Word similarity on the taxonomy of WordNet

Dongqiang Yang and David M. W. Powers

School of Informatics and Engineering
Flinders University of South Australia
PO Box 2100, Adelaide 5001, South Australia
Dongqiang.yang, David.Powers@flinders.edu.au

Abstract. In this paper, we introduced two kinds of word similarity tests to investigate the capability of WordNet in measuring word similarity . Both are tested on two noun and verb data sets.

Introduction

Methodology

Measuring word similarity, in terms of how to utilize knowledge base, can also be classified into knowledge-rich and knowledge-poor methods (Grefenstette 1993; Gasperin, Gamallo et al. 2001). Knowledge-rich method requires semantic networks or a semantically tagged corpus to define the concept of word in the relation with other concepts or in the surrounding context. Most methods that calculate semantic distance through WordNet (Miller 1995) or Roget's thesaurus fall into this category. Knowledge-poor method mainly depends on the information or probability theory to substitute for the knowledge base. It may be further categorized according to how co-occurrence frequency data is handled:

- Vector space

Suppose that semantically related words are more likely to co-occur in the corpus. A matrix is constructed in word-by-word or word-by-document order with a cell value such as term frequency (TF) or TF*IDF (inverse document frequency). Word similarity is established by comparing distance measures such as the cosine coefficient or Euclidean distance.

- Syntactic parsing

Suppose that the semantic relatedness of words leads to their use in similar grammatical structures in their contexts. Judging word similarity is achieved by tagging parts-of-speech in the corpus, shallow parsing of sentences, specifying the relationship between chunks and comparing the syntactic components along with their dependency relations (Grefenstette 1993).

Word similarity in the thesaurus.

The popular methodologies for measuring semantic relatedness with the help of a thesaurus can be classified into two categories: one uses the solely semantic links (i.e. edge-counting), the other combines corpus statistics with the taxonomic distance.

Edge-counting methods

The edge-counting or shortest path method derives from the geometric model in Cognitive Psychology, where the shorter distance entails the stronger association between stimuli and response. It can be traced back to Quillian's semantic memory model (Quillian 1967; Collins and Quillian 1969) where concept nodes are planted within the hierarchical network and the number of hops between the nodes specifies the similarity of the concepts. Generally the similarity of words in the thesaurus space can be described as,

$$Sim(i, j) = 2D - Dist(i, j) \qquad \textbf{(1)}$$

where D is a constant (e.g. the maximum depth in the taxonomy of WordNet, viz. 16 if we presume all the hierarchies have a common node), Dist(I,j) is links between two concept nodes I and j. In the edge-counting methods distance is typically assessed by counting the edges traversed from $c1$ to $c2$ via *ncn*, $Dist(c1, c2)$. In the following parts, we will introduce a few popular edge-counting models working in the semantic hierarchy (cf. (Pedersen, Banerjee et al. 2003)).

Wu and Palmer's model

Wu and Palmer (Rada, Mili et al. 1989; Wu and Palmer 1994) proposed to measure the verbal concept similarity in the projected domain hierarchy when translating from English verbs to Chinese. According to the claims of Wu and Palmer, the relatedness of two words is the weighted sum of all their senses comparison depicted in the following:

$$Sim(v_i, v_j) = \sum_k w_k * \frac{2 * dep(ncn(c_{i,k}, c_{j,k}))}{dep(c_{i,k}) + dep(c_{j,k})} \qquad \textbf{(2)}$$

where $ncn(c_{i,k},c_{j,k})$ is the nearest common node (*ncn*) for the conceptual nodes $c_{i,k}$, $c_{j,k}$ of verbs v_i and v_j, *dep* is the depth of node relative to the root, w_k is the weight of each pair of concepts in each domain. The sum of w_k is 1.

This model is appropriate for measuring both verbs and nouns in the "IS-A" hierarchical concept net.

Leacock and Chodorow's model

Leacock and Chodorow (Leacock and Chodorow 1998) adopted the concept of information content (Resnik 1995) in part to evaluate the relatedness of two words using the following model:

$$\begin{aligned} Sim(W_i, W_j) &= Max\left[-\log \frac{Dist(c_i, c_j)}{2 * D}\right] \qquad \textbf{(3)} \\ &= Max\left[\log 2D - \log Dist(c_i, c_j)\right] \end{aligned}$$

where $Dist(c_i,c_j)$ is the shortest distance between concepts c_i and c_j. In addition, they defined the similarity of two words as the maximized value of all the pairwise similarities.

