ASKNet: Automated Semantic Knowledge Network

Brian Harrington and Stephen Clark
Oxford University Computing Laboratory
Wolfson Building, Parks Road
Oxford, OX1 3QD, UK
{brian.harrington,stephen.clark}@comlab.ox.ac.uk

Abstract

The ASKNet system is an attempt to automatically generate large scale semantic knowledge networks from natural language text. State-of-the-art language processing tools, including parsers and semantic analysers, are used to turn input sentences into fragments of semantic network. These network fragments are combined using spreading activation-based algorithms which utilise both lexical and semantic information. The emphasis of the system is on wide-coverage and speed of construction. In this paper we show how a network consisting of over 1.5 million nodes and 3.5 million edges, more than twice as large as any network currently available, can be created in less than 3 days. We believe that the methods proposed here will enable the construction of semantic networks on a scale never seen before, and in doing so reduce the knowledge acquisition bottleneck for AI.

Introduction

The ASKNet (Automated Semantic Knowledge Network) system automatically extracts knowledge from natural language text and, using a combination of Natural Language Processing (NLP) tools and spreading activation theory, builds a semantic network to represent that knowledge. Semantic resources of the type the ASKNet system aims to create already exist; however, they are mostly constructed manually which severely limits their coverage and scale. The few attempts at automatically creating such resources, for example Microsoft's MindNet (Richardson *et al.* 1998), restrict the information they gather in order to limit computational complexity and maintain high precision. ASKNet aims to construct a very large network in a short amount of time, trading some precision for speed of construction. ASKNet can process large and varied text corpora to extract a wide variety of information, and we believe that the methods described here will allow the construction of semantic networks on a scale never seen before.

ASKNet uses a wide-coverage parser and semantic analyser which have been trained on newspaper text (as opposed to more structured resources such as dictionary or encyclopaedia text). Unlike typical resources of its kind, ASKNet does not limit the set of possible relations it can extract. The resource created is a semantic knowledge network which links objects, entities and complex concepts by named and directed relations. The nested structure of the network allows objects and relations to be grouped together to form a complex object. This allows the network to express complex concepts without requiring a rigid structure.

The semantic network is created by joining document level networks, which are in turn created by joining sentence level network fragments. Each new sentence encountered is seen as an update to the world knowledge base represented by the entire knowledge network. Spreading activation (Collins & Loftus 1975) is used to determine the semantic connections between entities and concepts. This is modelled on the workings of the human brain, and has the advantage of being localised, so that the computational complexity of any action does not grow with the size of the network.

Motivation

The potential of a large scale semantic knowledge base can be seen by the number of projects currently underway to build one. Projects such as Concept Net (Liu & Singh 2004) and Cyc (Lenat 1995) have spent decades of time and thousands of man-hours manually constructing semantic knowledge networks. However, manual construction severely limits the coverage and scale that can be achieved. After more than a decade of work, the largest semantic networks have on the order of 1.5-2.5 million relations connecting 200,000-300,000 nodes (Matuszek *et al.* 2006).

These networks have been applied to tasks such as question answering (Curtis, Matthews, & Baxter 2005) and predictive text entry (Stocky, Faaborg, & Lieberman 2004). However, many tasks either require a domain specific knowledge base, which needs to be created quickly for the task at hand, or require much wider coverage than is possible to achieve in manually created networks. Automatic generation allows us to acquire information from existing data to create new semantic resources very quickly, and to create resources which are many orders of magnitude larger.

One existing system which automatically creates semantic networks is MindNet (Dolan *et al.* 1993). MindNet uses a natural language parser to extract pre-defined relations from dictionary definitions. To illustrate the time difference for automated construction over manual creation, the MindNet

network of over 150,000 words, connected by over 700,000 relations (roughly half the size of the ConceptNet or Cyc networks), can be created in a matter of hours on a standard personal computer (Richardson *et al.* 1998). The difference between the ASKNet system and MindNet is that MindNet is limited to building networks with a small, pre-defined set of relations, and limited to extracting knowledge from well-formed data such as dictionaries. In contrast, ASKNet extracts the relations from the text itself using a natural language parser trained on newspaper text. ASKNet also integrates information from multiple sources by mapping together nodes which refer to the same real-world entity; a task which is not attempted by MindNet. This allows ASKNet to accommodate a much wider variety of information, use more varied sources of input, and extract more information than any similar system currently in development.

