STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LII**, Number 1, 2007

SEMANTIC SIMILARITY KNOWLEDGE AND ITS APPLICATIONS

DIANA INKPEN

Abstract. Semantic relatedness refers to the degree to which two concepts or words are related. Humans are able to easily judge if a pair of words are related in some way. For example, most people would agree that *apple* and *orange* are more related than are *apple* and *toothbrush*. Semantic similarity is a subset of semantic relatedness. In this article we describe several methods for computing the similarity of two words, following two directions: dictionary-based methods that use WordNet, Roget's thesaurus, or other resources; and corpus-based methods that use frequencies of co-occurrence in corpora (cosine method, latent semantic indexing, mutual information, etc). Then, we present results for several applications of word similarity knowledge: solving TOEFL-style synonym questions, detecting words that do not fit into their context in order to detect speech recognition errors, and synonym choice in context, for writing aid tools. We also present a method for computing the similarity of two short texts, based on the similarities of their words. Applications of text similarity knowledge include: designing exercises for second language-learning, acquisition of domain-specific corpora, information retrieval, and text categorization. Before concluding, we briefly describe cross-language extensions of the methods for similarity of words and texts.

1. Methods for Word Similarity

Semantic relatedness refers to the degree to which two concepts or words are related (or not) whereas semantic similarity is a special case or a subset of semantic relatedness. Humans are able to easily judge if a pair of words are related in some way. For example, most would agree that *apple* and *orange* are more related than are *apple* and *toothbrush*. Budanitsky and Hirst [4] point out that semantic similarity is used when similar entities such as *apple* and *orange* or *table* and *furniture* are compared. These entities are close to each other in an *is-a* hierarchy. For example, *apple* and *orange* are hyponyms of *fruit* and *table* is a hyponym of *furniture*. However, even dissimilar entities may be semantically related, for example, *glass* and *water*, *tree* and *shade*, or *gym* and *weights*. In this

Received by the editors: June 19, 2007.

2000 *Mathematics Subject Classification.* 91F20, 68T35.

1998 *CR Categories and Descriptors.* [I.2.7]: Natural Language Processing; [I.2.4]: Knowledge Representation Formalisms and Methods.

case the two entities are intrinsically not similar, but are related by some relationship. Sometimes this relationship may be one of the classical relationships such as meronymy (*is part of*) as in *computer – keyboard* or a non-classical one as in *glass – water*, *tree – shade* and *gym – weights*. Thus two entities are semantically related if they are semantically similar (close together in the *is-a* hierarchy) or share any other classical or non-classical relationships. Measures of the semantic similarity of words have been used for a long time in applications in natural language processing and related areas, such as the automatic creation of thesauri [6], [18], [17], automatic indexing, text annotation and summarization [20], text classification, word sense disambiguation [15], [17], information extraction and retrieval [3], [30], lexical selection, automatic correction of word errors in text [4], and discovering word senses directly from text [23]. A word similarity measure was also used for language modeling by grouping similar words into classes [1].

There are two types of methods for computing the similarity of two words: dictionary-based methods (using WordNet, Roget's thesaurus, or other resources) and corpus-based methods (using statistics). There are also a few hybrid methods that combine the two types.

Most of the dictionary-based methods compute path length in WordNet, in various ways. A short path means a high similarity. For example, using the WordNet entries for the words *apple* and *orange* the path length is 3:

```
apple  (sense 1)
   => edible fruit
       => produce, green goods, green groceries, garden truck
           => food
               => solid
                   => substance, matter
                       => object, physical object
                           => entity
orange (sense 1)
   => citrus, citrus fruit
       => edible fruit
           => produce, green goods, green groceries, garden truck
              => food
                => solid
                   => substance, matter
                       => object, physical object
                           => entity
```

The WordNet::Similarity Software Package[1] implements several WordNet-based similarity measures: Leacock & Chodorow (1998) [14], Jiang & Conrath (1997) [12], Resnik (1995) [25], Lin (1998) [18], Hirst & St-Onge (1998) [7], Wu & Palmer

[1] http://www.d.umn.edu/~tpederse/similarity.html

(1994) [28], extended gloss overlap, Banerjee & Pedersen (2003) [2], and context vectors, Patwardhan (2003) [24].