Note that in Equation (3)

$$Dist(c_i,c_j)=dep(c_i)+ dep(c_j)-2*dep(ncn(c_i,c_j)) \qquad \textbf{(4)}$$

$$Sim(W_i, W_j) = Max\left[\log \frac{2D}{Dist(c_i, c_j)}\right] \qquad \textbf{(5)}$$

Hence, the concept model is similar to Wu and Palmer's apart from the *log* normalization.

Corpus based approaches.

Resnik's information content

Resnik (Resnik 1995) argues that the links in the hierarchy of WordNet representing a uniform distance in the edge-counting measurement can not account for the semantic variability of a single link. He defines information content of *ncn* to explain the similarity of two words through frequency statistics retrieved from a corpus, not through the distance of edge-counting. Here the frequency of *ncn* subsumes all the frequency data of subordinate concept nodes. The information content can be quantified as the negative of the log likelihood, $-logP(c)$.

However, Resnik still employs the structure of a conceptual net and one drawback is that the *ncn* for all concept pairs that have the same parent node is the same.

Jiang and Conrath's model

On the basis of Resnik's work, Jiang and Conrath (Jiang and Conrath 1997) further assumed that a combination of information content and edge-counting will improve the correlation co-efficient (compared with human judgment). They also considered the link type, depth, conceptual density, and information content of concepts. Their simplified formula can be expressed as follows:

$$Dist(c_i,c_j) = IC(c_i) + IC(c_j) – 2*IC(ncn(c_i, c_j)) \qquad \textbf{(6)}$$

$$Sim(c_i, c_j) = –Dist(c_i, c_j) \qquad \textbf{(7)}$$

Lin's model

Lin (Lin 1997) introduced another way of in computing the similarity to disambiguate word sense,

$$Sim(c_i, c_j) = \frac{2 * IC(ncn(c_i, c_j))}{IC(c_i) + IC(c_j)} \qquad \textbf{(8)}$$

which is essentially another normalized form of Jing and Conrad's model.

A new model

Generally speaking, similarity models in the taxonomy of WordNet, proposed by Wu and Palmer, Leacock and Chodorow, Jiang and Conrath, and Lin, can be abstracted into one of the following forms,

$$Sim(c1, c2) = 2\gamma \div (\alpha + \beta) \qquad \textbf{(9)}$$

$$Sim(c1, c2) = 2\gamma - (\alpha + \beta) \qquad \textbf{(10)}$$

where α, β, γ, respectively denote attributes of concepts $c1$, $c2$, and the *ncn* of $c1$,$c2$ in the "IS-A" hierarchy. The attribute can be viewed as the depth in the taxonomy or information content extracted from the outer corpus.

Yang and Powers (Yang and Powers 2005) propose a new model to measure semantic similarity in the taxonomy of WordNet, on the ground of edge-counting techniques. Different from the above methods they take into account the part-whole (hol/meronym) relationships in the WordNet while employing two kinds of searching algorithms, i.e. bidirectional depth-limit search (BDLS) and uni-directional breadth-first search (UBFS), and devise a distinctive metric to generate the concept similarity on the findings of the two searchings.

On the assumption that a single link in the taxonomy always stands for the same depth-independent distance and that the distance between two conceptual nodes is the least number of links from one node to another, they define the similarity of two concepts as,

$$Sim(c1, c2) = \alpha_t \beta^{\lambda} \qquad \textbf{(11)}$$

Partially inspired by Hirst and St. Onge (Hirst and St.Onge 1995) assigning respectively 3 different weights for identical words, synonyms or antonyms, and hyper/hyponym in the process of building lexical chains to solve the problem of the detection and correction of malapropisms, they deal with the identity case where $c1$ and $c2$ are identical as $\alpha_{id} = 1$, $\gamma = 0$, the syn/antonym as an intermediate weight, $\alpha_{sa} = 0.9$, $\gamma = 0$, lower weight (e.g. $\alpha = \alpha_{hh} = \alpha_{hm} = 0.85$, $\beta = \beta_{hh} = \beta_{hm} = 0.7$) for the hyper/hyponym, hol/meronym where searching depth γ is more than one.