The Espresso system (Pantel & Pennacchiotti 2006) attempts to harvest semantic relations from natural language text by building word patterns which signify a specific relation (e.g., "X consists of Y" for the `part_of(Y,X)` relation), and searching large corpora for text which fits the patterns. The building of patterns is weakly-supervised, and each new relation the system extracts must be expressly chosen by a human user. Unlike ASKNet, Espresso only extracts binary relations, and does not build complex node structures or perform any information integration.

Schubert and Tong (2003) have also developed a system for automatically acquiring knowledge from text. However, they attempt to gain "possibilistic propositions" (e.g., *"A person can believe a proposition"*) rather than extracting direct knowledge from the text. Furthermore, they only extract information from a small treebank corpus rather than raw text. ASKNet can extract information from raw text because of its use of a wide coverage parser. This allows us to use the vast quantities of readily available English text to create networks, instead of comparatively small structured corpora.

Semantic resources created automatically will contain more errors than their manually created counterparts. However, for many tasks, the great decrease in time and labour required to build a network, combined with the ability to create extremely large networks, will make up for any decrease in accuracy (Dolan *et al.* 1993).

Parsing & Semantic Analysis

In order to create a very large network with a wide variety of relation types it is essential to have a parser which is extremely efficient and robust, and has wide coverage. It is only recently that such parsers have become available. ASKNet uses the Clark and Curran (2004b) parser, based on the linguistical formalism Combinatory Categorial Grammar (CCG) (Steedman 2000). CCG is a *lexicalised* grammar formalism, which means that it associates with each word in a sentence an elementary syntactic structure. In CCG's case, these structures are *lexical categories* which encode subcategorisation information.

The innovation in the CCG parser is to combine a linguistically-motivated grammar formalism with an efficient and robust parser. The robustness arises from the fact that the grammar is extracted from a treebank consisting of real-world text: 40,000 sentences of WSJ text manually annotated with CCG derivations (Hockenmaier 2003). The efficiency comes from the fact that the lexical categories can be assigned to words accurately using finite-state tagging techniques, which removes much of the practical complexity from the parsing (Clark & Curran 2004a).

A named entity (NER) recognition tool is also built into the parser. The tool treats the NER problem as a label sequencing problem, assigning tags to words in a sentence indicating whether the word is part of a named entity and the entity type. The tool handles the following semantic categories: `person, organisation, date, time, location and monetary amount`. The accuracy of the NER tool ranges roughly from 85 to 90%, depending on the data set and the entity type (Curran & Clark 2003).

Once the data has been parsed, ASKNet uses the semantic analysis tool Boxer (Bos *et al.* 2004) to convert the parsed output into a series of first order logic predicates. The logical theory used for the representation is Discourse Representation Theory (DRT) (Kamp & Reyle 1993), which originated in the formal semantics literature.

The output of Boxer is a Prolog style discourse representation structure with variables assigned to objects and first order predicates representing relations between those objects. Boxer captures the underlying semantic relations in a sentence such as "agent" and "patient" to construct labelled and directed relations. Propositions are assigned their own recursively defined sub-structures. (See Figure 1 for an example.)

A simple, low-coverage pronoun resolution scheme is also implemented which attempts to assign appropriate object variables to pronouns. ASKNet can efficiently translate Boxer's semantic output for each sentence into one or more semantic network fragments.

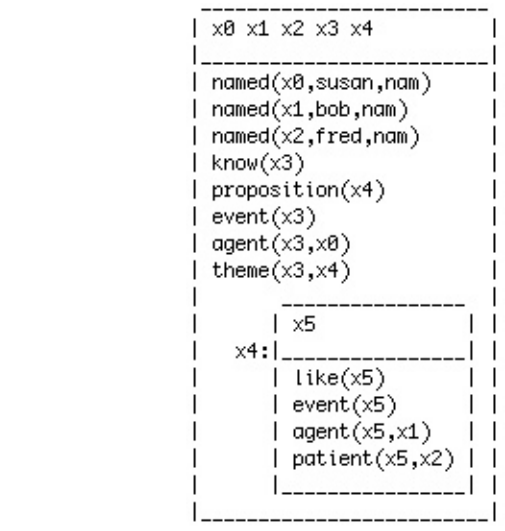

Figure 1: Example Boxer output for the sentence "Susan knows that Bob likes Fred"

The ASKNet framework has been designed to be flexible, and could easily be adapted to other NLP tools. However, we have chosen to use the Clark and Curran parser and Boxer because of their efficiency, coverage, robustness, and the relative sophistiation of their output.

