

Networks and Natural Language Processing

Dragomir R. Radev and Rada Mihalcea

■ Over the last few years, a number of areas of natural language processing have begun applying graph-based techniques. These include, among others, text summarization, syntactic parsing, word-sense disambiguation, ontology construction, sentiment and subjectivity analysis, and text clustering. In this paper, we present some of the most successful graph-based representations and algorithms used in language processing and try to explain how and why they work.

In a cohesive text, language units—whether they are words, phrases, or entire sentences—are connected through a variety of relations, which contribute to the overall meaning of the text and maintain the cohesive structure of the text and the discourse unity. Since the early ages of artificial intelligence, associative or semantic networks have been proposed as representations that enable the storage of such language units and the relations that interconnect them and that allow for a variety of inference and reasoning processes, simulating some of the functionalities of the human mind. The symbolic structures that emerge from these representations correspond naturally to graphs—where text constituents are represented as vertices and their interconnecting relations form the edges in the graph.

The last decade has brought on a number of exciting research papers that apply graph-based methods to an increasingly large range of natural language problems, ranging from lexical acquisition to text summarization. In this article, we overview several of these methods and their application to natural language processing. To reflect the fact that the algorithms and representations originate in different communities—natural language processing and graph theory—we will be using a dual vocabulary to describe these methods: networks are graphs, nodes are vertices, and links are edges.

In terms of *graph-based representations*, depending on the natural language-processing application, a variety of node and

edge types have been used. Text units of various sizes and characteristics can be added as vertices in the graph, for example, words, collocations, word senses, entire sentences, or even entire documents. Note that the graph nodes do not have to belong to the same category. For example, both sentences and words can be added as vertices in the same graph. Edges can represent cooccurrence (such as two words that appear in the same sentence or in the same dictionary definition), collocation (for example, two words that appear immediately next to each other or that may be separated by a conjunction), syntactic structure (for example, the parent and child in a syntactic dependency), and lexical similarity (for example, cosine between the vector representations of two sentences).

In terms of graph-based algorithms, the main methods used so far can be classified into: (1) semisupervised classification (Zhu and Ghahramani 2002; Zhu and Lafferty 2005; Toutanova, Manning, and Ng 2004; Radev 2004; Otterbacher, Erkan, and Radev 2005), where random walks or relaxation are applied on mixed sets of labeled and unlabeled nodes; (2) network analysis (Masucci and Rodgers 2006; , Caldeira et al. 2006), where network properties such as diameter, centrality, and so on, are calculated; (3) graph-based clustering methods (Pang and Lee 2004, Widows and Dorow 2002), such as min-cut methods; (4) minimum spanning-tree algorithms (McDonald et al. 2005).

In this article, we overview several

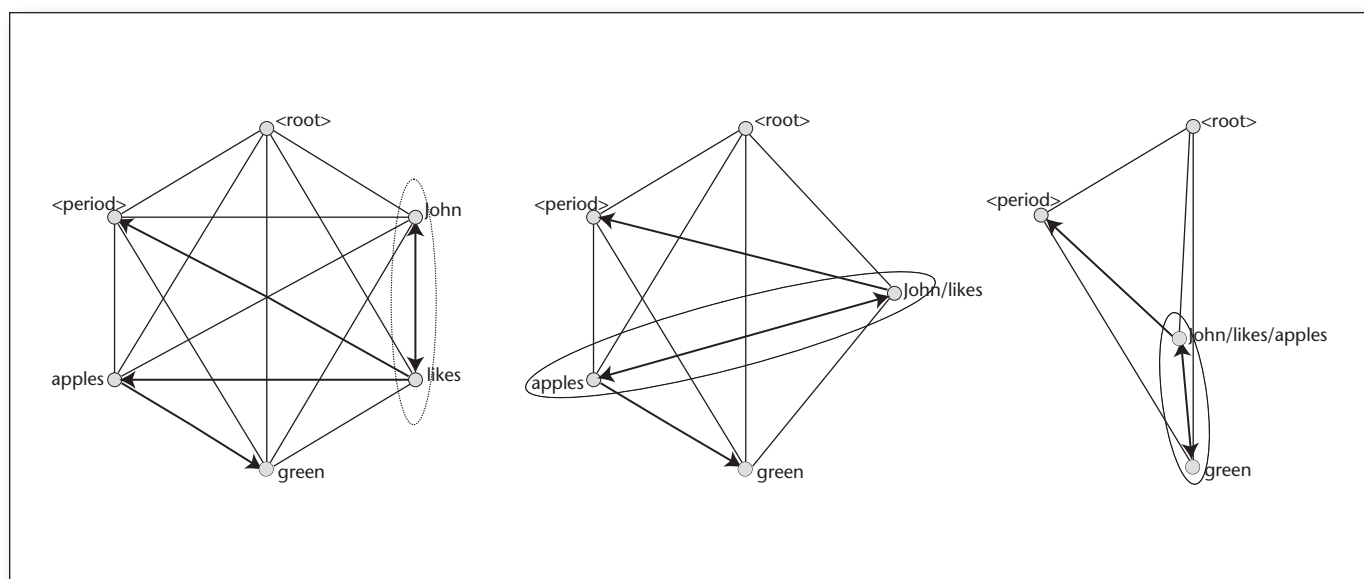


Figure 1. The Graphs Produced by Intermediate Steps of the MST Algorithm.

graph-based approaches for natural language-processing tasks, which we broadly group into three main categories. First, we review research work done in the area of syntax, including syntactic parsing, prepositional attachment, and coreference resolution. We then describe methods used in lexical semantics, including word-sense disambiguation, lexical acquisition, and sentiment and subjectivity analysis. Finally, we review several natural language-processing applications that rely on graph methods, including text summarization, passage retrieval, and keyword extraction.

Syntax

In this section we will discuss three papers, addressing methods for syntactic parsing (McDonald et al. 2005), prepositional attachment (Toutanova, Manning, and Ng 2004), and coreference resolution (Nicolae and Nicolae 2006).

Dependency Parsing

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajic (McDonald et al. 2005) take an unconventional approach to sentence parsing. They start by realizing that each dependency tree of a sentence is a directed subgraph of the full graph linking all words in the sentence. An approach like theirs would not work on the more widely known constituent trees, as they contain nonterminals. In dependency parsing, each sentence is represented as a tree, the root of which is typically the main predicate of the sentence (or it is a dummy node labeled root of which the main predicate is the sole child) and in which edges are used to connect each word to its dependency parent. For example, in the

sentence *John likes green apples*, the main predicate is *likes*, which takes two arguments: the liker (*John*) and the liked (*apples*). Finally, since *green* modifies *apples*, it is added to the tree as a child of *apples*. The final tree looks like this:

[likes [John] [apples [green]]].

McDonald and his colleagues build a full graph from the input sentence and then associate a score with each potential directed subtree of that graph that is equal to the sum of the scores of all edges that it includes. The score for each edge in the original graph is the product of a weight vector w and a feature representation of the edge $f(i, j)$. The goal of their parser is to find the tree with the highest score. Note that punctuation, including the final period of a sentence is used in the parsing process.

