Unsupervised Semantic Similarity Computation using Web Search Engines

Elias Iosif and Alexandros Potamianos
Department of Electronics and Computer Engineering
Technical University of Crete
Chania 73100, Greece
{iosife, potam}@telecom.tuc.gr

Abstract

In this paper, we propose two novel web-based metrics for semantic similarity computation between words. Both metrics use a web search engine in order to exploit the retrieved information for the words of interest. The first metric considers only the page counts returned by a search engine, based on the work of [1]. The second downloads a number of the top ranked documents and applies "wide-context" and "narrow-context" metrics. The proposed metrics work automatically, without consulting any external knowledge resource. The metrics are compared with WordNet-based methods. The metrics' performance is evaluated in terms of correlation with respect to the pairs of the commonly used Charles - Miller dataset. The proposed "wide-context" metric achieves 71% correlation, which is the highest score achieved among the fully unsupervised metrics in the literature up to date.

1 Introduction

Many applications dealing with information retrieval and natural language processing require knowledge of semantic similarity between words. For example, in query expansion the addition of semantically related words to the query it is likely to increase the relevant retrieved documents, which in other case may be missed [9]. This raises the idea of semantic representation and retrieval rather than lexical string matching [5]. In [24, 14, 7], the queries are expanded using related words, acquired from WordNet. In general, query expansion is shown to increase the recall of the retrieved documents. Gaugh et al. [9] added semantic similar words to queries according to a similarity measure. The similarity between query and candidate additional words was calculated using a corpus-based cosine similarity measure.

Semantic similarity metrics are also used in Semantic Web applications, like automatic annotation of web pages [3] and social networks construction [15, 17]. Mori et al. [17], applied a similarity metric to calculate the context similarity between entity pairs in order to cluster them according to their similarity. After the creation of clusters, a number of terms were extracted from each cluster and used as labels in order to describe the relations among the entity pairs.

Moreover, semantic similarity measures are important for many natural language processing (NLP) tasks, such as language modeling [8], word sense disambiguation, speech understanding and spoken dialogue systems. In [19, 11], several unsupervised, statistical metrics are discussed for automatic induction of semantic classes, applied on homogeneous and heterogeneous corpora.

The majority of the semantic similarity metrics employed today use hand-crafted language resources, e.g., ontologies and most commonly WordNet. The use and maintenance of resources, such as thesauri or ontologies, is a costly task. Also language resources are not ubiquitous, are unavailable for most language and provide no information for words or concepts not included in the repository. Continuously updating these resources is a time consuming and tedious task, demanding human labor and often expert knowledge.

Recently there has been much research interest in developing text-based approaches for estimating semantic similarity for unseen words; most of these methods use the web and search engines for text corpus mining. The web has a multilingual character, in which new words, neologisms and hapax legomena, are added frequently and efficiently. Thus it is the obvious place for mining semantic relationships for unseen words. Most of the text-based approaches to semantic similarity employ hand-crafted filtering rules and language resources to obtain and process the text corpus. As a result these methods are not of much use for applications where human and language resources are sparse.

In this paper, we focus on the problem of *fully unsupervised semantic similarity computation*, no hand-crafted rules or resources are employed. Web search engines are used for text corpus mining and context-based similarity

distances are automatically computed on this corpus. Unsupervised semantic similarity estimation algorithms are important because they require no expert knowledge and no language resources; for many languages and applications this is the only realistic choice. In addition to their practical interest, automatically acquiring semantic similarity from text can also help us better understand the human language acquisition process, which is also (at the semantic level) mostly unsupervised.

The remainder of the paper is organized as follows. In Section 2 we discuss the related work dealing with semantic similarity computation between words. The proposed metrics are defined in Section 3. The conducted experiments and the benchmark dataset are described in Section 4. In Section 5 we present the evaluation results for the proposed metrics and, also, we make a brief comparative discussion between these metrics and metrics that consult external knowledge resources, like WordNet. Section 6 concludes the paper and gives interesting directions for further research.