The Network

The semantic networks created by ASKNet consist of object nodes linked by directed labelled relations. The objects and relations roughly correspond to the entity variables and first order relations created by Boxer. In particular, this means that the relations are not bound to a particular set of types, and can be given any label appearing in the Boxer output. This vastly increases the expressiveness of the network.

Another important feature of the network is its nesting structure. ASKNet allows nodes and relations to be combined to form complex nodes which can represent larger and more abstract concepts. These complex nodes can be combined with further relations to represent even more complex concepts. An example is given in Figure 2.

The nested structure of the network allows for the expression of complex concepts without having to resort to a rigidly defined structure such as the hierarchical structure used by WordNet (Fellbaum 1998). While a pre-defined structure provides a simple and effective framework for network creation, it also limits which nodes may be linked, thereby decreasing the expressiveness of the network.

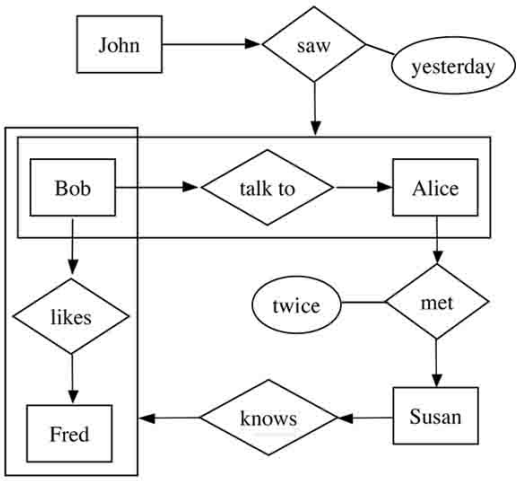

Figure 2: A simplified Semantic Network created from the sentences "John saw Bob talk to Alice yesterday. Alice met Susan twice. Susan knows that Bob likes Fred."

Each relation in the network has a weight which represents the confidence of the network in the "real world" existence of the relation and also its salience. Factors such as the confidence of the initial sentence parse, the source of the information, how recently the information has been encountered and the number of different sources verifying the information could all affect the weight given to a relation. For example, the program currently sets the weights of the relations based on the number of times that the relation has been seen, and also increases the weight of relations gathered from headlines over those gathered from the body of a document.

Spreading Activation

Concepts and ideas in the human brain have been shown to be semantically linked (Meyer & Schvaneveldt 1971) so that thinking about (or firing) one concept primes other related concepts making them more likely to fire in the near future. This is the idea behind the ASKNet network. Firing a node sends activation out to all of the nodes semantically connected to that node, which can either cause them to fire (analogous to one concept "triggering" thoughts of a related concept) or cause them to store activation, making them more likely to fire in the future (analogous to "priming" in the human brain). The weight of the relations connecting two nodes dictates the amount of activation that is transferred after a node fires.

By firing one or more nodes and analysing the way in which activation spreads through the network, we can determine the semantic distance between various entities and concepts. This allows us to determine how closely related two entities or concepts are even if they are not directly linked. When a test node is fired, we can measure how closely related any sample node is to our test node simply by measuring the total activation that came to the sample node through all of its links.

Spreading activation is an efficient means of determining semantic distance in the network because it is localised. Most search based algorithms would require traversal of the entire network before calculating the total degree of connectivity. Spreading activation only accesses a small number of nodes in a localised area, and thus the amount of time required for a single firing depends only on the amount of initial activation provided, and the amount of activation that is lost with each transfer. Hence, as the network grows in size, the time complexity of the firing algorithm does not grow with it.

Information Integration

A key problem when building the network is deciding which nodes are co-referent. This information integration allows the network to become more connected and provides a great deal of the potential power of the network. Without this step the network would simply be a series of small unconnected network fragments. However, this is a difficult task since it often requires semantic as well as lexical information.

The Update Algorithm

The update algorithm is the process by which ASKNet merges sentence level networks into document level networks, and merges document level networks into the overall knowledge network. The basic premise behind the algorithm is that when a smaller *update* network is combined with the larger knowledge network, some of the nodes in the update network may refer to the same real world entities as existing nodes in the knowledge network. Potential

node pair matches are initially scored based on lexical information, and then spreading activation is used to gradually refine the scores. Scores above a certain threshold indicate that the two nodes refer to the same real world entity and should be mapped together.