If the two words have multiple senses, the similarity between them, out of context, is the maximum similarity between any of the senses of the two words. Three of the above methods are hybrid (Jiang & Conrath (1997) [12], Resnik (1995) [25], Lin (1998) [18]), they use frequency counts for word senses from Semcor, which is a small corpus, annotated with WordNet senses.

Other resources that can be used are thesauri, such as Roget's Thesaurus. For example, the words *apple* and *orange* are in the same paragraph in Roget, but not in the same semicolon group:

```
301 FOOD
n.
fruit, soft fruit, berry, gooseberry, strawberry, raspberry,
loganberry, blackberry, tayberry, bilberry, mulberry;
currant, redcurrant, blackcurrant, whitecurrant;
stone fruit, apricot, peach, nectarine, plum, greengage, damson, cherry;
apple, crab apple, pippin, russet, pear;
citrus fruit, orange, grapefruit, pomelo, lemon, lime, tangerine,
clementine, mandarin;
banana, pineapple, grape;
rhubarb;
date, fig;
```

A similarity measure using Roget's thesaurus [11] computes the distance between the words by exploiting the structure of the thesaurus (path length):

- Length 0: same semicolon group. Example: *journey's end – terminus*
- Length 2: same paragraph. *devotion – abnormal affection*
- Length 4: same part of speech. *popular misconception – glaring error*
- Length 6: same head. *individual – lonely*
- Length 8: same head group. *finance – apply for a loan*
- Length 10: same sub-section. *life expectancy – herbalize*
- Length 12: same section. *Creirwy (love) – inspired*
- Length 14: same class. *translucid – blind eye*
- Length 16: in the thesaurus. *nag – like greased lightning*

Corpus-based methods use frequencies of co-occurrence in corpora. They range from the classic vector-space model (cosine, overlap coefficient, etc.) and latent semantic analysis, to probabilistic methods such as information radius and mutual information.

Examples of large corpora are the British National Corpus (BNC) (100 million words), the TREC data mainly newspaper text, the Waterloo Multitext corpus of webpages (one terabyte), the LDC English Gigabyte corpus, and the Web itself.

Examples of corpus-based measures are[2]: Cosine, Jaccard coefficient, Dice coefficient, Overlap coefficient, L1 distance (city block distance), Euclidean distance (L2 distance), Information Radius (Jensen-Shannon divergence), Skew divergence, and Lin's Dependency-based Similarity Measure[3].

The classic vector space model represents all the words as vectors in an high-dimensional space where the dimensions are the documents (we build a matrix of words by documents). The cosine between two vectors gives the similarity of two terms.

Latent Semantic Analysis (LSA) [4] [13] produces a reduced words by documents matrix, which has fewer dimensions corresponding to the *latent* topics of the documents.

Pointwise Mutual Information (PMI) is very simple distributional measure that works well only in very large corpora. The similarity between two words w_1 and w_2 is given by the probability of seeing the two words together in a corpus divided by the probability of seeing them separately. This compensates for the chance of random co-occurrence when the words are frequent.

$$PMI(w_1, w_2) = log\frac{P(w_1, w_2)}{P(w_1)\ P(w_2)}$$

$$PMI(w_1, w_2) = log\frac{C(w1, w2)\ N}{C(w_1)\ C(w_2)}$$

The probabilities are simply the observed frequencies divided by N, the number of words in the corpus. We used the Web as a corpus, therefore we used the number of retrieved documents (hits returned by a search engine) to approximate the word co-occurrence counts, ignoring the fact that a word can be repeated in a document. Our experiments showed that using document counts instead of word counts leads to similar results.