2 Related work

The metrics that measure semantic similarity between words or terms can be divided into three main categories regarding the use of knowledge resource or not: (a) resource-based metrics, consulting only human-built knowledge-bases, such as WordNet, (b) metrics that perform text mining, relying on knowledge resources, and (c) unsupervised metrics that are fully text-based, exploring only the raw textual information. The later category of metrics is fully automatic and does not use any knowledge resource.

Several methods of the first category have been proposed in the literature regarding the use of WordNet for semantic similarity computation. Edge counting methods consider the length of the paths that links the words, as well as words' positions in the taxonomic structure [13]. Information content methods find the difference of the contextual information between words as a function of their occurrence probability with respect to a corpus [12]. Hybrid methods combine synsets with word neighborhoods and other features [20].

In the work of Bollegala et al. [1], a hybrid method, among others, is defined that combines page counts, returned by a search engine, and lexico-syntactic patterns, extracted from the returned snippets using a number of synonymous nouns acquired from WordNet.

More recently, unsupervised, web-based similarity metrics have been proposed, that collect information by querying web search engines. Sahami et al. [21], measure the similarity between short text snippets by using web search engine results to get greater context for the examined snippets. Also, in [1], some co-occurence metrics are proposed that use only the returned by a search engine page counts as similarity features between words. Moreover, unsupervised, fully text-based similarity statistical metrics are defined in [19, 11], applied in various corpora.

3 Unsupervised web-based similarity metrics

In order to calculate the semantic similarity between words we present two types of unsupervised, web-based similarity metrics. The first type considers only the page counts returned by a web search engine, as in [1]. The second is fully text-based, exploring the contexts of downloaded documents that include the words of interest.

3.1 Page-count-based similarity metrics

The basic idea under this approach is that the word co-occurrence is likely to indicate some kind of semantic relationship between words. A quick approximation of word co-occurrence can be estimated exploring the web. However, the number of documents in which a certain word pair co-occurs, does not express a direct semantic similarity. In addition, it is reasonable to, also, take into account the number of documents that include the each pair component individually for normalization purposes. [1] In other words, for a word pair, we need to know the information that the two words share, normalized by the degree of their independence. We define the following [6]:
$\{D\}$: a set containing the whole document collection that are indexed and accessible by a web search engine
$|D|$: the number of documents in collection $\{D\}$
w_i: a word or term
$\{D|w_i\}$: a subset of $\{D\}$, documents indexed by w_i
$\{D|w_i, w_j\}$: a subset of $\{D\}$, documents indexed by w_i and w_j
$f(D|w_i)$: the fraction of documents in $\{D\}$ indexed with w_i
$f(D|w_i, w_j)$: the fraction of documents in $\{D\}$ indexed with w_i and w_j

We use three co-occurrence measures in this paper, Jaccard coefficient, Dice coefficient and Mutual Information, to compute semantic similarity between word pairs as in [1]. The Jaccard coefficient is a measurement calculating the similarity (or diversity) between sets. We use a variation of the Jaccard coefficient in this paper defined as:

$$Jaccard(w_i, w_j) = \frac{f(D|w_i, w_j)}{f(D|w_i) + f(D|w_j) - f(D|w_i, w_j)} \quad (1)$$

[1] It is interesting to note that web-based co-occurence metrics often outperform more elaborate corpus-based metrics. This shows that overcoming the data sparseness problem is sometimes more important than building an accurate estimator. For example an improved n-gram language probability estimation using web n-gram occurence can be found in the literature [26].

In probabilistic terms, Equation 1 finds the maximum likelihood estimate of the ratio of the probability of finding a document where words w_i and w_j co-occur over the probability of finding a document where either w_i or w_j occurs[2]. If w_i and w_j are the same word then the Jaccard coefficient is equal to 1 (absolute semantic similarity). If two words never co-occur in a document then the Jaccard coefficient is 0.

The Dice coefficient is related to the Jaccard coefficient and is computed as:

$$Dice(w_i, w_j) = \frac{2f(D\,|\,w_i, w_j)}{f(D\,|\,w_i) + f(D\,|\,w_j)} \quad (2)$$

Again, the Dice coefficient equals to 1 if w_i and w_j are identical, and 0 if two words never co-occur.