The Chu-Liu-Edmonds (CLE) algorithm (Chu and Liu 1965, Edmonds 1967) is used to find maximum spanning trees (MST) in directed graphs. The algorithm involves the following procedure: each node picks the neighbor with the highest score. The result is either a spanning tree or it has cycles. The CLE method collapses each such cycle into a single node and recomputes the scores of each edge incident on such a cycle. It can be shown that the MST built on the collapsed graph is the same as the MST computed on the original graph; the algorithm can be run in $O(n^2)$.

Let's consider an example consistent with McDonald, Pereira, Ribarov, and Hajic. The sentence that needs to be parsed is *John likes green apples*. The corresponding graph is shown in the leftmost graph in figure 1, and each node in the graph corresponds to a word in the sentence. After the first iteration of the algorithm, no tree is found that

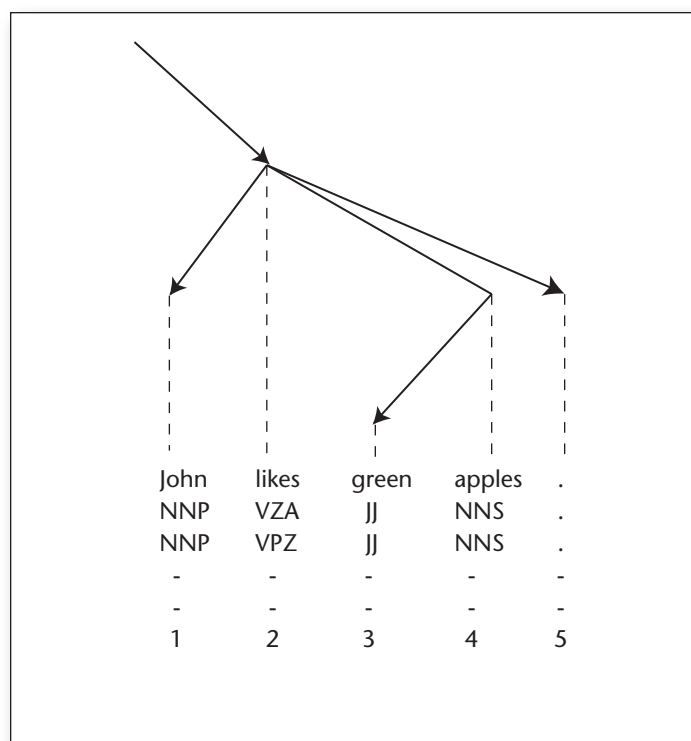


Figure 2. Output of the MST Parser.

covers all nodes, so the two closest nodes are collapsed, resulting in the second graph in figure 1. The process continues until the entire graph gets reduced to a single node through a series of iterations.

After all nodes have been collapsed into one, the MST is constructed by reversing the procedure and expanding all nodes into their constituents. The end result for this example is shown in figure 2.

McDonald, Pereira, Ribarov, and Hajic achieve state-of-the-art results with their parser on a standard English data set and better than state-of-the-art results on Czech (a free word-order language).

Prepositional Phrase Attachment

Prepositional phrase (PP) attachment is one of the most challenging problems in parsing. English grammar allows a prepositional phrase such as *with* to attach either to the main predicate of a sentence or to the noun phrase immediately preceding it. For example, *I ate pizza with olives* is an example of low (nominal) attachment whereas *I ate pizza with a knife* is an example of high (verbal) attachment. In naturally occurring English text, both types of attachment are commonly seen.

Kristina Toutanova, Christopher D. Manning, and Andrew Y. Ng. (2004) address the problem of PP attachment by casting it as a semisupervised learning process on graphs. Each node of the graph corresponds to a verb or a noun. Two nodes are connected if they appear in the same context (for

example, the verb nodes *hang* and *fasten* are connected because they both appear in the expressions *with a nail*. In a similar way, the noun nodes *nail* and *rivet* are connected to each other. Many other types of connections (more than 10 types, including links between words with the same root form, synonyms, and so on) are described in the paper. The algorithm then proceeds with a random walk on the graph until convergence. The evaluation is performed on a standard test set from the Penn Treebank. The results reported in the paper show a performance of 87.54 percent classification accuracy, which is very close to the upper bound corresponding to human performance (88.20 percent).

Coreference Resolution

Coreference resolution is defined as the problem of identifying relations between entity references in a text, whether they are represented by nouns or pronouns. Typical algorithms for coreference resolution attempt to identify chains of references by using rule-based systems or machine-learning classifiers. In recent work, Nicolae and Nicolae (2006) introduced a graph-based approach to coreference resolution that attempts to approximate the correct assignment of references to entities in a text by using a graph-cut algorithm.

A separate graph is created for each NIST-defined entity type, including *Person*, *Organization*, *Location*, *Facility*, and *GPE*. Next, weighted edges are drawn between the entity references, where the weights correspond to the confidence of a coreference relation. Finally, a partitioning method based on min-cut is applied on these graphs and separates the references corresponding to the same entity. When evaluated on standard benchmarks for coreference resolution, the graph-based algorithm was found to lead to state-of-the-art performance, improving considerably over previous algorithms.

Lexical Semantics

There has been growing interest in the automatic semantic analysis of text to support natural language-processing applications ranging from machine translation and information retrieval to question answering and knowledge acquisition. A significant amount of research has been carried out in this area, including work on word-sense disambiguation, semantic role labeling, textual entailment, lexical acquisition, and semantic relations. In this section, we will review several methods based on graph representations and algorithms that have been used to address various tasks in automatic semantic analysis.

Lexical Networks

One of the largest graph representations constructed to support a natural language-processing task is

perhaps the graph model proposed by Widdows and Dorow (2002) for unsupervised lexical acquisition. The goal of their work is to build semantic classes by automatically extracting from raw corpora all the elements belonging to a certain semantic category such as *fruits* or *musical instruments*.

The method starts by constructing a large graph consisting of all the nouns in a large corpus (British National Corpus, in their case), linked by the conjunction *and* or *or*. A cutoff value is used to filter out rare words, resulting in a graph of almost 100,000 nouns, linked by more than half a million edges. To identify the elements of a semantic class, first a few representative nouns are manually selected and used to form a seed set. Next, in an iterative process, the node found to have the largest number of links with the seed set in the cooccurrence graph is selected as potentially correct and thus added to the seed set. The process is repeated until no new elements can be reliably added to the seed set. Figure 3 shows a sample of a graph built to extract semantic classes.

An evaluation against 10 semantic classes from WordNet indicated an accuracy of 82 percent, which, according to the authors, was an order of magnitude better than previous work in semantic class extraction. The drawback of their method is the low coverage, given that the method is limited to those words found in a conjunction relation. However, whenever applicable, the graph representation has the ability to precisely identify the words belonging to a semantic class.

Another research area related to the work of D. Widdows and B. Dorow (2002) is the study of lexical network properties carried out by Ramon Ferrer-i-Cancho and Ricard V. Sole (2001). By building very large lexical networks of nearly half a million nodes, with more than ten million edges, constructed by linking words appearing in English sentences within a distance of at most two words, they proved that complex system properties hold on such cooccurrence networks.