A similarity measure that uses second-order co-occurrences (SOC-PMI) [10] works well even on a smaller corpus (BNC) because it looks at the words that co-occur with the two words. The method sort lists of important neighbor words of the two target words, using PMI, then it takes the shared neighbors and adds their PMI values, from the opposite list (normalizing by the number of neighbors).

2. Evaluation of Word Similarity Measures

Miller and Charles [22] asked several humans to judge the similarity of 30 noun pairs, a subset of the 65 noun pairs judged in a similar way by Rubenstein and Goodenough [26]. Here are some examples of pairs and similarity values, on a scale of 0 to 4 (averaged over the human judges):

[2] http://clg.wlv.ac.uk/demos/similarity/

[3] http://www.cs.ualberta.ca/~lindek/demos.htm

[4] http://lsa.colorado.edu/

Method Name	Miller and Charles 30 Noun Pairs	Rubenstein and Goodenough 65 Noun Pairs
Cosine (BNC)	0.406	0.472
SOC-PMI (BNC)	0.764	0.729
PMI (Web)	0.759	0.746
Leacock & Chodorow (WN)	0.821	0.852
Roget	0.878	0.818

Table 1. Correlations of similarity measures with human judges.

```
gem, jewel, 3.84
coast, shore, 3.70
asylum, madhouse, 3.61
magician, wizard, 3.50
shore,woodland,0.63
glass,magician,0.11
```

An automatic similarity method is considered good if it produces values that correlate well with the human values (correlation close to 1). Correlations for several measures are presented in table 1. Corpus-based values tend to have lower correlations than WordNet-based measures, because WordNet has a well-developed noun hierarchy. Among the WordNet-based measures we listed only the one with the highest correlation, the Leacock & Chodorow measure [11]. The Roget measure also has a very good correlation. Among the corpus-based measures, SOC-PMI and PMI are good.

The correlation with the human judges is a recommended evaluation step, but not sufficient because it can be done only on a small set of noun pairs. It can be used to filter out measures that are not promising.

The task-based evaluation section is the most indicative. The similarity measures can be evaluated in one or more tasks. The best measure is the one that achieves the highest performance in the evaluation measure appropriated for the task. It could be the case that different measures perform best for different tasks. Three tasks are presented in section 3.

A third type of evaluation measure consists in building an automatic thesaurus, by selecting a small number of close semantic neighbors for each word. Retrieval of semantic neighbors can be evaluated as in information retrieval systems [27]. The expected solution is an existing manually-built resource. A problem with this method is that resources tend to have different coverage.

3. Applications

3.1. **Solving TOEFL-style Synonym Questions.** A task commonly used in the evaluation of similarity measure is solving TOEFL-style questions. Two datasets

Method Name	Number of Correct Test Answers	Question/Answer Words Not Found	Percentage of Correct Answers
Roget	63	26	78.75%
SOC-PMI	61	4	76.25%
PMI-IR	59	0	73.75%
LSA	51.5	0	64.37%
Lin	32	42	40.00%

Table 2. Results on the 80 TOEFL Questions.

Method Name	Number of Correct Test Answers	Question/Answer Words Not Found	Percentage of Correct Answers
Roget	41	2	82%
SOC-PMI	34	0	68%
PMI-IR	33	0	66%
Lin	32	8	64%

Table 3. Results on the 50 ESL Questions.

are available: 80 synonym test questions from the Test of English as a Foreign Language (TOEFL) and 50 synonym test questions from a collection of English as a Second Language (ESL). An example of TOEFL question is:

```
The Smiths decided to go to Scotland for a short .......... They have already
booked return bus tickets.
      (a) travel
      (b) trip
      (c) voyage
      (d) move
```

The solution is one of the four choices that fits best into the context of the two sentences. The similarities between a choice word and each of the content words[5] in the sentences are added up, and the choice with the highest values is considered the solution. The results for the TOEFL questions results are presented in table 2 [10]. The results for the ESL questions are presented in table 3. The similarity measures from the tables are: Roget similarity [11], PMI-IR [29], SOC-PMI [10], LSA [13], and Lin [19]. The last one performs worse because many words were not available in the resource (a database of dependency relations). The best performance is achieved by the Roget measure.

