quad ; \quad a(v) = \sum_{u:(v,u)\in E} h(u)$$

Intuitively, a good hub is a node that points to many good authorities, whereas a good authority is a node that is pointed to by many good hubs. A major difference between HITS and PageRank is that the former is computed dynamically on a subgraph of relevant pages, whereas the latter takes the entire graph structure into account.

If we apply HITS to the graph in Figure 1, we get the following authority values: $a(d) = 0.484, a(e) = 0.435, a(b) = 0.404, \ldots, a(a) = 0.163, a(i) = 0.132$. The PageRank values are $PR(d) = PR(e) = PR(b) = PR(d) = 0.15$, $PR(f) = PR(g) = PR(c) = 0.1$, and $PR(i) = PR(a) = 0.05$. While HITS yields a fine-grained ranking, PageRank delivers only three different ranks, ranging from central to peripheral. Notice that, since our graphs are undirected, the authority and hub values coincide.

Key Player Problem (KPP) KPP is similar to the better known closeness centrality measure[1] [Freeman, 1979]. Here, a vertex is considered important if it is relatively close to all other vertices [Borgatti, 2003]:

$$KPP(v) = \frac{\sum_{u\in V:u\neq v} \frac{1}{d(u,v)}}{|V|-1}$$

where the numerator is the sum of the inverse shortest distances between v and all other nodes and the denominator is the number of nodes in the graph (excluding v).

[1]Closeness centrality is defined as the total geodesic distance from a given node to all other nodes. We consider only KPP since it outperformed closeness centrality in our experiments.

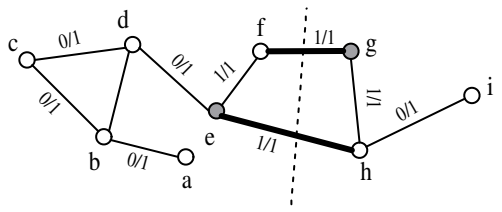

Figure 2: The maximum flow between nodes e and g (edges are labeled with the pair flow/capacity).

For example, the KPP for nodes a and f in Figure 1 is $KPP(a) = \frac{1+\frac{1}{2}+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\frac{1}{4}+\frac{1}{5}+\frac{1}{5}}{8} = 0.40$ and $KPP(f) = \frac{1+1+\frac{1}{2}+\frac{1}{2}+\frac{1}{3}+\frac{1}{3}+\frac{1}{3}+\frac{1}{4}}{8} = 0.53$, respectively.

Betweenness Centrality The betweenness of vertex v is calculated as the fraction of shortest paths between node pairs that pass through v [Freeman, 1979]. Formally, betweenness is defined as:

$$betweenness(v) = \sum_{s,t\in V:s\neq v\neq t} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

where σ_{st} is the number of shortest paths from s to t, and $\sigma_{st}(v)$ the number of shortest paths from s to t that pass through vertex v. We normalize by dividing $betweenness(v)$ by the maximum number of node pairs excluding v:

$$C_B(v) = \frac{betweenness(v)}{(|V|-1)(|V|-2)}$$

The intuition behind betweenness is that a node is important if it is involved in a large number of paths compared to the total set of paths. With reference to Figure 1, the pairs of vertices (x, g) and (g, x), with $x \in \{a, b, c, d, e\}$, are connected by two possible shortest paths, including either f or h as an intermediate vertex. Thus, $\sigma_{xg} = \sigma_{gx} = 2$ and $\sigma_{xg}(f) = \sigma_{gx}(f) = 1$. We can now calculate $betweenness(f) = 10 \cdot \frac{1}{2} = 5$ and $C_B(f) = \frac{5}{8\cdot 7} = \frac{5}{56}$.

Maximum Flow Let $G = (V, E)$ be a connected graph, and let $c : E \to \mathbb{R}$ be a capacity function such that every edge $(u, v) \in E$ is associated with capacity $c(u, v)$. We further distinguish two vertices, the source s and the sink t. Finally, let $f : V \times V \to \mathbb{R}$ be a function called flow.