Specifically, they observed a small-world effect with a relatively small number of two to three jumps required to connect any two words in the lexical network. Additionally, it has also been observed that the distribution of node degrees inside the network is scale free, which reflects the tendency of a link to be formed with an already highly connected word. Perhaps not surprisingly, the small-world and scale-free properties observed over lexical networks automatically acquired from corpora were also observed on manually constructed semantic networks such as WordNet (Sigman and Cecchi 2002, Steyvers and Tenenbaum 2005).

Semantic Similarity and Relatedness

Graph-based algorithms have also been successfully used in identifying word similarity and related-

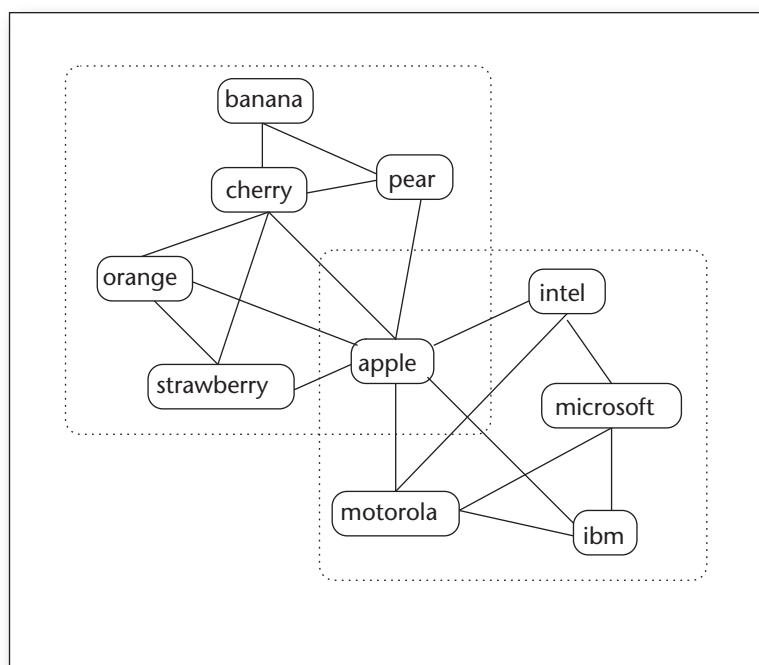


Figure 3. Lexical Network Constructed for the Extraction of Semantic Classes.

ness. A large class of methods for semantic similarity consists of metrics calculated on existing semantic networks, such as WordNet and Roget, by applying, for instance, shortest path algorithms that identify the closest semantic relation between two input concepts (Leacock, Chodorow, and Miller 1998).

More recently, an algorithm based on random walks was proposed by Hughes and Ramage (2007). Briefly, in their method, the PageRank algorithm is used to calculate the stationary distribution of the nodes in the WordNet graph, biased on each of the input words in a given word pair. Next, the divergence between these distributions is calculated, which reflects the relatedness of the two words. When evaluated on standard word-relatedness data sets, the method was found to improve significantly over previously proposed algorithms for semantic relatedness. In fact, their best performing measure came close to the upper bound represented by the interannotator agreement on these data sets.

Word-Sense Disambiguation

Another topic of interest in lexical semantics is word-sense disambiguation, defined as the problem of identifying the most appropriate meaning of a word given its context. Most of the work in this area assumes the availability of a predefined sense inventory, such as WordNet, and consists of methods that can be broadly classified as knowledge based, supervised, or semisupervised.

A graph-based method that has been successfully used for semisupervised word-sense disambigua-

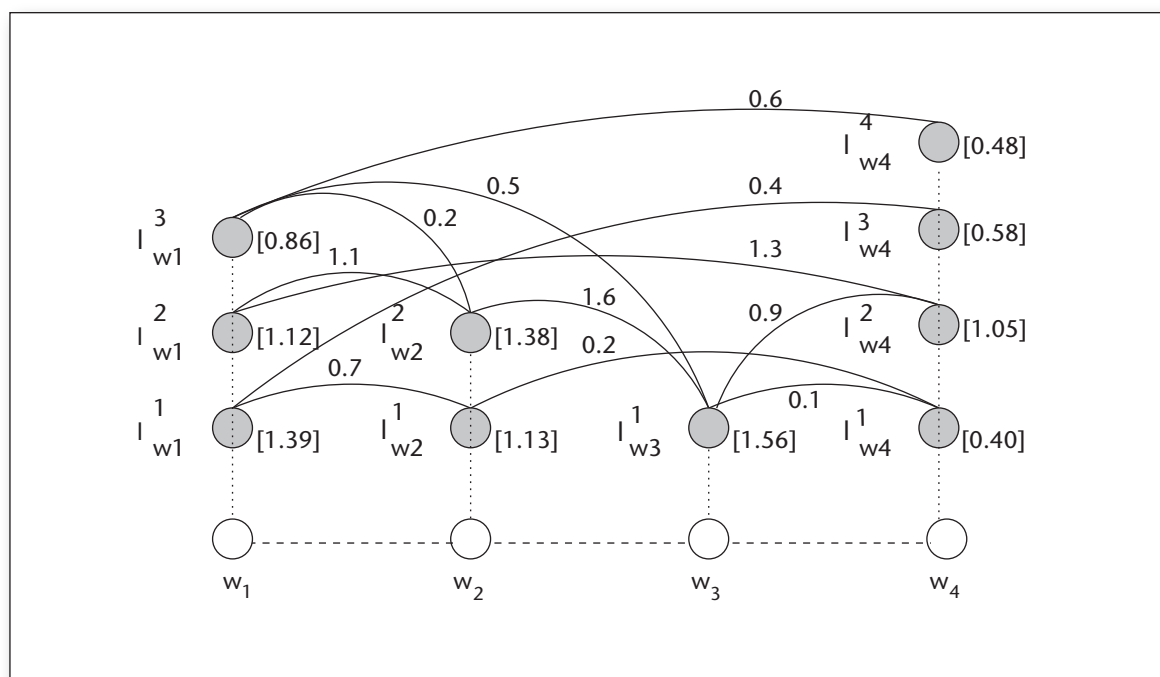


Figure 4. Graph Constructed over the Word Senses in a Sentence, to Support Automatic Word-Sense Disambiguation.

tion is the label propagation algorithm (Niu, Ji, and Tan 2005). In their work, Niu and colleagues start by constructing a graph consisting of all the labeled and unlabeled examples provided for a given ambiguous word. The word-sense examples are used as nodes in the graph, and weighted edges are drawn by using a pairwise metric of similarity. On this graph, all the known labeled examples (the seed set) are assigned with their correct labels, which are then propagated throughout the graph across the weighted links. In this way, all the nodes are assigned with a set of labels, each with a certain probability. The algorithm is repeated through convergence, with the known labeled examples being reassigned with their correct label at each iteration. In an evaluation carried out on a standard word-sense disambiguation data set, the performance of the algorithm was found to exceed the one obtained with monolingual or bilingual bootstrapping. The algorithm was also found to perform better than a Support Vector Machine (SVM) when only a few labeled examples were available.