3.2. **Detecting Speech Recognition Errors.** Another tasks is the detection of the words that do not fit into their context. For example, a spell-checker will not signal out words that are valid words but not the intended words. For example a

[5] We ignore function words such as prepositions, conjunctions, etc.

user could types *raw and column* when it was meant *row and column.* The task of real-word error correction [4] would detect that *raw* is a mistake, and suggest that *row* has higher similarity with the other words in the text than *raw.*

We applied this idea to the task of detecting speech recognition errors [9], as words that have low semantic similarity with their context. The data we used is 100 stories from the TDT corpus, which had manual transcripts. The automatic speech transcripts were produced with the BBN speech recognizer and had a word error rate of about 25%. Here is an example of automatic transcript and the corresponding manual transcript:

```
BBN transcript: time now for a geography was they were traveling down river
to a city that like many russian cities has had several names but this one
stanza is the scene of ethnic and national and world war two in which the
nazis were nine elated

Manual transcript: Time now for our geography quiz today. We’re traveling
down the Volga river to a city that, like many Russian cities, has had
several names. But this one stands out as the scene of an epic battle in
world war two in which the Nazis were annihilated.

Detected outliers:  stanza, elated
```

Our algorithm detected two words as potential errors (semantic outliers). For each word w in the automatic transcript, the algorithm executed the following steps:

(1) Compute the neighborhood $N(w)$, i.e. the set of content words that occur close to w in the transcript (include w).
(2) Compute pair-wise semantic similarity scores $S(w_i, w_j)$ between all pairs of words $w_i \neq w_j$ in $N(w)$, using a semantic similarity measure.
(3) Compute the semantic coherence $SC(w_i)$ by adding the pair-wise semantic similarities $S(w_i, w_j)$ of w_i with all its neighbors $w_j \neq w_i$ in $N(w)$.
(4) Let SC_{avg} be the average of $SC(w_i)$ over all w_i in the neighborhood $N(w)$.
(5) Label w as a recognition errors if $SC(w) \leq K\ SC_{avg}$.

The neighborhood of a word could be the whole speech segment or part of it (a context window). The average coherence of the segment times a parameter K is used for comparison, as a threshold for signaling semantic outliers.

We varied the parameter K in order to detect more or fewer semantic outliers as potential speech recognition errors. Detecting too many brings the risk of signaling words that are not really speech recognition errors. We evaluated the performance in terms of the precision of the detected outliers and of their recall. The results in figure 1 show that using the PMI similarity measure (computed in the Waterloo Multitext corpus of Web data) leads to better results than using the

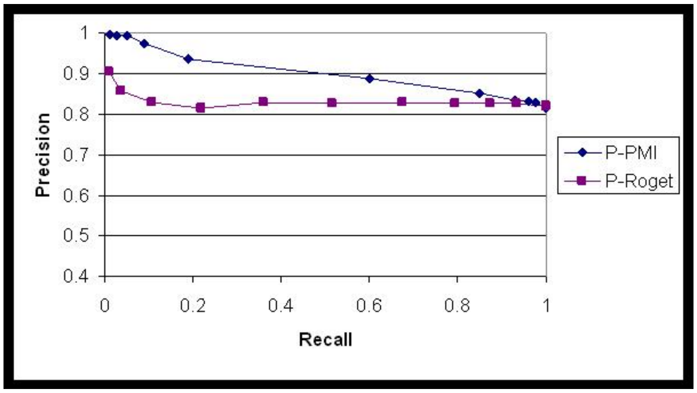

Figure 1. Results for detecting speech recognition errors.