Given an s-t-cut (S, T), i.e., a partition of V into two disjoint sets S and T, such that $s \in S$ and $t \in T$, the s-t-flow of the cut represents the amount of information that can be conveyed from s to t through the cut while obeying all capacity constraints. It is defined as:

$$f(S, T) = \sum_{u\in S, v\in T} f(u, v)$$

The maximum s-t-flow of a graph G has the highest value among all s-t-cuts. For example, if we fix e as the source and g as the sink (or viceversa) in the graph in Figure 2, the maximum flow that can be conveyed equals to the sum of the maximum flows $f(f, g) + f(e, h) = 1 + 1 = 2$. This flow is determined by taking into account the paths $e \to f \to g$ and $e \to h \to g$. In fact, Menger's theorem states that the maximum s-t-flow in undirected graphs corresponds to the number of independent paths between a pair of vertices.

In the context of WSD, maximum s-t-flows provide a relevance ranking on the set of vertices: the more flow is

conveyed from s to t, the more relevant the sink is. Initially, the capacity of each edge $(u, v) \in E$ is set to 1 and its flow $f(u, v)$ to 0. To compute an overall score for each vertex, we execute the following steps:

$\forall v \in V$ $score(v) := 0$
$\forall s, t \in V$, $s \neq t$, **do**
    $score(t) := score(t) + max$ s-t-flow
$\forall v \in V$, **do**
    $score(v) := \frac{score(v)}{\max_{u \in V} score(u)}$

The resulting score for each vertex $v \in V$ is the sum of the maximum flows having v as a sink normalized by the maximum score. If G is disconnected, we do not need to apply the algorithm separately to each connected component, since the maximum flow between s and t is 0 if t is not reachable from s. We calculate the maximum flow with the Ford-Fulkerson [1962] algorithm based on the notion of augmenting paths. We adopted Edmonds and Karp's [1972] efficient implementation.

3.2 Global Measures

Global connectivity measures are concerned with the structure of the graph as a whole rather than with individual nodes. Here we discuss three well-known measures, namely compactness, graph entropy, and edge density.

Compactness This measure represents the extent of cross referencing in a graph [Botafogo *et al.*, 1992]: when compactness is high, each vertex can be easily reached from other vertices. The measure is defined as:

$$CO(G) = \frac{Max - \sum_{u \in V} \sum_{v \in V} d(u, v)}{Max - Min}$$

where $Max = K \cdot |V|(|V|-1)$ is the maximum compactness (i.e., for a disconnected graph) and $Min = |V|(|V| - 1)$ is the minimum compactness (i.e., for a fully connected graph). The compactness of the graph in Figure 1 is: $CO(G) = \frac{(9 \cdot 9 \cdot 8) - 176}{(9 \cdot 9 \cdot 8) - (9 \cdot 8)} = \frac{472}{576} = 0.819$ (in this example, $K = |V| = 9$).

Graph Entropy Entropy measures the amount of information (or alternatively uncertainty) in a random variable. In graph-theoretic terms, high entropy indicates that many vertices are equally important, whereas low entropy indicates that only a few vertices are relevant. We define a simple measure of graph entropy as:

$$H(G) = -\sum_{v \in V} p(v) log(p(v))$$

where the vertex probability $p(v)$ is determined by the degree distribution $\left\{ \frac{indeg(v)}{2|E|} \right\}_{v \in V}$. To obtain a measure with a $[0, 1]$ range, we divide $H(G)$ by the maximum entropy given by $log|V|$. For example, the distribution associated with the graph in Figure 1 is: $(\frac{1}{20}, \frac{3}{20}, \frac{2}{20}, \frac{3}{20}, \frac{3}{20}, \frac{2}{20}, \frac{2}{20}, \frac{3}{20}, \frac{1}{20})$ leading to an overall graph entropy $H(G) = \frac{3.07}{log\ 9} = 0.969$.