Graph-based methods have also been used for knowledge-based word-sense disambiguation. In Mihalcea, Tarau, and Figa (2004) and Sinha and Mihalcea (2007), Mihalcea and colleagues proposed a method based on graphs constructed based on WordNet. Given an input text, a graph is built by adding all the possible senses for the words in the text, which are then connected on the basis of the semantic relations available in the WordNet lexicon (for example, synonymy, antonymy, and so on). For

instance, figure 4 shows an example of a graph constructed over a short sentence of four words.

A random walk applied on this graph results in a set of scores that reflects the “importance” of each word sense in the given text. The word senses with the highest score are consequently selected as potentially correct. An evaluation on sense-annotated data showed that this graph-based algorithm was superior to alternative knowledge-based methods that did not make use of such rich representations of word-sense relationships.

In follow-up work, Mihalcea developed a more general graph-based method that did not require the availability of semantic relations such as those defined in WordNet. Instead, she used derived weighted edges determined by using a measure of lexical similarity among word-sense definitions (Mihalcea 2005), which brought generality, as the method is not restricted to semantic networks such as WordNet but can be used on any electronic dictionaries.

Along similar lines with Mihalcea, Tarau, and Figa (2004), Roberto Navigli and Mirella Lapata (2007) carried out a comparative evaluation of several graph-connectivity algorithms applied on word-sense graphs derived from WordNet. They found that the best word-sense disambiguation accuracy is achieved by using a closeness measure, which was found superior to other graph-centrality algorithms such as in-degree, PageRank, and betweenness.

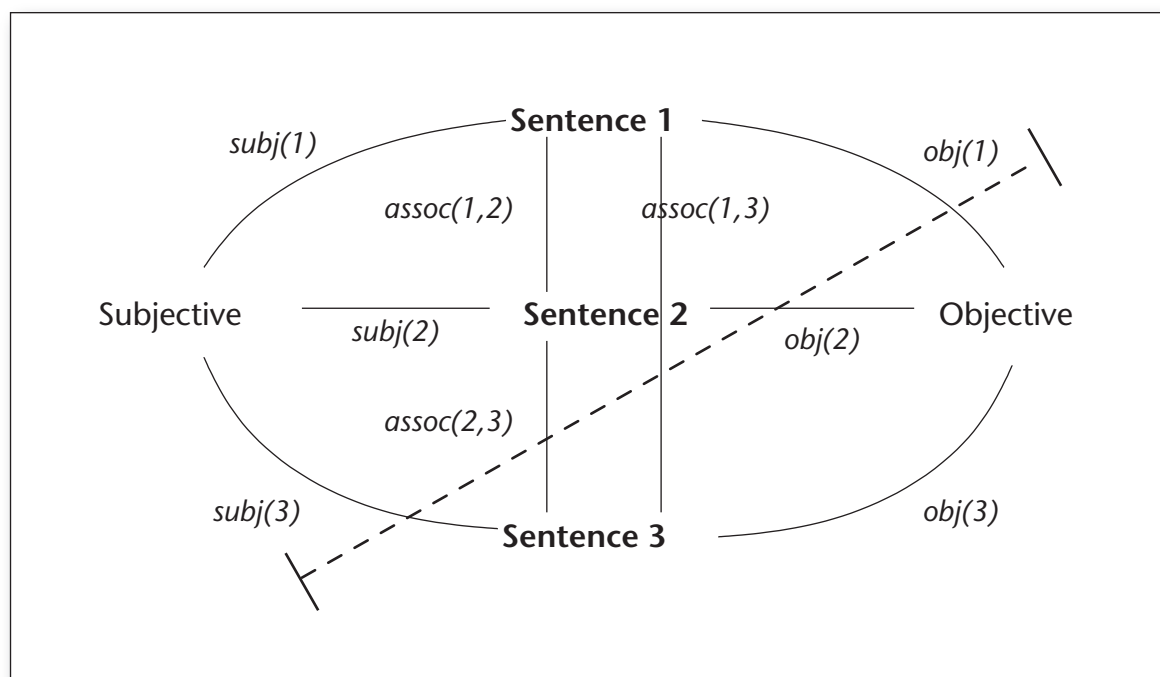


Figure 5. Subjectivity Classification Using a Min-cut Algorithm.

The dotted line represents the split between subjective and objective sentences, as obtained with the min-cut algorithm.

Sentiment and Subjectivity

Sentiment and subjectivity analysis, an area related to both semantics and pragmatics, has received a lot of attention from the research community. A method based on graphs has been proposed by Bo Pang and Lillian Lee (2004), where they show that a min-cut graph-based algorithm can be effectively applied to build subjective extracts of movie reviews.

First, they construct a graph by adding all the sentences in a review as nodes and by drawing edges based on sentence proximity. Each node in the graph is initially assigned with a score indicating the probability of the corresponding sentence being subjective or objective, based on an estimate provided by a supervised subjectivity classifier. A min-cut algorithm is then applied on the graph and used to separate the subjective sentences from the objective ones. Figure 5 illustrates the graph constructed over the sentences in a text on which the min-cut algorithm is applied to identify and extract the subjective sentences.

The precision of this graph-based subjectivity classifier was found to be better than the labeling obtained with the initial supervised classifier. Moreover, a polarity classifier relying on the min-cut subjective extracts was found to be more accurate than one applied on entire reviews.

Recent research on sentiment and subjectivity analysis has also considered the relation between word senses and subjectivity (Wiebe and Mihalcea 2006). In work targeting the assignment of subjectivity and polarity labels to WordNet senses, Esuli

and Sebastiani (2007) applied a biased PageRank algorithm on the entire WordNet graph. Similar to some extent to the label propagation method, their random-walk algorithm was seeded with nodes labeled for subjectivity and polarity. When compared to a simpler classification method, their random-walk algorithm was found to result in more accurate annotations of subjectivity and polarity of word senses.

Other Applications

A number of other natural language-processing applications such as text summarization, passage retrieval, and keyword extraction are amenable to graph-based techniques.

Summarization

One of the first graph-based methods for summarization was introduced by James Allan and colleagues (Salton et al. 1994, 1997). In it, they represented articles from the Funk and Wagnalls encyclopedia as graphs in which each node corresponds to a paragraph and lexically similar paragraphs are linked. A summary is then produced by starting at the first paragraph of a document and following paths defined by different algorithms that cover as much of the contents of the graph as possible.