Roget similarity measure. The Roget measure performed worse because some of the words were not found in the thesaurus.

3.3. **Synonym Choice in an Intelligent Thesaurus.** A third task that we describe concerns synonym choice in context, for writing aid tools. We developed an intelligent thesaurus [8], that allows a writer to select a word and to ask for synonym that would be alternative choices. There is a thesaurus is Microsoft Word that allows the writer to do this, but it does not order the choices by their suitability. Our thesaurus computes for each choice its similarity to the context, and orders the choices by these values. This helps the user to select the best choice.

In order to evaluate the method, we selected sentences and took out a word, creating a gap. Then we found synonyms for that word, and computed their similarity to the context. If the highest ranked synonym is exactly the word that we took out (the word that was in the original sentence), we consider that the recommendation of the intelligent thesaurus was correct.

Here are two examples of sentences and synonym sets. For the first one the original word was *error*, for the second one it was *job*.

```
Sentence: This could be improved by more detailed consideration of the
processes of ......... propagation inherent in digitizing procedures.
Solution set: mistake, blooper, blunder, boner, contretemps, error, faux pas,
goof, slip, solecism

Sentence:    The effort required has had an unhappy effect upon his prose,
on his ability to make the discriminations the complex ......... demands.
Solution set: job, task, chore
```

Test set	Baseline Most Freq. Syn.	Edmonds, 1997	Accuracy First Choice	Accuracy First Two Choices
Data set 1 Syns: WordNet (7 groups) Sentences: WSJ	44.8%	55%	66.0%	88.5%
Data set 2 Syns: CTRW (11 groups) Sentences: BNC	57.0%	–	76.5%	87.5%

Table 4. Results for the intelligent thesaurus.

We used the PMI measure with the Waterloo Multitext corpus and a context window of k content words before the gap and k words after the gap (k=2 was the best value, determined experimentally).

The results are presented in table 4. The first dataset used newspaper sentences (WSJ) and synonyms form WordNet. Our results were much better than a baseline of always choosing the most frequent synonym, and than a pervious method of Edmonds [5] that uses a lexical co-occurrence network. We improve over the baseline also on a second dataset, with sentences from the BNC and synonyms from a special dictionary of synonyms named *Choose the Right Word* (CTRW).

4. Text Similarity

The similarity of two texts can be computed in several ways, including the classic vector space model. Applications of text similarity knowledge include designing exercises for second language-learning, acquisition of domain-specific corpora, information retrieval, and text categorization.

Here we present a method for computing the similarity of two short texts, based on the similarities of their words. We used the SOC-PMI corpus-based similarity for two words. In addition, we used string similarity (longest common subsequence). The method selects a word from the first text and a word from the second text, which have the highest similarity. The similarity value is stored, and the two words are taken out. The method continues until there are no more words. At the end, the similarity scores are added and normalized.

For evaluation we used a data set of 30 sentence pairs for which similarity values computed by human judges were available [16]. In Figure 2 we present the correlation between the scores produced by our method and the average of the scores given by the human judges. Our results are better than the results of the method of Li et al. [16], based on a lexical co-occurrence network. The last two bars in the figure show how much the human judges varied from their mean.

The second dataset that we used for evaluation was the Microsoft Paraphrases corpus. It contains pairs of sentences that are marked as being paraphrases or not.

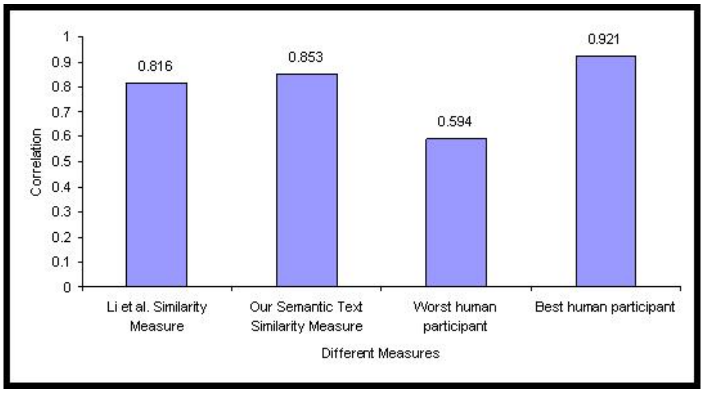

Figure 2. Correlation with human judges on the 30 sentence pairs.