If we consider the occurrence of words w_i and w_j as random variables X and Y, respectively, then the pointwise mutual information (MI) among X and Y measures the mutual dependence between the appearance of words w_i and w_j [2]. The maximum likelihood estimate of MI is

$$MI(X,Y) = \log \frac{\frac{f(D|w_i,w_j)}{|D|}}{\frac{f(D|w_i)}{|D|}\frac{f(D|w_j)}{|D|}} \quad (3)$$

Mutual information measures the information that variables X and Y share. It quantifies how the knowledge of one variable reduces the uncertainty about the other. For instance, if X and Y are independent, then knowing X does not give any information about Y and the mutual information is 0. For $X = Y$, the knowledge of X gives the value of Y without uncertainty and the mutual information is 1. Note that the fractions of documents are normalized by the number of documents indexed by the search engine, $|D|$, giving a maximum likelihood estimate of the probability of finding a document in the web that contains this word.

3.2 Fully text-based similarity metrics

Two semantic similarity metrics that are variations of the cosine similarity metric are used in order to measure the semantic distance between words and to automatically generate semantic classes. The first metric, CS^W_{WS}, computes "wide-context" similarity between words using a "bag-of-words" model, while the second metric, CS^N, computes "narrow-context" similarity using a bigram language model.

These metrics rely on the idea that *similarity of context implies similarity of meaning*. We assume that words, which appear in similar lexical environment (left and right contexts), have a close semantic relation [10, 23, 19].

In "bag-of-words" [22, 11] models, for each word w in the vocabulary a context window size WS is selected. The right and left contexts of length WS in the corpus are considered for word w, e.g., $[v_{WS,L}$ ... $v_{2,L}$ $v_{1,L}]$ w $[v_{1,R}$ $v_{2,R}$ ... $v_{WS,R}]$, where $v_{i,L}$ and $v_{i,R}$ represent the i^{th} word to the left and to the right of w respectively. The feature vector for every word w is defined as $T_{w,WS} = (t_{w,1}, t_{w,2}, ..., t_{w,N})$ where $t_{w,i}$ is a non-negative integer and WS is the context window size. Note that the feature vector size is equal to the vocabulary size N, i.e., we have a feature for each word in the vocabulary V. The i^{th} feature value $t_{w,i}$ reflects the occurrences of vocabulary word v_i within the left or right context window WS. This feature value is set according to a Binary (Bin.) or a Term Frequency (Freq.) Scheme. The binary Scheme assigns 1 if the word v_i appears within the left and right window context of size WS for the word w, while the term frequency scheme assigns the number of occurrences of v_i in left and right WS. Both schemes assign a 0 value if v_i does not exist within WS. The "bag-of-words" metric, CS^W_{WS}, using binary or term frequency scheme, measures the similarity of two words, w_1 and w_2, as the cosine distance of their corresponding feature vectors, $T_{w_1,WS}$ and $T_{w_2,WS}$ [18, 11]:

$$CS^W_{WS}(w_1, w_2) = \frac{\sum_{i=1}^N t_{w_1,i} t_{w_2,i}}{\sqrt{\sum_{i=1}^N (t_{w_1,i})^2}\sqrt{\sum_{i=1}^N (t_{w_2,i})^2}} \quad (4)$$

given a context window of length WS.

