Semantic Similarity Measurement between Words using Lexical Patterns

D. Hema Latha[1], D. Linga Reddy[2]
[1]Dept of Computer Science, Osmania University College For Women (OUCW)
Koti, Hyderabad, India
E mail : hlatha059@gmail.com
[2] Dept. of Physics, UCS, Osmania University, Hyderabad, India.
Email : dlreddy_phy@yahoo.com

Abstract - **Semantic similarity measurement between words is a tedious task in web mining, information extraction and natural language processing. The semantic similarity measurement between entities is required in Web mining applications such as community extraction, identification of relations etc. In this paper, the authors proposed an automatic approach to evaluate the logical or semantic similarity between words or entities with the help of web search engines. To describe distinct word co-occurrence measures and to integrate these with lexical patterns, page counts are used. In order to identify meaningful relationships between two given words, the authors proposed a new pattern extraction algorithm and a pattern clustering algorithm. Vector Support Machine (VSM) is used to acquire the optimal combination of page counts-based co-occurrence measures and lexical pattern clusters. The proposed method overcomes various previously proposed web-based similarity measures on the benchmark data sets that showed a high correlation with human ratings.**

Index terms - **Lexical Pattern, Web mining, Information Extraction**

I. INTRODUCTION

Semantic similarity, logical or meaningful association is a conception where a set of documents or terms within term lists are assigned a metric that is based on the likeness of their meaning or semantic content.

Given a group of words, similarity between two words can be calculated by the length of the shortest path that connects the two words in the group. Multiple paths may exist between the two words, if the word has many meanings. In such cases, only the shortest path between any two perceptions of the words is considered for similarity calculation.

The problem with this approach is that it depends on the concept that all the paths or links in the group represent a uniform distance. Resnik proposed a similarity calculation approach based on the content of the information. He described the similarity between two conceptions 'Y1' and 'Y2' in the group of words as the maximum of the information content of all concepts 'Y' that include both 'Y1' and 'Y2'. Then, the affinity between two words is described as the maximum of the similarity between any concepts that the words belong to. He used Word Net as the group or taxonomy and calculated the information content using the Brown corpus.

Li et al. combined structural semantic information from a lexical taxonomy and information content from a corpus in a nonlinear model. They proposed a similarity measure that uses short sighted length profundity depth and local compactness or density in taxonomy. Their experimental study reported a high Persuasion correlation coefficient of 0.8914 on the Miller and Charles example and reference or benchmark data set. They did not appraise or calculate their methodology in terms of similarities among named entities. Cilibrasi and Vitanyi proposed a distance metric between words using only page counts retrieved from a web search engine.

II. DESIRABLE FEATURES FOR RELATEDNESS MEASURE

Desirable features to measure semantic similarity in current Semantic Web applications.

1. Domain Independence: Presently, an increasing amount of online ontological and semantic data is available on the World Wide Web, enabling a new generation of semantic applications. If that kind of domain independent applications to be developed, this increasing heterogeneity should be dealt, without establishing the ontologisms to be accessed in advance.

2. Universality: The semantic measures, in the dynamic context of the Web, must be flexible, compatible and general enough to be used independently for their final purpose, and without relying on specific lexical resources or knowledge representation languages.

3. Maximum coverage: Maximum coverage of possible interpretations of the words must be warranted, in the context of web applications with no predefined domain. If it is limited to a particular knowledge source, such as WordNet2, or a certain set of ontology, then one is compelled to use those applications only.

III. LITERATURE SURVEY

Given a set of words, a direct approach to calculate similarity between two words is to compute the length of the shortest path connecting the two words in the set. If a word has many meanings (polysemy), then numerous paths might exist between the two words. In such cases, only the nearest path between any two perceptions of the words is considered for calculating similarity. A problem with this approach is that all paths or links in the group of words

represent a uniform distance. Resnik[8] proposed a logical resemblance measure using information content. He described the similarity between two conceptions 'Y1' and 'Y2' in the taxonomy or group as the maximum of the information content of all concepts 'Y' that include both 'Y1' and 'Y2'. Then, the affinity or closeness between two words is defined as the maximum of the similarity between any concepts that the words belong to. He used Word Net as the taxonomy; information content is calculated using the Brown corpus. Li et al. [9] combined organized meaningful information from a lexical taxonomy and information content from a corpus in a non-linear model. They proposed a similarity metric that uses short sighted length, profundity or depth, and local compactness or density in group. Their experimental study reported a high persuasion correlation coefficient of 0.8914 on the Miller and Charles [10] example, reference or benchmark data set. They did not appraise or calculate their methodology in terms of similarities among named entities. Lin [11] defined the similarity between two concepts as the information that is in common to both concepts and the information contained in each individual concept. Cilibrasi and Vitanyi [12] proposed a distance metric between words using only page counts retrieved from a web search engine.

IV HISTORY

The searching process shown in fig.1 gives the results based on the text search algorithms. Present all Search Engines on the web are based on the text search algorithms.