Erkan and Radev (2004) and Mihalcea and Tarau (2004) take the idea of graph-based summarization further by introducing the concept of lexical centrality. Lexical centrality is a measure of impor-

SNo	ID	Text
1	d1s1	Iraqi Vice President Taha Yassin Ramadan announced today, Sunday, that Iraq refuses to back down from its decision to stop cooperating with disarmament inspectors before its demands are met.
2	d2s1	Iraqi Vice president Taha Yassin Ramadan announced today, Thursday, that Iraq rejects cooperating with the United Nations except on the issue of lifting the blockade imposed upon it since the year 1990.
3	d2s2	Ramadan told reporters in Baghdad that "Iraq cannot deal positively with whoever represents the Security Council unless there was a clear stance on the issue of lifting the blockade off of it.
4	d2s3	Baghdad had decided late last October to completely cease cooperating with the inspectors of the United Nations Special Commission (UNSCOM), in charge of disarming Iraq's weapons, and whose work became very limited since the fifth of August, and announced it will not resume its cooperation with the Commission even if it were subjected to a military operation.
5	d3s1	The Russian Foreign Minister, Igor Ivanov, warned today, Wednesday against using force against Iraq, which will destroy, according to him, seven years of difficult diplomatic work and will complicate the regional situation in the area.
6	d3s2	Ivanov contended that carrying out air strikes against Iraq, who refuses to cooperate with the United Nations inspectors, "will end the tremendous work achieved by the international group during the past seven years and will complicate the situation in the region."
7	d3s3	Nevertheless, Ivanov stressed that Baghdad must resume working with the Special Commission in charge of disarming the Iraqi weapons of mass destruction (UNSCOM).
8	d4s1	The Special Representative of the United Nations Secretary-General in Baghdad, Prakash Shah, announced today, Wednesday, after meeting with the Iraqi Deputy Prime Minister Tariq Aziz, that Iraq refuses to back down from its decision to cut off cooperation with the disarmament inspectors.
9	d5s1	British Prime Minister Tony Blair said today, Sunday, that the crisis between the international community and Iraq "did not end" and that Britain is still "ready, prepared, and able to strike Iraq."
10	d5s2	In a gathering with the press held at the Prime Minister's office, Blair contended that the crisis with Iraq "will not end until Iraq has absolutely and unconditionally respected its commitments" towards the United Nations.
11	d5s3	A spokesman for Tony Blair had indicated that the British Prime Minister gave permission to British Air Force Tornado planes stationed in Kuwait to join the aerial bombardment against Iraq.

Figure 6. A Cluster of 11 Related Sentences.

tance (centrality) of nodes in a graph formed by linking lexically related sentences or documents. A random walk is then executed on the graph and the nodes that are visited the most frequently are selected as the summary of the input graph (which, in most cases, consists of information from multiple documents). One should note however,

that in order to avoid nodes with duplicate or near duplicate content, the final decision about including a node in the summary also depends on its maximal marginal relevance as defined in Carbonell and Goldstein (1998). Erkan and Radev (2004) also build on top of earlier summarization technology, namely the first web-accessible news

	1	2	3	4	5	6	7	8	9	10	11
1	1.00	0.45	0.02	0.17	0.03	0.22	0.03	0.28	0.06	0.06	0.00
2	0.45	1.00	0.16	0.27	0.03	0.19	0.03	0.21	0.03	0.15	0.00
3	0.02	0.16	1.00	0.03	0.00	0.01	0.03	0.04	0.00	0.01	0.00
4	0.17	0.27	0.03	1.00	0.01	0.16	0.28	0.17	0.00	0.09	0.01
5	0.03	0.03	0.00	0.01	1.00	0.29	0.05	0.15	0.20	0.04	0.18
6	0.22	0.19	0.01	0.16	0.29	1.00	0.05	0.29	0.04	0.20	0.03
7	0.03	0.03	0.03	0.28	0.05	0.05	1.00	0.06	0.00	0.00	0.01
8	0.28	0.21	0.04	0.17	0.15	0.29	0.06	1.00	0.25	0.20	0.17
9	0.06	0.03	0.00	0.00	0.20	0.04	0.00	0.25	1.00	0.26	0.38
10	0.06	0.15	0.01	0.09	0.04	0.20	0.00	0.20	0.26	1.00	0.12
11	0.00	0.00	0.00	0.01	0.18	0.03	0.01	0.17	0.38	0.12	1.00

Figure 7. Cosine Similarities across All Sentence Pairs in a Cluster of 11 Sentences.

summarization system, NewsInEssence (Radev et al. 2001).

An example from Erkan and Radev (2004) is shown in figure 6. The input consists of 11 sentences from several news stories on related topics. Figure 7 shows the cosine similarities of all pairs of sentences while figure 8 shows the distribution of cosines.

It is important to realize that the cosine matrix hides in itself an infinite number of graphics for each value of a cosine cutoff, t . This can be seen in figures 9 and 10. For example, if one lowers the threshold too much, the graph is almost fully connected. Conversely, raising the threshold eventually turns the graph into a set of disconnected components. The random walk is typically performed at the value of t at which approximately half of the node pairs are connected through edges. Figure 11 shows the LexRank Java interface as used for text summarization.

Semisupervised Passage Retrieval

Jahna Otterbacher and colleagues (2005) take Erkan and Radev (2004) one step further by introducing the concept of a biased random walk to address the problem of question-focused passage retrieval. In that problem the user issues a query in the form of a natural language question and expects to get a set of passages from the input documents that contain the answer to that question. The biased random walk is performed on a graph that is already seeded with known positive and negative examples. Then, each node is labeled in

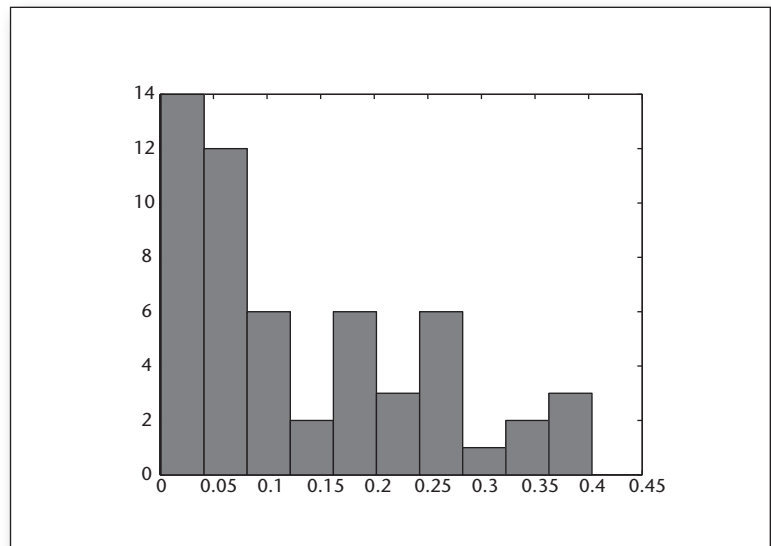


Figure 8. LexRank Cosine Histogram.

proportion to the percentage of times a random walk on the graph ends at that node. Given the presence of the initially labeled nodes, the nodes with the highest score eventually are the ones that are both similar to the seed nodes and are central to the document set. In other words, they are chosen eventually as the answer set by a mixture model that takes into account the known seeds (positive or negative) and the centrality score as in the previous section. The graph consists of both sen-