They appraise the model against a benchmark set by human similarity judgment, and achieve a much improved result compared with other methods: the correlation with average human judgment on a standard 28 noun pair dataset (Resnik 1995) is 0.921, which is better than anything reported in the literature and also significantly better than average individual human judgments. As this set has been effectively used for algorithm selection and tuning, they also cross-validate an independent 37 noun pair test set (0.876) and present results for the full 65 noun pair (Rubenstein and Goodenough 1965) superset (0.897). Note that their best performance on these data sets is achieved on the maximum of the sense distances model, with respect to most words are polysemous.

Verb model

To investigate the correctness of the model on the judging word similarity we also apply it on verbs, because the verbs are another important part in the WordNet. Not like its counterpart noun taxonomy rich in the complexity and links, the verbs are orgernized into very shallow hierarchy according to their hyper/troponymy relations. The further distance to reach from most verbs is no more than 4 nodes, which make it difficult to find more relationships between verbs (Fellbaum 1998). As an improvement to yang and powers noun model, we design a new one to account for the similarity of verbs to attack the sparseness of verb hierarchy. As a supplement of the verb hierarchy, we also consider the derived noun hierarchy, definitions, and stemming effect. Generally we consider the following factors in constructing the model of verb similarity.

1. the similarity on the verb taxonomy is still similar to the noun hierarchy , viz. equation (11) and (12), except we exclude holo/meronym relationships. We set up thresholds for the syno/antonyms, hyper/troponyms, which are same as the noun model.
2. some verbs have the derived noun forms in its signature which are morphologically related. So we can introduce the noun hierarchy into the verbs which will flourish the relations among verbs. α_{der}
3. the definition of verb can give a hint to the relation with other verbs while there are no apparent linkages in the verb and noun hierarchies. Lesk (Lesk 1986) proposed to calculate the overlaps of target word and other words in the context in the definitions to select an appropriate sense. Pedersen et al. (Pedersen, Banerjee et al. 2003) treat the definitions in WordNet as a over one million word corpus, and build a co-occurrence matrix to specify how many times the two concepts turn up together in the gloss of WordNet. In this paper we assume verbs in the definition of WordNet, which are not in the frequent word list like “make”, “do”, etc., bring about a strong semantic relation with its target word. It is denoted as α_{gls} , gls(gloss).
4. the stemming effect, α_{stm} stm(stem).

Comprehensively considering these new factors, and other link type and depth factors, which we need to readjust in the taxonomy of WordNet, the new model is, (note that yang and powers have draw a optimum model in the noun hierarchy, which means we just simply inherited and make no adjustment)

$$Sim(c1,c2) = \alpha_{stm} * \alpha_t \prod_{i=1}^{dist(c1,c2)} \beta_{t_i}, \quad dist(c1, c2) < \gamma \qquad \textbf{(12)}$$

$$Sim(c1,c2) = 0, \quad dist(c1, c2) \geq \gamma \qquad \textbf{(13)}$$

$$Sim_{max}(v1,v2) = \underset{(i,j)}{Max}\left[Sim(c_{1,i}, c_{2,j})\right] \qquad \textbf{(14)}$$

where $0 \leq Sim(c1, c2) \leq 1$ and

- $t = ht$ (hyper/troponym), *sa* (syn/antonym), *der* (derived nouns), *gls* (definition).
- α_t: a link type factor applied to a sequence of links of type *t*. ($0 < \alpha_t \leq 1$).
- α_{stm}: the stemming factor, if c1 is linking c2 without stemming, $\alpha_{stm} = 1$
- β_t: the depth factor depending on the link type.
- γ: an arbitrary threshold on the distance, which will no more than five in the verb taxonomy.
- $dist(c1, c2)$: the distance (the shortest path) of $c1$ and $c2$.
- $c1$, $c2$: concept node 1 and concept node 2.

The most strongly related concepts are the identity case where *c*1 and *c*2 are identical, $\alpha_{id} = 1$, $Dist(c1, c2) = 0$. For the link type of syn/antonym, we assign an intermediate weight (e.g. $\alpha_{sa} = 0.9$, $Dist(c1, c2) = 0$), and we assign lowest weight (e.g. $\alpha_{ht} = 0.85$) for the hyper/troponymy. Recall that any syn/antonym and identity links constitute entire paths and cannot be part of a multilink path.