Edge Density Finally, we propose the use of edge density as a simple global connectivity measure. Edge density is calculated as the ratio of edges in a graph over the number of edges of a complete graph with $|V|$ vertices (given by $2 \cdot \binom{|V|}{2}$). Formally:

$$ED(G) = \frac{|E(G)|}{2 \cdot \binom{|V|}{2}}$$

For example, the graph in Figure 1 has edge density $ED(G) = \frac{10}{2 \cdot \binom{9}{2}} = \frac{10}{72} = 0.138$.

4 Experimental Setup

Sense inventory The graph connectivity measures just described were incorporated in the disambiguation algorithm introduced in Section 2. As explained earlier, disambiguation proceeds on a sentence-by-sentence basis. Each sentence is represented by a graph corresponding to meaning distinctions provided by a reference lexicon. In our experiments we employed two such lexicons. The first is WordNet 2.0 [Fellbaum, 1998], a resource commonly used in WSD research (see Snyder and Palmer [2004]). We also used an extended version of WordNet created by Navigli [2005]. The latter contains additional *semantic relatedness* edges (approximately 60,000) that relate associated concepts across parts of speech (e.g., *dog* and *bark*, *drink* and *glass*). These were automatically extracted from collocation resources (e.g., Oxford Collocations, Longman Language Activator) and semi-automatically disambiguated.

Data We selected two standard data sets for evaluating our connectivity measures, namely the SemCor corpus [Miller *et al.*, 1993] and the Senseval-3 English all-words test set [Snyder and Palmer, 2004]. SemCor is a subset of the Brown corpus, and includes more than 200,000 content words manually tagged with WordNet senses. Senseval-3 is a subset of the Penn Treebank corpus and contains 2,081 content words, again labeled with WordNet senses. We exhaustively tested our measures on the SemCor dataset. The best performing one was also evaluated on Senseval-3 and compared with state-of-the-art.

Graph construction In order to speed up the graph construction process, all paths connecting pairs of senses in both versions of WordNet were exhaustively enumerated and stored in a database which was consulted at run-time during disambiguation. Unfortunately, the use of global connectivity measures makes our WSD algorithm susceptible to combinatorial explosion, since all possible interpretations of a given sentence must be ranked (see Section 2). We used simulated annealing to heuristically explore the entire space of interpretations for a given sentence [Cowie *et al.*, 1992].

Baseline and Upper Bound Our graph-based algorithm was compared against a naive baseline that selects a sense for each word at random. As an upper bound, we used the first-sense heuristic which assigns all instances of an ambiguous word its most frequent sense according to the manually annotated SemCor. It is important to note that current unsupervised WSD approaches—and also many supervised ones—rarely outperform this simple heuristic [McCarthy *et al.*, 2004].

5 Results

Our results on SemCor are summarized in Table 1. We report performance solely on polysemous words, i.e., words with more than one WordNet sense.

Let us first concentrate on the results we obtained with the standard WordNet inventory. As can be seen, almost all measures perform better than the random sense baseline. The

	Measure	WordNet Prec	WordNet Rec	WordNet F1	EnWordNet Prec	EnWordNet Rec	EnWordNet F1
	Baseline	23.7	23.7	23.7	23.7	23.7	23.7
Local	InDegree	35.3	24.0	28.6	44.2	37.0	40.3
Local	Betweenness	38.4	15.5	22.1	45.0	31.1	36.8
Local	**KPP**	**31.8**	**31.8**	**31.8**	**40.5**	**40.5**	**40.5**
Local	HITS	31.7	17.2	22.3	39.4	31.1	34.8
Local	PageRank	35.3	24.0	28.6	44.0	36.8	40.0
Local	Maxflow	33.0	24.3	28.0	41.8	35.2	38.2
Global	Compactness	29.8	27.9	28.8	36.3	35.5	35.9
Global	GraphEntropy	30.3	28.4	29.4	30.9	30.2	30.5
Global	EdgeDense	29.9	27.9	28.9	35.6	34.6	35.1
	UpperBnd	68.8	68.8	68.8	68.8	68.8	68.8

Table 1: Performance of connectivity measures on SemCor.