Metric	Accuracy	Precision	Recall	F-measure
Random Baseline	51.3	68.3	50.0	57.8
Vector-based	65.4	71.6	79.5	75.3
Jiang & Conrath	69.3	72.2	87.1	79.0
Leacock & Chodorow	69.5	72.4	87.0	79.0
Lesk	69.3	72.4	86.6	78.9
Lin	69.3	71.6	88.7	79.2
Wu & Palmer	69.0	70.2	92.1	80.0
Resnik	69.0	69.0	96.4	80.4
Combined (Supervised)	71.5	72.3	92.5	81.2
Combined (Unsupervised)	70.3	69.6	97.7	81.3
PMI-IR	69.9	70.2	95.2	81.0
LSA	68.4	69.7	95.2	80.5
STS	**72.6**	**74.7**	**89.1**	**81.3**

Table 5. Results on the MicroSoft Paraphrases corpus.

In this case we can evaluate if our method considers the two sentences as similar or not, we cannot evaluate the scores themselves.

Table 5 compares our results (the last line – Semantic Text similarity – STS) with the results obtained by Mihalcea et al. [21] on the same dataset. They used several WordNet-based measures, and combinations of these measures. We also compare to the PMI-IR and LSA corpus-based similarity measures. Our results are similar or slightly better than those of other methods.

5. Conclusion and Future Work

We presented an overview of the methods for computing word similarity. We discussed several ways to evaluate them. The main one is to evaluate them by how well they perform when solving specific tasks. We looked at three particular applications. We also discussed methods of computing the similarity of two short texts based on the similarity of their words.

There are several directions for future work. We plan to extend our second-order co-occurrences similarity measure to use a Web corpus, specifically the Google 5-gram corpus. This measure is promising because it worked well on the BNC

More investigation is needed in combining word similarity methods, in order to produce hybrid methods that use very large corpora. Such corpora are not annotated with WordNet senses. Automatic words sense disambiguation methods, though not powerful enough in general, could be sufficient for gathering statistics on word sense distribution in very large corpora.

We plan to develop cross-language similarity methods, for two words in different languages. If the two words are translations of each other, their similarity is maximal. If they are not translation the similarity could vary between zero and a value close to 1. For example, the similarity between the French word *pomme* and the English word *orange* can be computed by simply translating the French word into English (let's say the translations are *apple*, *potato*, and *head*), and take the maximum similarity between the translations and the second word. In this case the cross-language similarity is reduced to the similarity between the English words *apple* and *orange*. All is needed is a bilingual dictionary with a good coverage.

The cross-language similarity of two texts can be computed in the same way as the similarity of two texts in the same language, by using the similarity between the words (the cross-language word similarity measure that we sketched above). The cross-language similarity of two texts could be used in second language teaching to select similar texts, or in cross-language information retrieval.

References

[1] P.F. Brown, P.V. DeSouza, R.L. Mercer, T.J. Watson, V.J. Della Pietra, and J.C. Lai. Class-based n-gram models of natural language. Computational Linguistics, 18:467-479, 1992.

[2] S. Banerjee and T. Pedersen. Extended gloss overlaps as a measure of semantic relatedness. In Proceedings of IJCAI 2003.

[3] C. Buckley, J.A. Salton and A. Singhal. Automatic query expansion using Smart: TREC 3. In The third Text Retrieval Conference, Gaithersburg, MD, 1995.