With the fact of most verbs being polysemous we assign the maximum value of the similarity among all the n_i senses $c_{i,j}$ of word v_i.

To demonstrate the finalized model of verb similarity in the WordNet we explain it in the following algorithm.

```
Algorithm: Input(verb1,verb2), Output(similarity score)
For each sense c1 and c2 in the verb1 and verb2
  if c1 and c2 are synonymous or antonymous semantically
    assign sim_sa(c1,c2)= •sa; Go next loop;
  elsif c1 can find c2 in its definition or vice verse
    sim_gls(c1,c2)= •gls ;
  else
    if both c1 and c2 have derived noun form
      go into noun taxonomy, BDLS search;
      sim_der(c1,c2)= •der*sim_noun(c1,c2).
    endif
    while the searching depth d is less than •
      if finding c1 and c2 are hyper/troponymy, or antonym on each
joint node
        sim_hta(c1,c2)= •ht* •ht^d .
      elsif finding c1 and c2 have same stemming word in the verb
taxonomy
        sim_stm(c1,c2)= •stm*•ht*•ht^d ;
      endif
    endwhile
  endif
  return the maximum value of all the similarity score, Simmax(v1,v2)
endfor.
```

Evaluation

Task

Unfortunately, there is no benchmark data set for verbs in the literature. We have to make our own data set to inspect the model. We select 20 verb synonym tests in 80 TOEFL(Test of English as a Foreign Language)[12] questions, which is first used by (Landauer and Dumais 1997)and 16 from 50 ESL(English as a

---

[1] Test of English as a Foreign Language (TOEFL), Educational Testing Service, Princeton, New Jersey, http://www.ets.org/.

second language) questions (Tatsuki 1998) , which are widespread taken as a test set in computational linguistics and regarded as the basic requirement for the entry of universities or working in the English countries. For these 36 questions, we take each target word and one of its four choices to construct one pair of verbs in the questionnaire, where there are 144 pairs verbs. We randomly arrange these word pairs and reverse the order of target verb and its choices. There are 6 of colleagues in our school (2 academic staffs and 4 postgraduates) voluntarily participating in this test. Four of them are native English speakers; the other two take English as a second language and a main communication tool in the academic and ordinary life. We ask them to carefully read the instructions and think about how likely these words are to occur in the same context (sentence/paragraph) before the questionnaire, and then indicate how strongly these words are related in meaning using integers from 0 to 4, which respectively means not at all related, vaguely, indirectly, strongly or inseparably related. If they think something falls in between two of these categories, they must push it up or down. We eliminate some of the less well-related ones, and select 130 words with average human scores as the last data set. These 130 words are sorted in descending order, and dived into 26 words in each category. We randomly pick up 13 words in each category and at last produce two 65 pairs data sets, i.e. data1 and data2. the average correlation among these six subjects is r = 0.866.

We optimize verb model in each data set through calculating the correlation with average human acores, and comprimise the factors of the models as the last values for the verb model. Here we just show how to regulate the verb model on the data1.

To distinguish the different effect of each factor we suppose, we assume the contribution of verb hierarchy, as well as derived noun hierarchy and the gloss are independent. Therefore we first seek for the optimal pattern in the verb hierarchy without any interfaction with other α_{der} and α_{gls}, then consider if the noun hierarchy is helpful to improve the correlation, and then we tune the definition factor.

Tuning

There are totally 6 factors we need to adjust, the path type factor α_{t}, the link type factor β, the depth factor γ, the stemming factor α_{stm} included in the verb taxonomy, the derive noun forms α_{der}, and the gloss factor α_{gls}.

Step1: the distance-limit (γ)

Once the values of α, α_{stm} and β had been assigned initially, i.e. respectively 0.85, 0.5 and 0.5, we varied the distance-limit γ (for the combined path length), enlarging the search distance of each node from 1 to 5 (essentially the maximum distance is no more than 5 in the WordNet), viz. the total distance of two node in the BDLS is from 2 to 10, to investigate if by expanding of the distance-limit, the model could produce a judgment that is more accurate. We can tell in the figure (1) that there is a drop in the correlation when we increase the searching scope from 1 level to 2 level, after that the curve approached level. Our purpose in the paper is to investigate the function of verb hierarchy, so we select γ = 6, denoted as a rich hierarchy explorationl (RHE). On the other hand we also keep γ = 2 as a reference point, along with similar fine-tuning process, and separately list the result at last which will be denoted shallow hierarchy exploration (SHE). In the following part we just illustrate how to calibrate RHE model.