In an n-gram language model a word w is considered with its neighboring words $v_{1,L}$ and $v_{1,R}$ in the left and right contexts within a sequence. In order to calculate the similarity of two words, w_1 and w_2, we compute the cosine similarity between two feature vectors; each feature vector of a word w measures the conditional probability of all possible contexts v_i given that word $p(v_i|w)$, i.e., each vector contains bigram language model probabilities for (context, word) pairs. Semantic similarity is defined as

$$CS^N(w_1, w_2) = CS^N_L(w_1, w_2) + CS^N_R(w_1, w_2) \quad (5)$$

where the two terms of Eq. (5) are [19, 11]:

$$CS^N_L(w_1, w_2) = \frac{\sum_{i=1}^N p(v_{i,L}|w_1) p(v_{i,L}|w_2)}{\sqrt{\sum_{i=1}^N p(v_{i,L}|w_1)^2}\sqrt{\sum_{i=1}^N p(v_{i,L}|w_2)^2}} \quad (6)$$

$$CS^N_R(w_1, w_2) = \frac{\sum_{i=1}^N p(v_{i,R}|w_1) p(v_{i,R}|w_2)}{\sqrt{\sum_{i=1}^N p(v_{i,R}|w_1)^2}\sqrt{\sum_{i=1}^N p(v_{i,R}|w_2)^2}} \quad (7)$$

where $V = (v_1, v_2, ..., v_N)$ is the vocabulary set, and $p(v_i|w)$ is the conditional probability of word v_i preceding w in the corpus given word w, i.e., the (v_i, w) bigram

[2]Normalization by the total number of documents $|D|$ is the same for the nominator and denominator, and can be ignored.

model probability. The bigram language model was built using the CMU Statistical Language Modeling Toolkit [4]. The "wide-context" and "narrow-context" metrics assign 0 similarity score to completely dissimilar words, and 1 in the case of identical words.

4 Experiments

We experimented with (i) page-count-based, and (ii) contextual similarity metrics, described in Section 3.1 and Section 3.2, respectively.

As a benchmark we used the commonly used Miller-Charles dataset [16]. This dataset consists of noun pairs that were rated according to their semantic similarity by 38 human subjects. The assigned similarity scores range from 0 (not similar) to 4 (perfect synonymy). The selection of this dataset was motivated by its wide use. This fact enabled us to compare our work with other approaches of different nature that were, also, evaluated on this dataset.

For the contextual similarity metrics, for each pair, "w_1 w_2", of Miller-Charles dataset, we downloaded the 100 top ranked documents for the following query types:

Type 1	"w_1 AND w_2" (e.g., "boy AND lad")
Type 2	"w_1", "w_2" (e.g., "boy", "lad")

Table 1. Query types

The *URLs* for the top ranked documents were retrieved using the Yahoo search engine via the Yahoo Search API which is freely available [25]. In Table 1 the first query type is a single query, which retrieves documents containing both words. The second query type consists of two distinct queries; the first one requires documents that contain, at least, w_1, while the second query is satisfied by documents in which w_2, at least, appears. For the first query type up to 100 documents were downloaded for each word pair (100 documents per word were downloaded for the second query type).

The motivation behind using the two different query types is that the type of documents retrieved by each type of query is semantically different. The Type 1 "and" query is expected to retrieve documents that are semantically homogeneous, while the Type 2 query will produce documents that are semantically more diverse. As a result using corpora generated from "and" type queries, i.e., for a particular word pair, the corresponding feature vectors are built on the basis of a relatively coherent lexical environment, since the component words co-occur in each retrieved document. In contrast, the use of Type 2 query type results to more diverse feature vectors. In this case the feature vector for each pair word is constructed using different documents of, probably, different expressive style and semantic content. In previous work [11], we have seen better results for semantic similarity computation in semantically homogeneous corpora. Also the optimum context length (window size) varies significantly depending on the semantic homogeneity of the corpus (smaller for semantically homogeneous corpora).

5. Evaluation

In this section, we present a comparative evaluation of the referred similarity metrics, in terms of correlation, with respect to the human rating of Miller-Charles pairs. First, the page-count-based similarity metrics defined in Section 3.1 are evaluated. Next, we evaluate the proposed fully text-based similarity metrics defined in Section 3.2. The proposed metrics are also compared with metrics that use WordNet as a knowledge source.