differences are significant both in terms of precision and recall (using a χ^2 test). HITS and Betweenness yield significantly better precision but worse recall. The best performing local measure is KPP (F1 31.8%), whereas the best performing global measure is graph entropy (F1 29.4%). KPP is significantly better than graph entropy both in terms of precision and recall (again using a χ^2 test). We conjecture that the inferior performance of the global measures is due to the use of a heuristic algorithm for searching the interpretation space. Interestingly, PageRank yields significantly better recall and precision than HITS. We attribute the difference in performance to the fact that PageRank implements the random surfer model. Finally, note that a relatively simple measure like InDegree performs as well as PageRank (F1 is 28.6% for both measures). This is not entirely surprising. The PageRank value of a node is proportional to its degree in undirected graphs. Furthermore, research on directed graphs has experimentally shown that the two measures are broadly equivalent [Upstill *et al.*, 2003].

We now turn to the performance of the different measures when the enriched WordNet (EnWordNet) is used. Here we also observe that all measures are significantly better than the baseline (in terms of precision and recall). The best performing global measure is Compactness (F1 35.9%). The best local measures are InDegree, KPP and PageRank (F1 is around 40%). KPP performs consistently well with WordNet and its enriched version. All three local measures achieve significantly better precision and recall than Compactness. It seems that local measures benefit from a denser reference lexicon, with a large number of semantic relations, whereas global measures are disadvantaged due to the combinatorial explosion problem discussed above. To further substantiate this, we analyzed how KPP's performance varies when an increasing number of edges is considered for disambiguation. Figure 3 shows that F1 increases when a sense has a large number of edges. In fact, when more than 200 edges are taken into account, KPP obtains an F1 of 85%. Notice that we are excluding unambiguous words and that there are at least 1,500 occurrences of word senses in the SemCor corpus for each interval in the graph.

We next assess how KPP performs on the Senseval-3 English all-words test set when using the enriched WordNet. We also compare our results with the best unsupervised system that took part in the Senseval-3 competition[2]. The latter is

[2]See `http://www.senseval.org/senseval3` for details on the competition and participating systems.

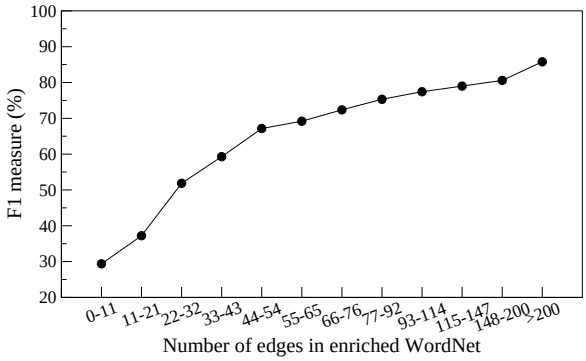

Figure 3: Performance of KPP by number of edges.

Measure	Part of speech	Prec	Rec	F1
KPP	Nouns	61.9	61.9	61.9
	Adjectives	62.8	62.8	62.8
	Verbs	36.1	36.1	36.1
IRST-DDD	Nouns	63.3	61.2	62.2
	Adjectives	68.2	65.6	66.9
	Verbs	51.6	49.2	50.4

Table 2: Results on the Senseval-3 all words task by part of speech.

a similarity-based algorithm. It was developed by Strapparava *et al.* [2004] and performs domain driven disambiguation (IRST-DDD). The approach compares the domain of the context surrounding the target word with the domains of its senses and uses a version of WordNet augmented with domain labels (e.g., economy, geography). Table 2 shows how performance varies across parts of speech.[3] KPP performs comparably to IRST-DDD for nouns and adjectives (the differences in recall and precision are not statistically significant). IRST-DDD yields significantly better results for verbs. This can be explained by the fact that the enriched WordNet contains a significantly smaller number of relatedness edges for verbs than for nouns or adjectives and this impacts the performance of KPP. Also note that our experiments focused primarily on graph connectivity measures. Consequently, we employed a relatively generic WSD algorithm (see Section 2) without additional tuning. For instance we could obtain improved results by considering word sequences larger than sentences or by weighting edges according to semantic importance (e.g., *hypernymy* is more important than *meronymy*).