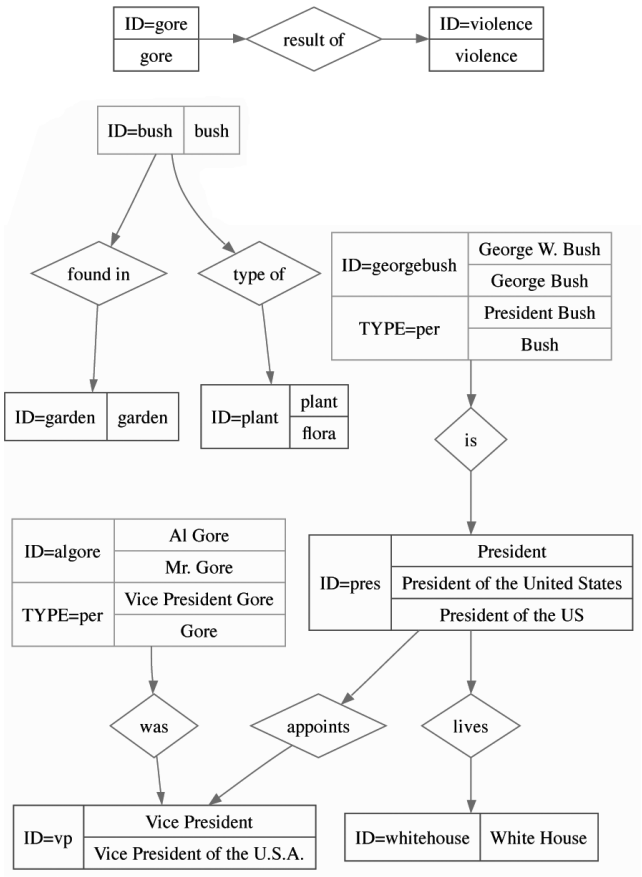

Figure 3: An example knowledge network containing information about United States politics, gardening and violence

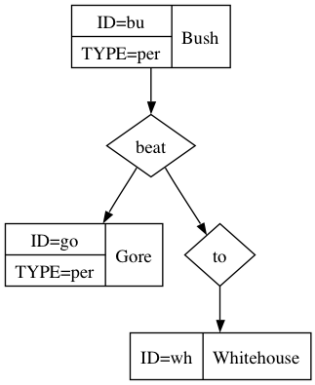

Figure 4: An example update network created from the sentence "Bush beat Gore to the Whitehouse"

In order to understand the operation of the update algorithm, we will walk through a single iteration of a simplified example, updating the knowledge network in Figure 3 with the network fragment in Figure 4. All nodes will be referred to by their *node ID*; thus `go` refers to the node with the label "`Gore`" in the update network, while `algore` refers to the node with the label "`Al Gore`" in the knowledge network.

Initially, all potential node pairs are scored based on named entity type and label similarity. The label similarity score is based on the percentage of labels having an edit distance below a set threshold, with label order being disregarded. The named entity type similarity score is a set value which can be either added to or subtracted from the total, and is only calculated if both nodes have an assigned type. These scores are entered into the *similarity matrix* given in Table 1.

	georgebush	bush	algore	gore	whitehouse
bu	0.5	0.7			
go			0.5	0.7	
wh					0.8

Table 1: Similarity Matrix: Initial scoring

Table 1 shows that the initial scoring is more likely to match `bu` with `bush` instead of the correct matching with `georgebush`. This is because the labels in `bu` and `bush` are identical, which outscores the named entity type similarity in `bu` and `georgebush`. Similarly, the initial scoring shows an intial best match of `go` with `gore` instead of the correct match with `algore`.

Once the initial scoring is completed, the algorithm chooses an evaluation node from the update network (in this case `bu`) and attempts to improve the scores in its row. The `bu` node is fired in the update network, which sends activation to the `go` and `wh` nodes. For all nodes in the update network which received more than a minimum threshold of activation, their corresponding nodes are fired in the main network, with an initial activation level determined by their similarity score. For example, the amount of initial activation the `algore` node receives is based on the activation level of the `go` node and the similarity score between `algore` and `go`.

The `whitehouse` and `gore` nodes will fire in the main network, with the `gore` node receiving slightly more activation than the `algore` node, because of its higher similarity score. This firing pattern will cause activation to spread throughout the network; the `georgebush` node will receive some activation from the firing, while the `bush` node will not receive any.

Since the `georgebush` node received activation, its similarity score with the original evaluation node (`bu`) will be increased, while the similarity score between the `bush` and `bu` nodes will be decreased. Table 2 shows some typical scores resulting from this stage of the process.