Step 2: the link type factor (β)

We tested β over the range 0.3 to 0.7 tuning with increments of 0.1, to see if it affects the correlation with human judgment. Note that each link in the taxonomy is of uniform distance if we give β = 1. In fact, we find that the performance of the system begins to deteriorate as β becomes bigger than 0.6 with a max at 0.5.

Step 3: the path type factor (α)
We varied the value of α, by increments of 0.05 from 0.5 to 0.95. The optimal value for α is around 0.8 but there is very little sensitivity to its precise value.

Step4: the stemming factor (α_{stm})
After the optimal value, 0.4, the correlation began to drop down quickly.

Step5: the derived noun factor (α_{der})
There are no big differences when α_{der} increase from 0 to 0.5, after that the correlation deteriorated slowly. We set 0.4 as a compromising choice.

Step 6: the gloss factor (α_{gls})
There starts a jump at 0.4, ending until 0.9.

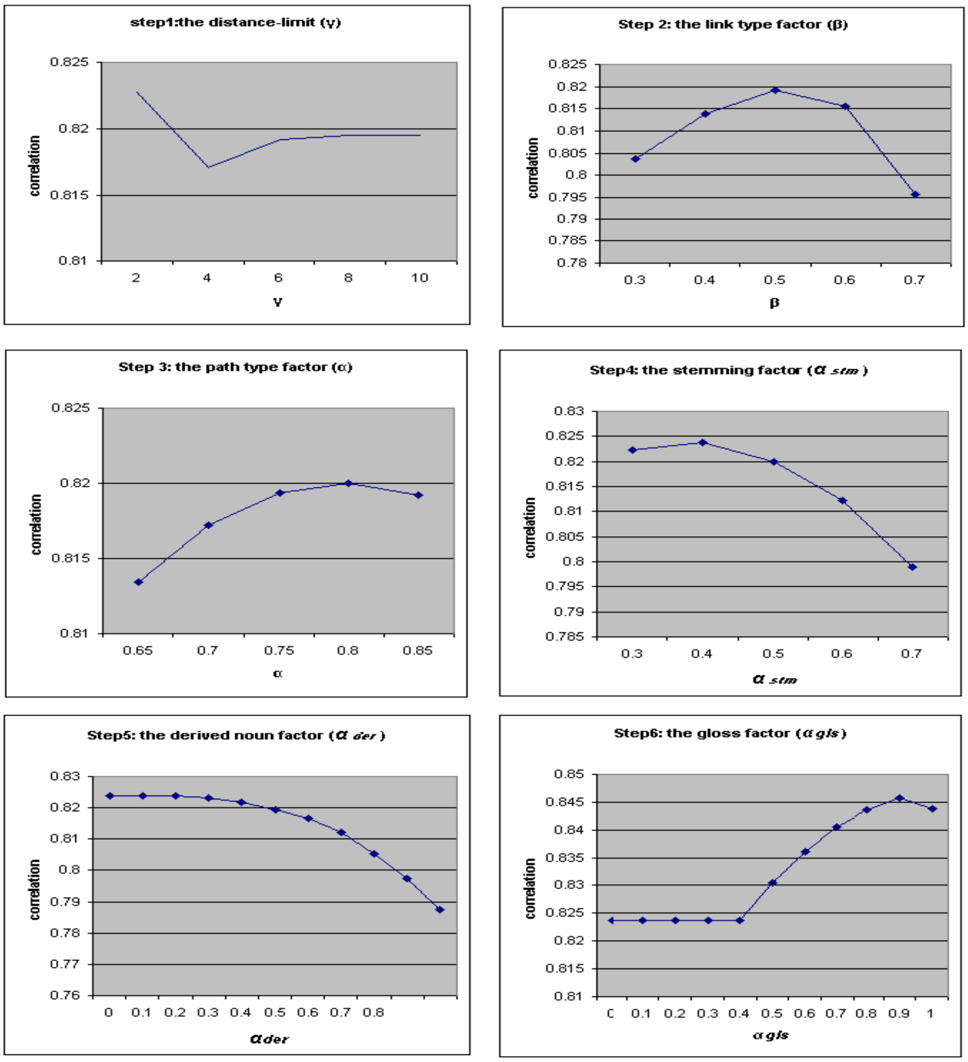

Fig. 1. the tuning process on the RHE

Result

After we optimized the verb model on each data set we found their premier values fit not very well with each other. Owing to the sensitivity of each data set on the correlation, say r, we make a compromise on these values. Table 1 shows the final parameters and correlations with the average human scores with RHE and SHE. There is no big difference on the final verb model through RHE and SHE with the exception of their depth limitation.