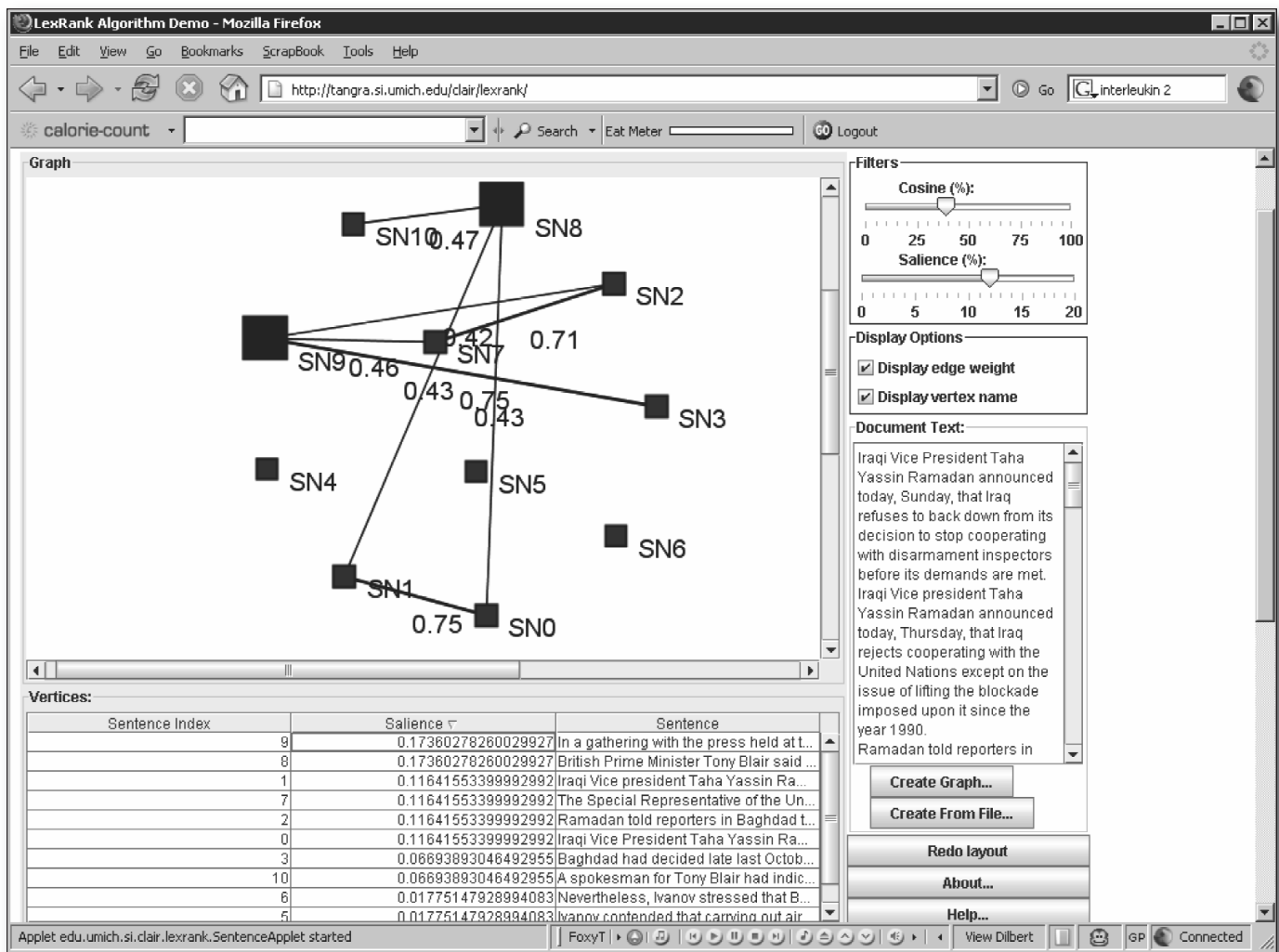


Figure 11. LexRank Interface.

tences (paragraphs) and features (content words that appear in these sentences). The graph is bipartite as a sentence can only link to a feature and vice versa.

In the example shown in figure 12, the top right-hand node is initially labeled as positive (dark) whereas the bottom right-hand node is labeled as negative (clear node). During the labeling (performed using the method of relaxations), each node's shadedness changes until the process converges. At the end, darker nodes are returned as relevant to the user question. Note that some of them contain no words in common with the original query.

Keyword Extraction

The task of a keyword extraction application is to automatically identify in a text a set of terms that best describe the document. Such keywords may

constitute useful entries for building an automatic index for a document collection, can be used to classify a text, or may serve as a concise summary for a given document. A system for automatic identification of important terms can also be used for the problem of terminology extraction and construction of domain-specific dictionaries. A random-walk algorithm for keyword extraction has been proposed in Mihalcea and Tarau (2004), where a graph is constructed on an input text by adding all the words in the text as nodes in the graph and connecting them by a cooccurrence relation constrained by the distance between the words.

Figure 13 shows a sample graph built for a short scientific text. A random-walk run on such a graph of cooccurrences leads to a ranking over the importance of the words in the text; in a postprocessing phase, words selected as important by the ranking algorithm and found next to each other in