5.1 Evaluation of page-count-based metrics

The correlation scores between the page-count-based semantic similarity metrics of Eqs. (1) - (3) and human ratings of Miller-Charles pairs are presented in Table 2. The similarity metrics based on the Jaccard and Dice coefficients achieve similar correlation. This is reasonable given the similarity of the two metrics. The Mutual Information metric achieves significantly better performance. This is due to the use of the logarithm non-linearity in the score computation and the different normalization used in this metric. Overall, all the page-count-based metrics obtain poor correlation results. This is expected considering that no textual information is included in this metrics, only the page counts information.

Metric	Jaccard	Dice	Mutual Information
Correlation	0.32	0.33	0.43

Table 2. Correlation of page-count metrics.

5.2 Evaluation of fully text-based metrics

Next we evaluate the "wide-context" metric, CS^W_{WS}, which computes similarity between words using a "bag-of-words" model and the "narrow-context" metric, CS^N, which is based on a bigram language model. For the "wide-context" metric we studied the impact of the feature vector weighting scheme (binary or term frequency), as well as how the window size (WS) affects the semantic similarity computation.

Figure 1 illustrates the correlation between the "wide-context" similarity metric and the human ratings of 28 Miller-Charles pairs. The correlation is shown for both binary and term frequency weighting schemes and various

window sizes WS. The similarity metric was computed on the top 100 documents retrieved by the search engine for web queries of Type 1 ("and") for each of the 28 word pairs. The correlation performance of the "narrow-context" metric is also shown for the same query type as a dotted line.

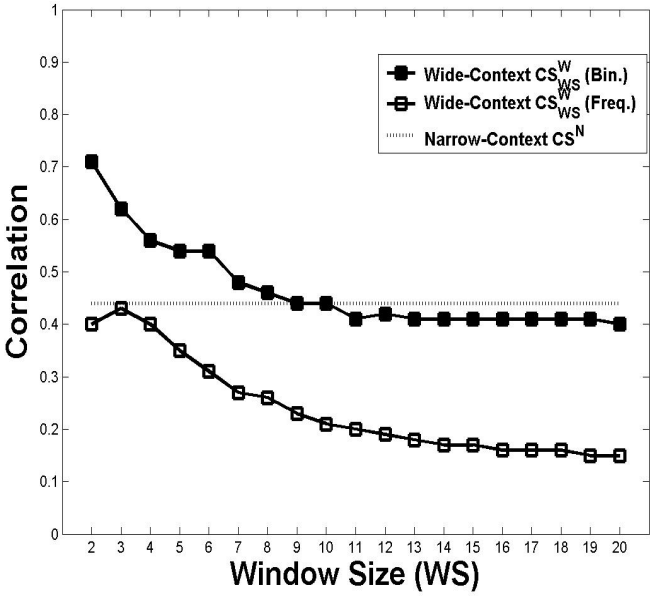

Figure 1. Correlation of fully text-based metrics for queries of Type 1

We observe that the correlation decreases as the value of window size, WS, increases; best results are obtained for window size two or three. This suggests that the immediate context (preceding and following two words) is sufficient to compute semantic similarity using "and" query type 1. Furthermore, the "wide-context" metric using the binary scheme outperforms the term frequency weighting scheme. This is probably due to the fact that the term frequency scheme gives more weight to the commonly occurring context words that are often "non-content" words (aka stopwords). These non-content words dominate the similarity calculation between the two term frequency vectors, while the binary weighting scheme is more robust to the presence of such words. The bigram "narrow-context" metric achieves almost equal performance with the "wide-context" metric of the term frequency scheme and $WS = 2$, since both metrics use contextual frequency weighting schemes. Overall, the highest correlation is obtained by the "wide-context" metric using the binary scheme and $WS = 2$, and it is equal to 0.71.

In Table 3 the correlation for the "wide-context" metric using the binary scheme and $WS = 2$ is shown as a function of the number of Type 1 retrieved documents. As expected performance degrades as the number of downloaded documents per word pair decreases. Note that good correlation (0.69) is achieved even when the top 50 downloaded documents are used in the similarity computation.

No. of Docs	5	10	25	50	75	100
Correlation	0.41	0.50	0.66	0.69	0.69	0.71

Table 3. Correlation vs. number of docs.