6 Conclusions

In this paper we presented a study of graph connectivity measures for unsupervised WSD. We evaluated a wide range of local and global measures with the aim of isolating those that are particularly suited for this task. Our results indicate that local measures yield better performance than global ones. The best local measures are KPP, InDegree, and PageRank. KPP has a slight advantage over the other two measures, since it performs consistently well across experimental conditions. Our results are in agreement with Borgatti [2003] who shows

[3]F1 scores here are higher than those reported in Table 1. This is expected since the Senseval-3 data set contains monosemous words as well.

in the context of social network analysis that KPP is better than other measures (e.g., betweeness or in-degree centrality) at identifying which node in the graph is maximally connected to all other nodes. In linguistic terms this means that KPP selects maximally cohesive nodes which typically correspond to topical senses, thus indirectly enforcing the one-sense per discourse constraint. We also find that the employed reference dictionary critically influences WSD performance. We obtain a large F1 improvement (8.7% for KPP, 11.4% for InDegree) when adopting a version of WordNet enriched with thousands of relatedness edges. Interestingly, we observe that Indegree and PageRank yield performances comparable to KPP when the enriched WordNet is used. This is due to the increased number of relatedness edges which result in more densly connected graphs with more outgoing edges for every node. Centrality-based measures are particularly suited at identifying such nodes.

Beyond the specific WSD algorithm presented in this paper, our results are relevant for other graph-based approaches to WSD [Mihalcea, 2005; Navigli and Velardi, 2005]. Our experiments indicate that performance could potentially increase when the right connectivity measure is chosen. The proposed measures are independent of the adopted reference lexicon and the graph construction algorithm. They induce a sense ranking solely by considering graph connectivity and can thus be easily ported across algorithms, languages, and sense inventories.

An important future direction lies in combining the different measures in a unified framework. Notice in Table 1 that certain measures yield high precision (e.g., Betweeness) whereas others yield high recall (e.g., KPP). We will also evaluate the impact of KPP in other applications such as graph-based summarization and the recognition of entailment relations.

References

[Barzilay and Elhadad, 1997] Regina Barzilay and Michael Elhadad. Using lexical chains for text summarization. In *Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, pages 10–17, Madrid, Spain, 1997.

[Bonacich, 1972] B. P. Bonacich. Factoring and weighing approaches to status scores and clique identification. *Journal of Mathematical Sociology*, 2:113–120, 1972.

[Borgatti, 2003] Stephen P. Borgatti. Identifying sets of key players in a network. In *Proceedings of the Conference on Integration of Knowledge Intensive Multi-Agent Systems*, pages 127–131, Boston, USA, 2003.

[Botafogo *et al.*, 1992] Rodrigo A. Botafogo, Ehud Rivlin, and Ben Shneiderman. Structural analysis of hypertexts: Identifying hierarchies and useful metrics. *ACM Transactions on Information Systems*, 10(2):142–180, 1992.

[Brin and Page, 1998] Sergey Brin and Michael Page. Anatomy of a large-scale hypertextual web search engine. In *Proceedings of the 7th Conference on World Wide Web*, pages 107–117, Brisbane, Australia, 1998.

[Brody *et al.*, 2006] Samuel Brody, Roberto Navigli, and Mirella Lapata. Ensemble methods for unsupervised WSD. In *Proceedings of the ACL/COLING*, Sydney, Australia, 2006.

[Cowie *et al.*, 1992] Jim Cowie, Joe Guthrie, and Louise Guthrie. Robust textual inference via graph matching. In *Proceedings of the 14th COLING*, pages 359–365, Nantes, France, 1992.

[Edmonds and Karp, 1972] J. Edmonds and R. M. Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM*, 19(2):248–264, 1972.

[Fellbaum, 1998] Christiane Fellbaum, editor. *WordNet: an Electronic Lexical Database*. MIT Press, 1998.

[Ford and Fulkerson, 1962] Lestor R. Ford and D. R. Fulkerson. *Flows in Networks*. Princeton University Press, 1962.