The `go` node is then evaluated, with almost identical results, except that the `georgebush` node will fire with more activation than the `bush` node because of its improved score from the previous step, which results in the `algore` node receiving an even stronger score improvement. The com-

	georgebush	bush	algore	gore	whitehouse
bu	0.7	0.35			
go			0.5	0.7	
wh					0.8

Table 2: Similarity Matrix: After evaluating the bu node

bined improved results produce an even stronger effect when the wh node is evaluated. After one iteration, the similarity matrix is as given in Table 3.

	georgebush	bush	algore	gore	whitehouse
bu	0.7	0.35			
go			0.8	0.35	
wh					0.95

Table 3: Similarity Matrix: After one iteration

After several iterations, the similarity scores between bu and georgebush, go and algore, and wh and whitehouse will increase, and all other scores will drop to zero. Once a stopping criteria (number of iterations, or minimum change in scores) has been met, any node pairs with a similarity score above a pre-set threshold are assumed to denote the same real-world entity and are mapped together.

In this instance, lexical as well as semantic information was used to determine that the bu, go and wh nodes referred to George Bush, Al Gore and the United States' White House respectively. This was a simplified example, but the principle can be extended to deal with more complex networks.

Evaluation

By processing approximately 2 million sentences, we were able to build a network of over 1.5 million nodes and 3.5 million links in less than 3 days. This time also takes into account the parsing and semantic analysis (See Table 4). This is a vast improvement over manually created networks which take years or even decades to achieve networks of less than half this size (Matuszek *et al.* 2006).

Total Number of Nodes	1,500,413
Total Number of Edges	3,781,088
Time: Parsing	31hrs : 30 min
Time: Semantic Analysis	16 hrs: 54 min
Time: Building Network & Information Integration	22 hrs : 24 min
Time: Total	70 hrs : 48 min

Table 4: Statistics pertaining to the creation of a large scale semantic network

As the network grows, the time to perform the information integration step begins to climb exponentially. However, because the spreading activation algorithms are localised, once the network becomes so large that the activation does not spread to the majority of nodes, any increase in size ceases to have an effect on the algorithm. Therefore the average time to add a new node to the network is asymptotic and will eventually become constant regardless of network growth.

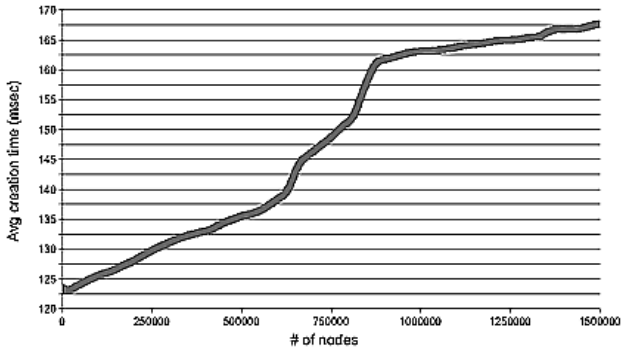

Figure 5: Average time to add a new node to the network vs. total number of nodes

The precision of large scale semantic networks is difficult to evaluate. Networks as diverse as those produced by ASKNet can represent the same information in several different ways, all of which could still be potentially semantically valid. Furthermore, the sheer size of the intended network raises many evaluation difficulties.

There is no "gold standard" against which to evaluate the generated networks. The manually created networks mentioned earlier cannot serve as a gold standard because they are incomplete, and thus it would be impossible to tell whether differences in the networks were due to an error in ASKNet, or information which is not represented in other networks. Even if the manually created networks had adequate coverage to serve as a gold standard within a particular domain, the structure of ASKNet networks are more complex than any manually created network currently available.

Ultimately, we believe that the best course of action is to use human evaluation, but once again the sheer size of the network and the diversity of its representations causes problems. It is not possibly for an evaluator to simply look at the network and determine if it is correct, nor is it practical to have an evaluator read all of the input given to the system in order to determine if information is missing.

We plan to solve these problems by evaluating the network indirectly; using the generated network as a world knowledge base for such tasks as multi document summarisation and question answering.

Future Work

There are several new features which we plan to implement such as: varying the strength of links based on the confidence of the parse or the source from which the information came; distinguishing between type and token instances of objects in the network; and implementing a more rigourous pruning algorithm to remove extraneous network fragments.