The use of queries of Type 2 resulted to unexpectedly poor correlation scores. This is probably due to the very different types of document retrieved with Type 2 queries, i.e., lack of semantic homogeneity in the downloaded corpus.

5.3 Unsupervised vs supervised metrics

In this section we compare the performance of the proposed unsupervised metrics with other supervised and unsupervised metrics. All of them were evaluated with respect to the 28 pairs of Charles - Miller dataset in terms of correlation. A metric is considered to be unsupervised if does not consult any knowledge resource. The main characteristics of each metric are summarized in Table 4. The detailed semantic similarity scores and the overall correlation with human scores are presented in Table 5 for each metric.

The Li [13], Jiang [12], and X-Similarity [20] metrics exploit the semantic hierarchical structure of WordNet, to compute semantic similarity as described in Section 2. All three metrics achieve higher correlation, but at the cost of using additional information that was not derived from text in an automatic unsupervised manner.

The performance of the web-based metrics is summarized as follows. The resource-based SemSim metric, proposed in [1], achieves a correlation score that is similar to the ontology-based methods above. The fully unsupervised Sahami [21] metric is shown to have a moderate correlation (results are reproduced from the implementation and evaluation in [1]). The lowest correlation scores are achieved by the metrics that consider only the returned page counts for a query: Jaccard, Dice and mutual information (MI) metrics. The proposed unsupervised, "wide-context" metric $CS^W_{WS=2}$ with the binary weighting scheme achieves the highest correlation (0.71) among the unsupervised metrics.

6 Conclusions and future work

We presented two types of unsupervised, web-based metrics for semantic similarity computation between words. Both types use a web search engine in order to exploit the retrieved information for the words of interest. The first type considers only the page counts returned by the search

Metric	Use of ($\surd$: yes, X: no)						Need of external knowledge	Correlation
	WWW Search engine	Page counts	Snippets	Lexico-Syntactic patterns	WordNet	Download documents		
Jaccard	$\surd$	$\surd$	X	X	X	X	X	0.32
Dice	$\surd$	$\surd$	X	X	X	X	X	0.33
MI	$\surd$	$\surd$	X	X	X	X	X	0.43
Sahami	$\surd$	X	$\surd$	X	X	X	X	0.58
SemSim	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$	X	$\surd$	0.83
Li	X	X	X	X	$\surd$	X	$\surd$	0.82
Jiang	X	X	X	X	$\surd$	X	$\surd$	0.83
X-Similarity	X	X	X	X	$\surd$	X	$\surd$	0.75
Proposed $CS^{W}_{WS=2}$ (Binary)	$\surd$	X	X	X	X	$\surd$	X	0.71

Table 4. Characteristics of several similarity metrics

engine. The second type is fully text-based and needs a number of the top ranked documents to be downloaded. We applied two "wide-context" metrics and a "narrow-context" metric to the downloaded documents. The proposed metrics do not consult any external knowledge resource. The metric performance was evaluated on the commonly used Charles - Miller word pair dataset.

The page-count-based metrics produced low to mid correlation with human scores. Good correlation scores were obtained with the fully text-based metric using a binary weighting scheme, especially for small context windows. The semantic distance was computed for each word pair on a corpus of 50-100 retrieved documents where the words co-occured. The best performance achieved for this metric was 0.71, which is the highest correlation score among the fully unsupervised metrics in the literature. Furthermore, we compared the proposed metrics with various state-of-art resource-based metrics that use ontologies (e.g., WordNet) to compute semantic similarity. Overall, the performance of the proposed method is satisfactory given that the method is language-independence, fully automatic, requires little computation-power and small amounts of web text.

We are currently investigating a variety of criteria for improving the semantic similarity method including better document selection (as opposed to web search engine ranking) and better context word feature extraction (including unsupervised algorithms for stemming and part of speech tagging). Further research is needed to better understand the limited performance for Type 2 queries and for the frequency weighting scheme.

Acknowledgments This work was partially supported by the EU-IST-FP6 MUSCLE network of excellence. The authors wish to thank Kelly Zervanou for many useful discussions.