[4] A. Budanitsky and G. Hirst. Evaluating WordNet-based measures of semantic distance. Computational Linguistics, 32(1), 2006.

[5] P. Edmonds. Choosing the word most typical in context using a lexical co-occurrence network. In Proceedings of ACL 1997.

[6] G. Grefenstette. Automatic thesaurus generation from raw text using knowledge-poor techniques. In Making Sense of Words, 9th Annual Conference of the UW Centre for the New OED and Text Research, 1993.

[7] G. Hirst and D. St-Onge. Lexical Chains as representations of context for the detection and correction of malapropisms. In WordNet An Electronic Database, 1998.

[8] D. Inkpen. Near-synonym choice in an Intelligent Thesaurus, HLT-NAACL 2007.

[9] D. Inkpen and A. Desilets. Semantic similarity for detecting recognition errors in automatic speech transcripts. In Proceedings of EMNLP 2005.

[10] A. Islam and D. Inkpen. Second order co-occurrence PMI for determining the semantic similarity of words. In Proceedings of LREC 2006.

[11] M. Jarmasz and S. Szpakowicz. Roget's thesaurus and semantic similarity. In Proceedings of RANLP 2003.

[12] J. Jiang and D. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In Proceedings of COLING 1997.

[13] T.K. Landauer and S.T. Dumais. A Solution to Plato's problem: The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge. Psychological Review, 104(2), 1997.

[14] C. Leacock and M. Chodorow. Combining local context and WordNet sense similarity for word sense identification. In WordNet, An Electronic Lexical Database, 1998.

[15] Lesk, M.E. Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In Proceedings of the SIGDOC Conference , Toronto, 1986.

[16] Y. Li, D. McLean, Z. Bandar, J. O'Shea, K. and Crockett. Sentence similarity based on semantic nets and corpus statistics. IEEE Trans. Knowledge and Data Eng. 18:8, 2006.

[17] H. Li and N. Abe. Word clustering and disambiguation based on co-occurrence data. In Proceedings of COLING-ACL, 1998, pp. 749-755.

[18] D. Lin. An information-theoretic definition of similarity. In Proceedings of ICML 1998.

[19] D. Lin. Automatic retrieval and clustering of similar words. In Proceedings of COLING-ACL, 1998, pp. 768-774.

[20] C.Y. Lin and E.H. Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In Proceedings of HLT-NAACL, 2003.

[21] R. Mihalcea, C. Corley, C. Strapparava. Corpus-based and knowledge-based measures of text semantic similarity. In Proceedings of AAAI 2006.

[22] G.A. Miller and W.G. Charles. Contextual correlates of semantic similarity. Language and Cognitive Processes, 6(1): 1-28, 1991.

[23] P. Pantel and D. Lin. Discovering word senses from text. In Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2002, pp. 613-619.

[24] S. Patwardhan. Incorporating dictionary and corpus information into a vector measure of semantic relatedness. MSc Thesis, University of Minnesota, 2003.

[25] P. Resnik. Semantic similarity in a taxonomy: An information-based measure and its applications to problems of ambiguity in natural language. JAIR 11, 1999.

[26] H. Rubenstein and J.B. Goodenough. Contextual correlates of synonymy. Communications of the ACM, 8(10): 627-633, 1995.

[27] J. Weeds, D. Weir and D. McCarthy. Characterising measures of lexical distributional similarity. In Proceedings of COLING 2004.

[28] Z. Wu and M. Palmer. Verb semantics and lexical selection. In Proceedings of ACL 1994.

[29] P.D. Turney. Mining the Web for synonyms: PMI-IR versus LSA on TOEFL. In Proceedings of ECML 2001.

[30] J. Xu and B. Croft. Improving the effectiveness of information retrieval. ACM Transactions on Information Systems, 18(1):79-112, 2000.

University of Ottawa, School of Information Technology and Engineering
E-mail address: diana@site.uottawa.ca