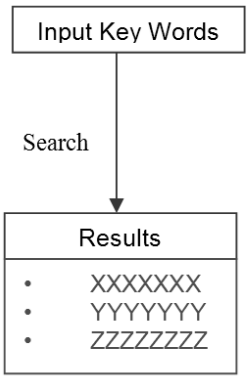

Figure 1. Generic search process

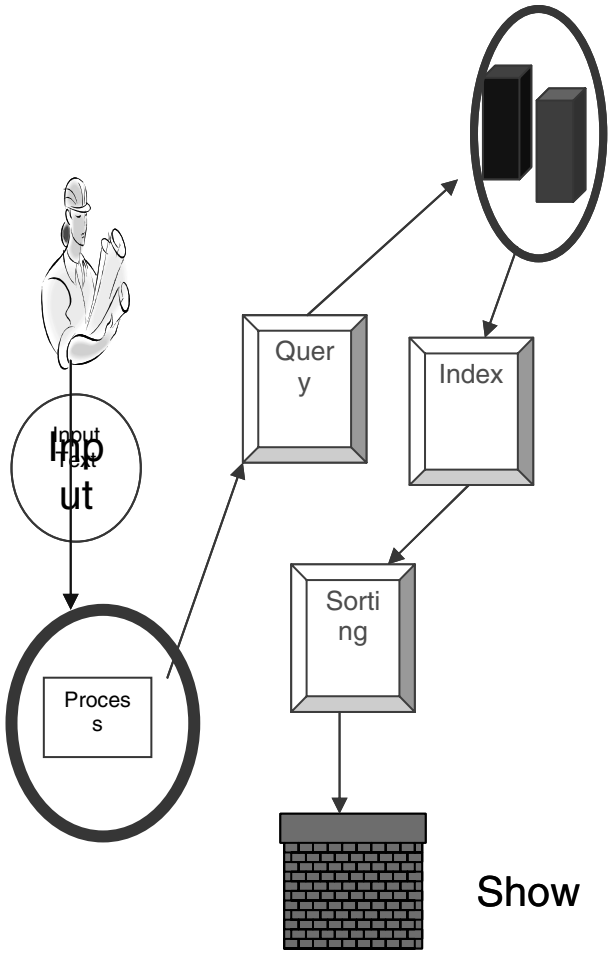

Figure 2. Generic Search Process Model

Fig. 1 shows the generic search process model

- A similarity measure can represent the similarity between two documents, two queries, or one document and one query
- It is possible to arrange the extracted documents in the order of presumed importance that is ranking the extracted documents in the order of presumed importance
- A similarity measurement is a strategy which computes the degree of similarity between a pair of text objects
- Many number of similarity measures are proposed in the literature, because the best similarity measure doesn't exist (yet!)

V. VECTOR-SPACE MODEL-VSM

1960s Salton provided Vector Space Model, which has been victoriously or favorably applied on SMART (a text searching system).

DOI:10.32377/cvrjst0814

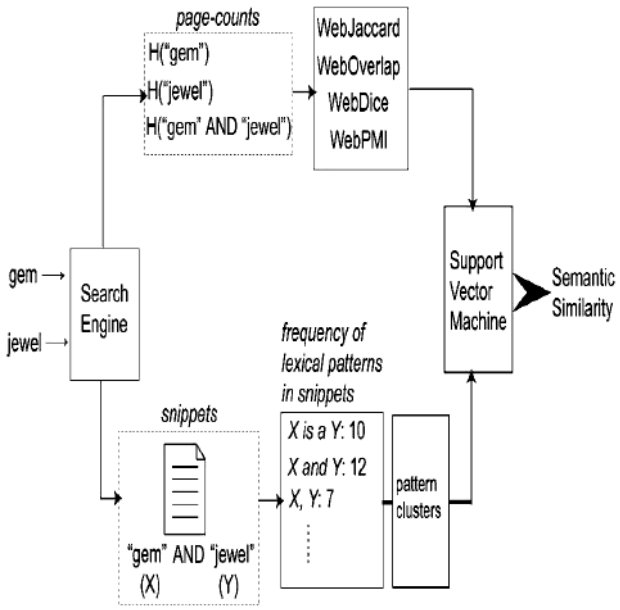

Figure. 3 Architecture of Vector Space Model

The Vector Space Model architecture is shown in fig.3

The vector space model procedure is categorized in to **three phases**.

- The **first** is the **document indexing** phase in which content bearing terms are extracted from the document text.
- An **indexed term weighting** is the second phase enhances the retrieval of document relevant to the user.
- The last phase provides ranking **to the document** with respect to the query according to a similarity measure.

From fig.3 'gem' and 'jewel' are given as input to web search engine. The search engine search for pages gem, jewel and gem as jewel in the form of page counts.
It displays text snippets 'gem' (X) and 'jewel' (Y).
The frequency of occurrence of lexical patterns is measured based on the snippets.
These patterns are clustered and fed to Vector Support Machine (VSM) as input.
The Vector Support Machine (VSM) allots weights and ranking is done based on the weights.