References

[1] Bollegala, D., Matsuo, Y., Ishizuka, M. Measuring Semantic Similarity between Words using Web Search Engines. In: Proc. Int. WWW2007 Conf., 2007.

[2] Church, K., Hanks, H. Word association norms, mutual information, and Lexicography. In: Proc. 27th. Annual Meeting of the Association for Computational Linguistics, 1989.

[3] Cimano, P., Handschuh, S., Staab, S. Towards the Self-annotating Web. In: Proc. Int. WWW2004 Conf., 2004.

[4] Clarkson, P.R., Rosenfeld, R. Statistical Language Modeling Using the CMU-Cambridge Toolkit. In: Proc. EUROSPEECH, 1997.

[5] Drymonas, E., Zervanou, K., Petrakis, G.M.E. . Exploiting MultiWord Similarity for Information Retrieval: the TSRM Approach. ACM 16th Conference on Information and Knowledge Management (to appear), 2007.

[6] Feldman, R., Dagan, I. Mining Text using Keywords Distributions. Journal of Intelligent Information Systems, 1998.

[7] Flank, S. A Layered Approach to NLP-based Information Retrieval. In: Proc. 17th Int. Conf. on Computational linguistics, 1998.

[8] Fosler-Lussier, E., Kuo, H.-K. J. Using Semantic Class Information for Rapid Development of Language Models Within ASR Dialogue Systems. In: Proc. ICASSP, 2001.

[9] Gauch, S., Wang, J. A Corpus Analysis Approach for Automatic Query Expansion. In: Proc. 6th Int. Conf. on Information and Knowledge Management, 1997.

[10] Herbert R., Goodenough, B.J. Contextual Correlates of Synonymy. Communications of the ACM, vol. 8, 1965.

[11] Iosif, E., Tegos, A., Pangos, A., Fosler-Lussier, E., Potamianos, A. Unsupervised Combination of Metrics for Semantic Class Induction. In: Proc. IEEE/ACL Spoken Language Technology Workshop, 2006.

[12] Jiang, J.J., Conrath, D.W. Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In: Proc. Int. Conf. in Computational Linguistics, 1998.

[13] Li, Y., Bandar, Z.A., McLean, D. An Approach for Measuring Semantic Similarity between Words using Multiple

Word Pair	Miller-Charles	Jaccard	Dice	MI	Sahami	SemSim	Li	Jiang	X-Similarity	Binary $CS^{W}_{WS=2}$
chord-smile	0.13	0.14	0.14	0.78	0.09	0	0.13	0.35	0.2	0.4
rooster-voyage	0.08	0.03	0.03	0.76	0.2	0.02	0	0.08	0	0
noon-string	0.08	0.2	0.21	0.79	0.08	0.02	0	0.18	0	0.16
glass-magician	0.11	0	0	0.82	0.14	0.18	0.14	0.68	0.14	0.18
monk-slave	0.55	0.23	0.24	0	0.1	0.38	0.45	0.39	0.32	0.19
coast-forest	0.42	0.61	0.63	0.81	0.25	0.41	0.25	0.29	0.18	0.76
monk-oracle	1.1	0.07	0.07	0.77	0.05	0.33	0.25	0.34	0.28	0.47
lad-wizard	0.42	0.09	0.1	0.82	0.15	0.22	0.45	0.32	0.36	0.37
forest-graveyard	0.84	0.13	0.13	0.85	0	0.55	0.09	0.19	0.07	0.11
food-rooster	0.89	0.03	0.03	0.76	0.8	0.6	0.04	0.4	0.05	0.35
coast-hill	0.87	0.99	0.99	0.82	0.29	0.87	0.44	0.71	0.34	0.18
car-journey	1.16	0.33	0.35	0.75	0.19	0.29	0	0.33	0.03	0.52
crane-implement	1.68	0.08	0.09	0.78	0.15	0.13	0.44	0.59	0.29	0.1
brother-lad	1.66	0.22	0.23	0.88	0.24	0.34	0.45	0.28	0.45	0.58
bird-crane	2.97	0.28	0.29	0.86	0.22	0.88	0.55	0.73	0.37	0.59
bird-cock	3.05	0.16	0.17	0.79	0.06	0.6	0.82	0.73	0.45	0.44
food-fruit	3.08	0.85	0.86	0.83	0.18	0.99	0.13	0.63	0.09	0.79
brother-monk	2.82	0.33	0.35	0.88	0.27	0.38	0.82	0.91	0.44	0.63
asylum-madhouse	3.61	0.07	0.08	0.95	0.21	0.77	0.82	0.97	0.44	0.51
furnace-stove	3.11	0.67	0.68	1	0.31	0.89	0.23	0.39	0.47	1
magician-wizard	3.5	0	0	0.92	0.23	1	1	1	0.1	0.59
journey-voyage	3.84	0.37	0.39	0.84	0.52	0.99	0.82	0.88	0.52	0.75
coast-shore	3.7	0.8	0.82	0.87	0.38	0.95	0.81	0.99	0.66	0.5
implement-tool	2.95	1	1	0.87	0.42	0.68	0.81	0.97	0.48	0.8
boy-lad	3.76	0.22	0.23	0.87	0.47	0.97	0.82	0.88	0.44	0.67
automobile-car	3.92	0.61	0.63	0.82	1	0.98	1	1	1	0.76
midday-noon	3.42	0.14	0.15	0.88	0.29	0.82	1	1	1	0.74
gem-jewel	3.84	0.44	0.45	0.91	0.21	0.69	1	1	1	0.53
Correlation	1	0.32	0.33	0.43	0.58	0.83	0.82	0.83	0.75	0.71