Table 1. the final result on the each 65 data sets and the total dataset. (r_t: the correlation on the tuning set, r_e: the correlation on the evaluation set, data1 is the evaluation set of data2, and vice verse.)

		γ	β	α	α_{stm}	α_{der}	α_{gls}	r_t	r_e
R	Data1(65)	2	0.5	0.8	0.4	0.1	0.9	0.846	0.775
H	Data2(65)	2	0.2	0.85	0.7	0.8	0.5	0.864	0.823
E	Total (130)	2	0.5	0.8	0.5	0.75	0.6	0.808	
S	Data1 (65)	0	0.6	0.75	0.4	0.7	0.9	0.838	0.824
H	Data2 (65)	0	0.4	0.8	0.6	0.7	0.5	0.846	0.835
E	Total (130)	0	0.5	0.8	0. 5	0.75	0.6	0.833	

Discussion

Table 2. significant test on both RHE and SHE, r_a: the correlation with average human, σ: standard deviation, μ: mean,

	r_a	σ/μ	RHE		SHE	
			z-score	Significance	z-score	Significance
Subject1	0.88	0.292	-3.25	0.001	-2.113	0.035
Subject2	0.733	0.45	0	1	-0.802	0.423
Subject3	0.878	0.488	-3.07	0.002	-3.421	<0.001
Subject4	0.926	0.485	-3.52	<0.001	-1.14	0.254
Subject5	0.913	0.397	-4.47	<0.001	-3.596	<0.001
Subject6	0.868	0.402	-1.89	0.059	-1.61	0.107
RHE	0.808	0.308	0	1	-1.484	0.138
SHE	0.833	0.561	-1.484	0.138	0	1

In yang and powers noun work they employed Wilcoxon Signed Rank Test to substitute the two-sample t test with respect to the scale discrepancy of human judgment. We also perform the rank significance test at 95% level, listed in the table 2. the verb model with RHE and SHE makes no difference in the ability of judging verb similarity, and is only significantly better than one subject. The distinctive part of RHE and SHE is that 3 subjects can not significantly do better than SHE, 1 for RHE, although their judgements keep very high correlation with average human.

the differences of yang and powers noun model from our verb model

1. the maximum links each node can reach in the verb model are much less than the γ in the noun model. Moreover the link type factor in the verb model also more quickly reduce the similarity of node in the next level with the target node. So do the path type factor. All of these facts partly tell us that verb hierarchy exists in a very shallow way in human, or the hierarchy does a limit help in assessing the similarity of verbs.

After taking into account the definition of verb

The definition of "*concoct#v#4*" is "*devise or invent*" in the WordNet. However there are no any other links like hyper/troponym, or syn/antonym relations we can exploit.