One potential application of ASKNet is the development of a multi-document summariser similar to those used in the

Document Understanding Conferences[1]. We can evaluate the network creation process indirectly by running ASKNet on the conference documents, and outputting small segments of the resulting network which a human evaluator can compare against the evaluation summaries. This task is particularly well suited as an evaluation for ASKNet as the documents processed all refer to the same topic and thus will have a high incidence of co-referent objects, which will provide more direct testing of the information integration properties of the system.

Conclusion

This paper has demonstrated that very large semantic networks can be created quickly using the output of a natural language parser. Parsing technology has reached a point where accurate, efficient, robust and linguistically-motivated parsers are now available. Sentences of natural language are a potential gold mine of semantic knowledge, and they are available in large quantities. The idea in this paper is to use a parser, together with spreading activation-based algorithms, to mine some of that knowledge, and then use spreading activation based algorithms to integrate the knowledge into a single cohesive resource.

We have chosen to represent the knowledge in the form of a semantic network, since this is a general semantic resource with potential uses in a number of AI applications. The enormous manual effort given to projects such as Cyc demonstrate the need for such a network. The innovation in this paper is that we have designed a method for efficiently generating large scale, expressive and well integrated semantic networks. We were able to create a network twice as large as Cyc in less than 3 days. Our approach has the potential to reduce the knowledge acquisition bottleneck, one of the key problems for AI.

References

Bos, J.; Clark, S.; Steedman, M.; Curran, J. R.; and Hockenmaier, J. 2004. Wide-coverage semantic representations from a CCG parser. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING-04)*, 1240–1246.

Clark, S., and Curran, J. R. 2004a. The importance of supertagging for wide-coverage CCG parsing. In *Proceedings of COLING-04*, 282–288.

Clark, S., and Curran, J. R. 2004b. Parsing the WSJ using CCG and log-linear models. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL '04)*, 104–111.

Collins, A. M., and Loftus, E. F. 1975. A spreading-activation theory of semantic processing. *Psychological Review* 82(6):407–428.

Curran, J. R., and Clark, S. 2003. Language independent NER using a maximum entropy tagger. In *Proceedings of the Seventh Conference on Natural Language Learning (CoNLL-03)*, 164–167.

Curtis, J.; Matthews, G.; and Baxter, D. 2005. On the effective use of Cyc in a question answering system. In *Papers from the IJCAI Workshop on Knowledge and Reasoning for Answering Questions*.

Dolan, W. B.; Vanderwende, L.; ; and Richardson, S. 1993. Automatically deriving structured knowledge base from on-line dictionaries. In *Proceedings of the Pacific Association for Computational Linguistics*.

Fellbaum, C., ed. 1998. *WordNet : An Electronic Lexical Database*. Cambridge, Mass, USA: MIT Press.

Hockenmaier, J. 2003. *Data and Models for Statistical Parsing with Combinatory Categorial Grammar*. Ph.D. Dissertation, University of Edinburgh.

Kamp, H., and Reyle, U. 1993. From discourse to logic : Introduction to modeltheoretic semantics of natural language. *Formal Logic and Discourse Representation Theory*.

Lenat, D. B. 1995. Cyc: A large-scale investment in knowledge infrastructure. *Communications of the ACM* 38(11):33 – 38.

Liu, H., and Singh, P. 2004. ConceptNet a practical commonsense reasoning tool-kit. *BT Technology Journal* 22:211 – 226.

Matuszek, C.; Cabral, J.; Witbrock, M.; and DeOliveira, J. 2006. An introduction to the syntax and content of Cyc. In *2006 AAAI Spring Symposium on Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*.

Meyer, D., and Schvaneveldt, R. 1971. Facilitation in recognizing pairs of words: Evidence of a dependence between retrieval operations. *Journal of Experimental Psychology* 90(2):227–234.

Pantel, P., and Pennacchiotti, M. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of Conference on Computational Linguistics / Association for Computational Linguistics (COLING/ACL-06)*, 113–120.

Richardson, S. D.; Dolan, W. B.; ; and Vanderwende, L. 1998. MindNet: Acquiring and structuring semantic information from text. In *Proceedings of COLING '98*.

Schubert, L., and Tong, M. 2003. Extracting and evaluating general world knowledge from the brown corpus. In *Proceedings of the HLT/NAACL 2003 Workshop on Text Meaning*.

Steedman, M. 2000. *The Syntactic Process*. Cambridge, MA.: The MIT Press.

Stocky, T.; Faaborg, A.; and Lieberman, H. 2004. A commonsense approach to predictive text entry. In *Proceedings of Conference on Human Factors in Computing Systems*.

[1] http://www-nlpir.nist.gov/projects/duc/index.html