Table 5. Correlation for several types of similarity metrics

Information Sources. IEEE Trans. on Knowledge and Data Engineering, 2003.

[14] Mihalcea, R., Dan Moldovan, D. Semantic Indexing using WordNet Senses. In: Proc. ACL-2000 Workshop on Recent Advances in Natural Language Processing and Information Retrieval, 2000.

[15] Mika, P.,. Ontologies are Us: A Unified Model of Social Networks and Semantics. In: Proc. ISWC2005, 2005.

[16] Miller, G., Charles, W. Contextual Correlates of Semantic Similarity. Language and Cognitive Processes, 1998.

[17] Mori, J., Tsujishita, T., Matsuo, Y., Ishizuka, M. Extracting Relations in Social Networks from the Web Using Similarity Between Collective Contexts. In: Proc. 20th IJCAI, 2006.

[18] Pangos, A., Iosif, E., Potamianos, A., Fosler-Lussier, E. Combining Statistical Similarity Measures for Automatic Induction of Semantic Classes. In: Proc. IEEE Automatic Speech Recognition and Understanding Workshop, 2005.

[19] Pargellis, A., Fosler-Lussier, E., Lee, C., Potamianos, A., Tsai, A. Auto-Induced Semantic Classes. Speech Communication. 43, 183-203., 2004.

[20] Petrakis, G.M.E., Varelas, G., Hliaoutakis, A., Raftopoulou, P. X-Similarity: Computing Semantic Similarity between Concepts from Different Ontologies. Journal of Digital Information Management, 2006.

[21] Sahami, M., Heilman, T. A Web-based Kernel Function for Measuring the Similarity of Short Text Snippets. In: Proc. Int. WWW2006 Conf., 2006.

[22] Sebastiani, F. Machine learning in automated text categorization. ACM Computing Surveys, 34(1):1 47, 2002.

[23] Siu, K.-C., Meng, H.M. Semi-Automatic Acquisition of Domain-Specific Semantic Structures. In: Proc. EUROSPEECH, 1999.

[24] Voorhees, E. Query Expansion using Lexical-Semantic Relations. In: Proc. 17th Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, 1994.

[25] YahooSearchAPI. http://developer.yahoo.com/search/.

[26] Zhu, X., Rosenfeld, R. Improving Trigram Language Modeling with the World Wide Web. In: Proc. International Conference on Acoustics, Speech, and Signal Processing, 2001.