Conclusion

Reference

Collins, A. M. and M. R. Quillian (1969). "Retrieval time from semantic memory." Journal of Verbal Learning and Verbal Behavior 8: 240-47.
Fellbaum, C. (1998). An Electronic Lexical Database. London, England, The MIT Press.
Gasperin, C., P. Gamallo, et al. (2001). Using Syntactic Contexts for Measuring Word Similarity. Workshop on Semantic Knowledge Acquisition & Categorisation (ESSLLI 2001). Helsinki.
Grefenstette, G. (1993). Evaluation Techniques for Automatic Semantic Extraction: Comparing Syntactic and Window Based Approaches. Workshop on acquisition of lexical knowledge from text columbus.
Hirst, G. and D. St.Onge (1995). Lexical chains as representations of context for the detection and correction of malapropisms. WordNet. C. Fellbaum. Cambridge, MA, The Mit Press.
Jiang, J. and D. Conrath (1997). Semantic similarity based on corpus statistics and lexical taxonomy. proceedings on international conference on research in computational linguistics, Taiwan.
Landauer, T. K. and S. T. Dumais (1997). " A solution to Plato’s problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge." Psychological Review 104: 211-240.
Leacock, C. and M. Chodorow (1998). Combining local context and WordNet similarity for word sense identification. WordNet: An electronic lexical database. C. Fellbaum, MIT Press: 265-283.
Lesk, M. (1986). Automatic sense diaambiguation using machine readable dictionaries: how to tell a pine code from an ice cream cone. the 5th annual international conference on systems documentation, ACM Press.
Lin, D. (1997). Using syntactic dependency as a local context to resolve word sense ambiguity. Proceedings of the 35th annual meeting of the association for computational linguistics, Madrid.
Miller, G. (1995). "A lexical database for English." Communications of the ACM 38,11: 39-41.
Pedersen, T., S. Banerjee, et al. (2003). Maximizing Semantic Relatedness to Perform Word Sense Disambiguation.
Quillian, M. R. (1967). "Word concepts: A theory and simulation of some basic semantic capabilities." Behavioral Science 12: 410-30.
Rada, R., H. Mili, et al. (1989). "Development and application of a metric on semantic nets." IEEE Transactions on Systems, Man and Cybernetics 19(1): 17-30.
Resnik, P. (1995). Using information content to evaluate semantic similarity in a taxonomy. Proceedings of IJCAI-95.
Rubenstein, H. and J. B. Goodenough (1965). "Contextual correlates of synonymy." communications of the ACM 8(10): 627-633.
Tatsuki, D. (1998). Basic 2000 Words - Synonym Match 1. In: Interactive JavaScript Quizzes for ESL Students. http://www.aitech.ac.jp/~iteslj/quizzes/js/dt/mc-2000-01syn.html.
Wu, Z. and M. Palmer (1994). Verb semantics and lexical selection. 32nd. Annual Meeting of the Association for Computational Linguistics.
Yang, D. and D. M. W. Powers (2005). Measuring Semantic Similarity in the Taxonomy of WordNet. Twenty-Eighth Australasian Computer Science Conference (ACSC2005), Newcastle, Australia, ACS.

Appendix:

the total 130 pairs of verbs.

brag	boast	hail	acclaim	refer	explain	request	levy	anger	approve
concoct	devise	dissipate	disperse	finance	build	arrange	study	approve	boast
divide	split	approve	support	expect	deserve	relieve	hinder	research	distribute
build	construct	impose	levy	terminate	postpone	move	swell	request	concoct
end	terminate	hasten	accelerate	yell	boast	weave	print	boast	yield
accentuate	highlight	rap	tap	swell	curl	swear	think	furnish	impress
demonstrate	show	lean	rest	rotate	situate	forget	resolve	refine	sustain
solve	figure out	make	earn	seize	request	supervise	concoct	acknowledge	distribute
consume	eat	show	publish	approve	scorn	situate	isolate	clean	concoct
position	situate	sell	market	supply	consume	explain	boast	lean	grate
swear	vow	weave	intertwine	clip	twist	ache	spin	postpone	show
furnish	supply	refer	direct	divide	figure out	evaluate	terminate	hail	judge
merit	deserve	distribute	commercialize	advise	furnish	recognize	succeed	remember	hail
submit	yield	twist	intertwine	complain	boast	dilute	market	scrape	lean
seize	take	drain	tap	want	deserve	hasten	permit	sweat	spin
spin	twirl	depict	recognize	twist	fasten	scorn	yield	highlight	restore
enlarge	swell	build	organize	swing	crash	swear	describe	seize	refer
swing	sway	hail	address	make	trade	arrange	explain	levy	believe
circulate	distribute	call	refer	hinder	yield	discard	arrange	alter	highlight
recognize	acknowledge	swing	bounce	build	propose	list	figure out	refer	carry
resolve	settle	yield	seize	express	figure out	stamp	weave	empty	situate
prolong	sustain	split	crush	resolve	examine	market	sweeten	flush	spin
tap	knock	challenge	yield	bruise	split	boil	tap	shake	swell
block	hinder	hinder	assist	swing	break	sustain	lower	imitate	highlight
arrange	plan	welcome	recognize	catch	consume	resolve	publicize	correlate	levy
twist	curl	need	deserve	swear	explain	dissipate	isolate	refer	lean