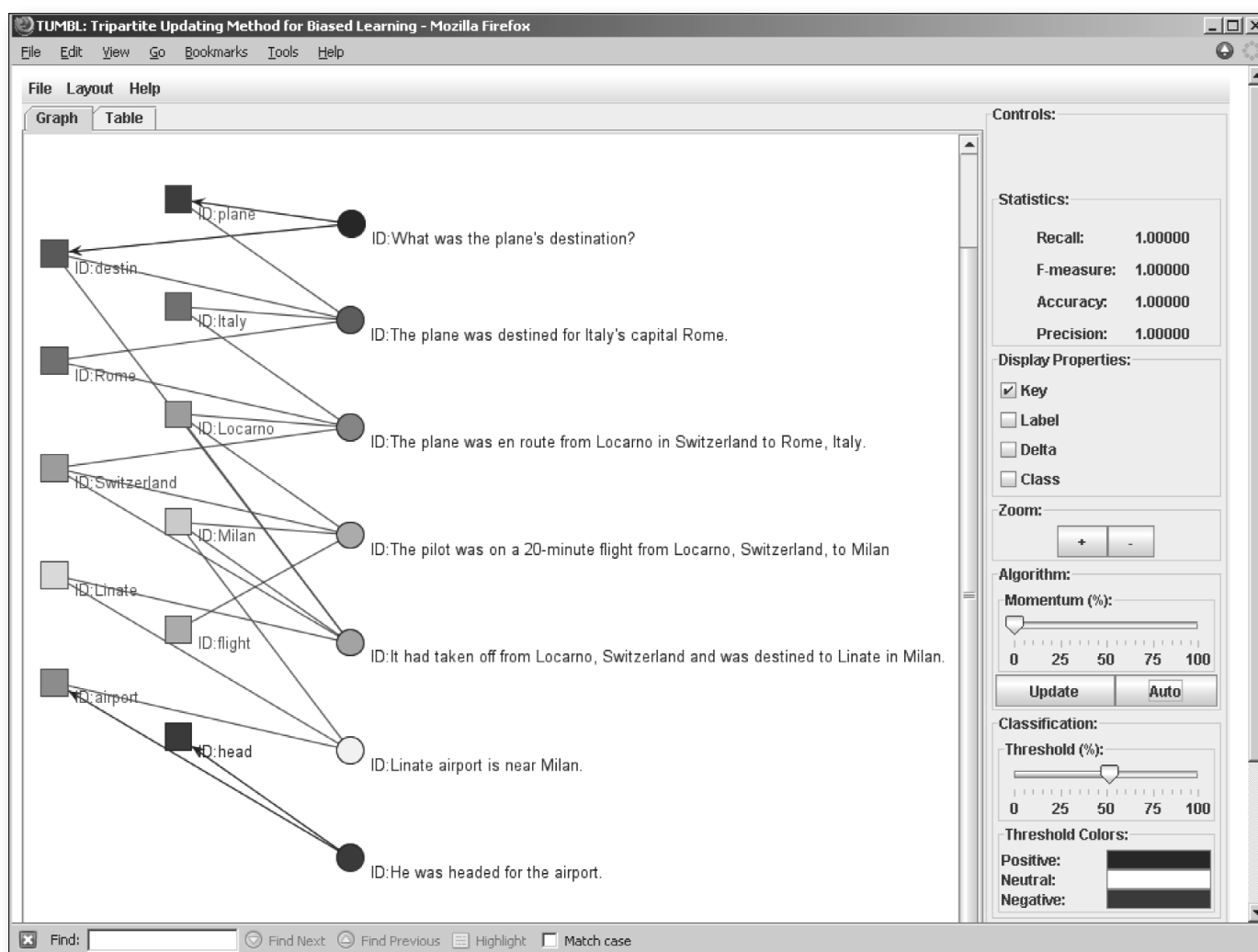


Figure 12. Biased LexRank as Used for Semisupervised Passage Retrieval.

the text are collapsed into a single phrase. Interestingly, experiments comparing this ranking with the traditional *tf.idf* showed that the scores assigned by the random walk can differ significantly. In fact, evaluations on a data set of scientific abstracts showed that the random-walk method is superior to the *tf.idf* method for keyword extraction, and it also improved over previously published state-of-the-art supervised methods for keyword extraction.

Further Reading

A large bibliography appears on the first author's web site and also on www.textgraphs.org.

Acknowledgments

This paper is partially based on work done previously by the two authors. That work was partially funded by NSF grants IIS 0534323 "Collaborative

Research: BlogCenter—Infrastructure for Collecting, Mining and Accessing Blogs," IIS 0329043 "Probabilistic and Link-Based Methods for Exploiting Very Large Textual Repositories," BCS 0527513 "DHB: The Dynamics of Political Representation and Political Rhetoric," and by NIH grants R01 LM008106 "Representing and Acquiring Knowledge of Genome Regulation" and U54 DA021519 "National Center for Integrative Bioinformatics," all to Dragomir Radev. This work was also supported in part by research grant no. 003594 on "Graph-Based Natural Language Processing" from the Texas Advanced Research Program and by a Google grant on "Finding Important Information in Unstructured Text," both of them awarded to Rada Mihalcea.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the

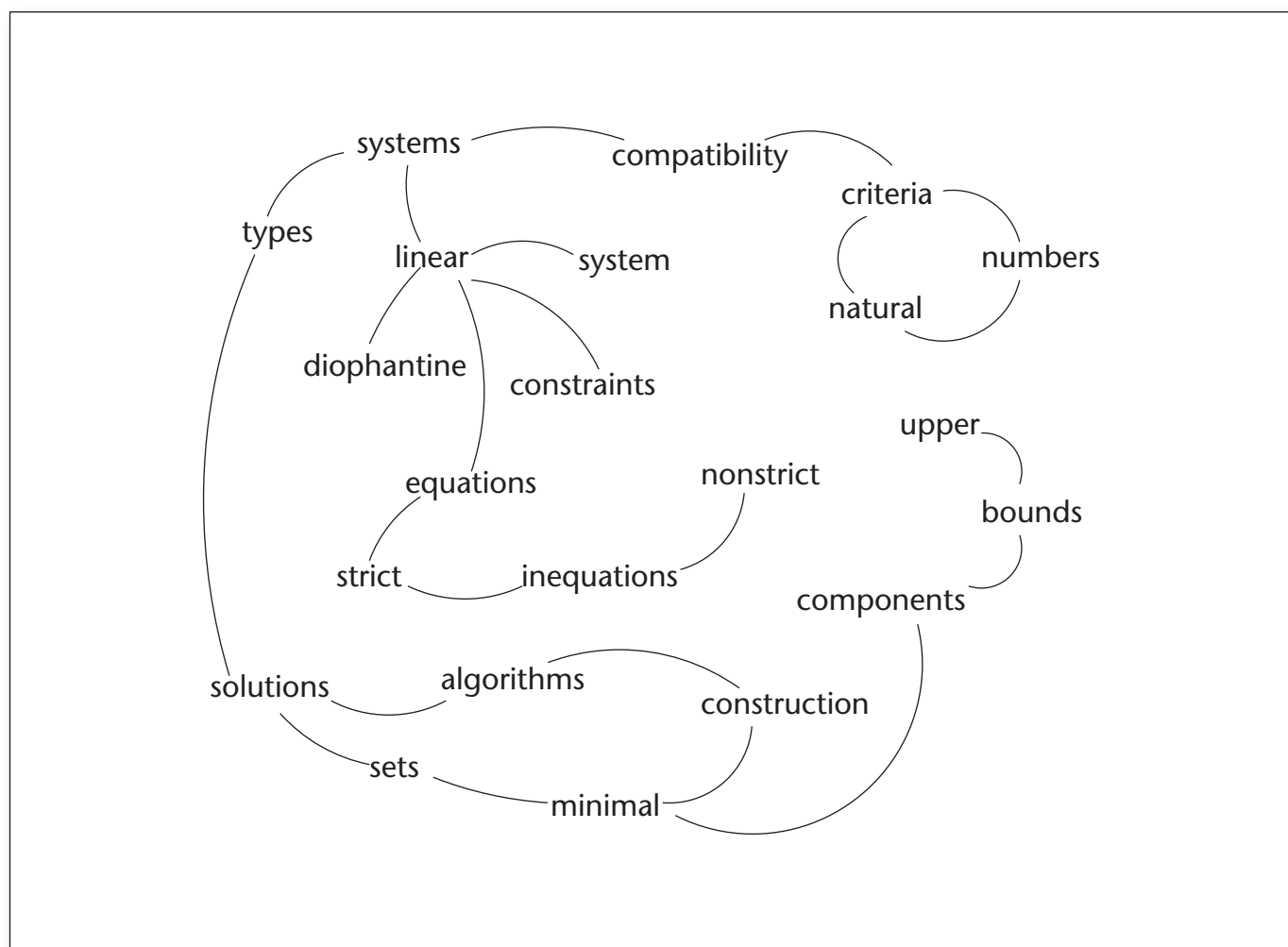


Figure 13. Sample Graph Built for Keyword Extraction.

views of the National Science Foundation or the other sponsors.