[Freeman, 1979] L C. Freeman. Centrality in networks: I. conceptual clarification. *Social Networks*, 1:215–239, 1979.

[Hage and Harary, 1995] P. Hage and F. Harary. Eccentricity and centrality in networks. *Social Networks*, 13:57–63, 1995.

[Kleinberg, 1998] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. In *Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms*, pages 668–677, San Francisco, California, 1998.

[Lesk, 1986] Michael Lesk. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the 5th SIGDOC*, pages 24–26, New York, NY, 1986.

[McCarthy *et al.*, 2004] Diana McCarthy, Rob Koeling, Julie Weeds, and John Carroll. Finding predominant senses in untagged text. In *Proceedings of the 42nd ACL*, pages 280–287, Barcelona, Spain, 2004.

[Mihalcea, 2005] Rada Mihalcea. Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling. In *Proceedings of the HLT/EMNLP*, pages 411–418, Vancouver, BC, 2005.

[Miller *et al.*, 1993] George Miller, Claudia Leacock, Tengi Randee, and Ross Bunker. A semantic concordance. In *Proceedings of the 3rd DARPA Workshop on HLT*, pages 303–308, Plainsboro, New Jersey, 1993.

[Mohammad and Hirst, 2006] Saif Mohammad and Graeme Hirst. Determining word sense dominance using a thesaurus. In *Proceedings of the 11th EACL*, pages 121–128, Trento, Italy, 2006.

[Navigli and Velardi, 2005] Roberto Navigli and Paola Velardi. Structural semantic interconnections: a knowledge-based approach to word sense disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(7):1075–1088, 2005.

[Navigli, 2005] Roberto Navigli. Semi-automatic extension of large-scale linguistic knowledge bases. In *Proceedings of the 18th FLAIRS*, pages 548–553, Clearwater Beach, Florida, 2005.

[Ng, 1997] Tou Hwee Ng. Getting serious about word sense disambiguation. In *Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*, pages 1–7, Washington, DC, 1997.

[Ramakrishnan *et al.*, 2003] Ganesh Ramakrishnan, Apurva Jadhav, Ashutosh Joshi, Soumen Chakrabarti, and Pushpak Bhattacharyya. Question answering via bayesian inference on lexical relations. In *Proceedings of the ACL Workshop on Multilingual Summarization and Question Answering*, pages 1–10, Sapporo, Japan, 2003.

[Snyder and Palmer, 2004] Benjamin Snyder and Martha Palmer. The English all-words task. In *Proceedings of the ACL SENSEVAL-3 Workshop*, pages 41–43, Barcelona, Spain, 2004.

[Stokoe, 2005] Christopher Stokoe. Differentiating homonymy and polysemy in information retrieval. In *Proceedings of the HLT/EMNLP*, pages 403–410, Vancouver, BC, 2005.

[Strapparava *et al.*, 2004] Carlo Strapparava, Alfio Gliozzo, and Claudio Giuliano. Pattern abstraction and term similarity for word sense disambiguation: Irst at senseval-3. In *Proceedings of the ACL SENSEVAL-3 Workshop*, pages 229–234, Barcelona, Spain, 2004.

[Upstill *et al.*, 2003] Trystan Upstill, Nick Craswell, and David Hawking. Predicting fame and fortune: Pagerank or indegree? In *Proceedings of the Australasian Document Computing Symposium*, Canberra, Australia, 2003.

[Vickrey *et al.*, 2005] David Vickrey, Luke Biewald, Marc Teyssier, and Daphne Koller. Word-sense disambiguation for machine translation. In *Proceedings of the HLT/EMNLP*, pages 771–778, Vancouver, BC, 2005.

[Wasserman and Faust, 1994] Stanley Wasserman and Kathernie Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, Cambridge, UK, 1994.

[Yarowsky and Florian, 2002] David Yarowsky and Radu Florian. Evaluating sense disambiguation across diverse parameter spaces. *Natural Language Engineering*, 9(4):293–310, 2002.