References

- Caldeira, S. M. G.; Lobão, T. C.; Andrade, R. F. S.; Neme, A.; and Miranda, J. G. V. 2006. The Network of Concepts in Written Texts. *European Physical Journal B*, 49(4): 523–529.
- Carbonell, J. G., and Goldstein, J. 1998. The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 335–336. New York: Association for Computing Machinery.
- Chu, Y. J., and Liu, T. H. 1965. On the Shortest Arborescence of a Directed Graph. *Science Sinica* 14: 1396–1400.
- Edmonds, J. 1967. Optimum Branchings. *Journal of Research of the National Bureau of Standards* 71b: 233–240.
- Erkan, G., and Radev, D. 2004. The University of Michigan at Duc 2004. Paper presented at the Document Understanding Conference (Duc), Boston, Massachusetts, May.
- Esuli, A., and Sebastiani, F. 2007. Pageranking WordNet Synsets: An Application to Opinion Mining. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics*. East Stroudsburg, PA: Association for Computational Linguistics.
- Ferrer-I-Cancho, R., and Sole, R. V. 2001. The Small World of Human Language. *Proceedings of the Royal Society of London, Series B, Biological Sciences*, 268(1482) (November): 2261–2265.
- Hughes, T., and Ramage, D. 2007. Lexical Semantic Relatedness with Random Graph Walks. Paper presented at the Conference on Empirical Methods in Natural Language Processing (EMNLP 2007), Prague, Czech Republic.
- Leacock, C.; Chodorow, M.; and Miller, G. A. 1998. Using Corpus Statistics and WordNet Relations for Sense Identification. *Computational Linguistics* 24(1): 147–165.
- Masucci, A. P., and Rodgers, G. J. 2006. Network Properties of Written Human Language. *Physical Review E*, 74 (026102), August 2.
- McDonald, R.; Pereira, F.; Ribarov, K.; and Hajic, J. 2005. Nonprojective Dependency Parsing Using Spanning Tree Algorithms. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in*

- Natural Language Processing*. East Stroudsburg, PA: Association for Computational Linguistics.
- Mihalcea, R. 2005. Unsupervised Large-Vocabulary Word Sense Disambiguation with Graph-Based Algorithms for Sequence Data Labeling. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*. East Stroudsburg, PA: Association for Computational Linguistics.
- Mihalcea, R., and Tarau, P. 2004. Textrank: Bringing Order into Texts. Paper presented at the Conference on Empirical Methods in Natural Language Processing (EMNLP 2004), 404–411, Barcelona, Spain, July. East Stroudsburg, PA: Association for Computational Linguistics.
- Mihalcea, R.; Tarau, P.; and Figa, E. 2004. Pagerank on Semantic Networks, with Application to Word Sense Disambiguation. Paper presented at the 20th International Conference on Computational Linguistics (Coling 2004), Geneva, Switzerland.
- Navigli, R., and Lapata, M. 2007. Graph Connectivity Measures for Unsupervised Word Sense Disambiguation. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*. Menlo Park, CA: AAAI Press.
- Nicolae, C., and Nicolae, G. 2006. Bestcut: a Graph Algorithm for Coreference Resolution. Paper presented at the 2006 Conference on Empirical Methods in Natural Language Processing, Sydney, Australia, July.
- Niu, Z. Y.; Ji, D. H.; and Tan, C. L. 2005. Word Sense Disambiguation Using Label Propagation Based Semi-Supervised Learning. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*. East Stroudsburg, PA: Association for Computational Linguistics.
- Otterbacher, J.; Erkan, G.; and Radev, D. 2005. Using Random Walks for Question-Focused Sentence Retrieval. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, 915–922. East Stroudsburg, PA: Association for Computational Linguistics.
- Pang, B., and Lee, L. 2004. A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04)*, 271–278. East Stroudsburg, PA: Association for Computational Linguistics.
- Radev, D. 2004. Weakly Supervised Graph-Based Methods for Classification. Technical Report CSE-Tr-500-04, University of Michigan. Department of Electrical Engineering and Computer Science. Ann Arbor, MI.
- Radev, D.; Blair-Goldensohn, S.; Zhang, Z., and Raghavan, R. 2001. Newsinessence: A System for Domain-Independent, Real-Time News Clustering and Multi-Documents Summarization. Paper presented at the Human Language Technology Conference (HLT 2001), San Diego, CA, March.
- Salton, G.; Allan, J.; Buckley, C.; and Singhal, A. 1994. Automatic Analysis, Theme Generation, and Summarization of Machine-Readable Texts. *Science* 264(5164): 1421–1426.
- Salton, G.; Singhal, A.; Mitra, M.; and Buckley, C. 1997. Automatic Text Structuring and Summarization. *Information Processing and Management* 33(2) (March): 193–207.
- Sigman, M., and Cecchi G.. 2002. Global Organization of the Wordnet Lexicon. *Proceedings of the National Academy of Sciences of the United States of America*, 99(3) (February 5): 1742–1747.
- Sinha, R., and Mihalcea, R. 2007. Unsupervised Graph-Based Word Sense Disambiguation Using Measures of Word Semantic Similarity. In *Proceedings of the IEEE International Conference on Semantic Computing (ICSC 2007)*. New York: Institute of Electrical and Electronics Engineers.
- Steyvers, M., and Tenenbaum, J. B. 2005. Graph Theoretic Analyses of Semantic Networks: Small Worlds In Semantic Networks. *Cognitive Science* 29: 41–78.
- Toutanova, K.; Manning, C.; and Ng, A. 2004. Learning Random Walk Models for Inducing Word Dependency Distributions. In *Proceedings of the Twenty-First International Conference on Machine Learning*, 103. New York: Association for Computing Machinery.
- Widdows, D., and Dorow, B. 2002. A Graph Model for Unsupervised Lexical Acquisition. In *Proceedings of the 19th International Conference on Computational Linguistics*. East Stroudsburg, PA: Association for Computational Linguistics.
- Wiebe, J., and Mihalcea, R. 2006. Word Sense and Subjectivity. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. East Stroudsburg, PA: Association for Computational Linguistics.
- Zhu, X., and Ghahramani, Z. 2002. Learning from Labeled and Unlabeled Data with Label Propagation. Technical Report CMU-Cald-02-107, Carnegie Mellon University, Pittsburgh, PA.
- Zhu, X., and Lafferty, J. 2005. Harmonic Mixtures: Combining Mixture Models and Graph-Based Methods for Inductive and Scalable Semi-Supervised Learning. In *Proceedings of the Twenty-Second International Conference on Machine Learning (ICML '05)*. New York: ACM Press.



Dragomir R. Radev is an associate professor at the School of Information and the Department of Electrical Engineering and Computer Science at the University of Michigan. Radev received a Ph.D. in computer science from Columbia University in 1999. He is currently secretary of the Association for Computational Linguistics and program chair of the North American Computational Linguistics Olympiad and also coach of the US national linguistics teams. His areas of research include natural language processing, information retrieval, and network theory.



Rada Mihalcea is an associate professor of computer science at the University of North Texas. Her research interests are in lexical semantics, graph-based algorithms for natural language processing, minimally supervised natural language learning, and multilingual natural language processing. She has published numerous articles in books, journals, and proceedings in these and related areas. Mihalcea is the recipient of a National Science Foundation CAREER award.