A. Document Indexing

In the document non significant words may appear, by using document indexing these non-significant words (function words) is removed so that the document is represented by content bearing words. This document indexing is done based on the frequent occurrence of the terms, where low frequency terms within a document are considered to be function words. Stopping list is used to remove high frequency words (stop words) which hold common words, which makes the indexing method language dependent. With the help of stop list 40% - 50% of the total number of words in a document is removed.

Probability Indexing is used which shows the statistical difference in the distribution of content bearing words, and the function words. Probabilistic indexing ranks the terms with respect to the term frequency in the entire collection. The function words are prototyped by a Poisson distribution in the overall documents, as content bearing terms cannot be prototyped. Recently, an automatic indexing method which uses serial clustering of words in text has been introduced. The value of such clustering is an indicator if the word is content bearing.

B.Term Weighting

Term weighting has been described in terms of recall and precision. There are three main components for calculating term weighting - term frequency component, collection frequency component and length normalization component. These three components are multiplied together to make the resulting term weight.

A common weighting scheme for terms within a document utilizes the frequency of occurrence as mentioned by Luhn. The term frequency for documents is generally used as the basis of a weighted document vector. It is also possible to use binary document vector, but the results are not that good when compared to the term frequency when using the vector space model.

Different weighting schemes are available to discriminate one document from the other. In general this component is called accumulation or collection frequency document. Most of them, e.g. the inverse document frequency, assume that the importance of a term is proportional with the number of document the term appears in. Experimentally it has been shown that these document discrimination factors lead to a more effective extraction, i.e., an improvement in precision and recall.

The third possible weighting factor is document length normalization factor. Lengthy documents have usually a much greater term set than small documents, which makes lengthy documents to be retrieved faster than small documents.

Experiments have been done on various weight schemes and achieved best results, with respect to recall and precision, are acquired by using term frequency with inverse document frequency and length normalization.

C. Similarity Coefficients

The associative coefficients determine the similarity in vector space model and these associative coefficients are dependent on the inner product of the document vector and query vector, and the similarity is indicated by the word overlap. The inner product is usually normalized. The most popular similarity measure is the cosine coefficient, which measures the angle between the document vector and the query vector.

VI. RESEARCH ELABORATION

- This paper contains of four page-count-based similarity scores and automatically extracted lexico-syntactic patterns from text snippets.
- Most web search engines provide Page counts and text snippets which are the main source of information.

Few problems that may occur with Page counts are:

- Page count perusal overlook the position of a word in a page
- Two words appear in a page, they might not be related with each other
- Polysemous word (a word with multiple senses). For example:
  - *apple* as a fruit
  - *apple* as a computer

Lexico-syntactic patterns

The various semantic relations *also known as,*

- *is a,*
- *part of,*
- *is an example of*

Page-count-based Similarity Scores (co-occurrence measures)

$$\text{WebJaccard}(P,Q) = \begin{cases} 0 & \text{if } H(P\cap Q) \le c \\ \frac{H(P\cap Q)}{H(P)+H(Q)-H(P\cap Q)} & \text{otherwise.} \end{cases} \quad (1)$$

$$\text{WebOverlap}(P,Q) = \begin{cases} 0 & \text{if } H(P\cap Q) \le c \\ \frac{H(P\cap Q)}{\min(H(P),H(Q))} & \text{otherwise.} \end{cases}$$

$$\text{WebDice}(P,Q) = \begin{cases} 0 & \text{if } H(P\cap Q) \le c \\ \frac{2H(P\cap Q)}{H(P)+H(Q)} & \text{otherwise.} \end{cases} \quad (3)$$

$$\text{WebPMI}(P,Q) = \begin{cases} 0 & \text{if } H(P\cap Q) \le c \\ \log_2\left(\frac{\frac{H(P\cap Q)}{N}}{\frac{H(P)}{N}\frac{H(Q)}{N}}\right) & \text{otherwise.} \end{cases} \quad (4)$$

VII. ALGORITHM FOR EXTRACTING PATTERNS

Given a set *S* of synonymous

Algorithm 3.1: EXTRACTPATTERNS(S)

comment: Given a set S of word-pairs, extract patterns.

for each word-pair $(A,B) \in S$
  do $D \leftarrow$ GetSnippets("A B")
$N \leftarrow null$
for each snippet $d \in D$
  do $N \leftarrow N +$ GetNgrams(d,A,B)
$Pats \leftarrow$ CountFreq(N)
return $(Pats)$

Figure.4: Pattern extracts from text snippets

  - n-grams : n=2,3,4, and 5

- A set *S* of closely associated word-pairs
  - 5000 word pairs of closely associated nouns from Word Net
  - 4,562,471 unique patterns
  - 80% occur less than 10 times
- A set of non-associated word-pairs
  - 5000 word pairs of non- associated nouns from Word Net
  - nouns from WordNet

Table 1: Contingency table

	v	other than v	All
Freq. in snippets for synonymous word pairs	p_v	$P-p_v$	P
Freq. in snippets for non-synonymous word pairs	n_v	$N-n_v$	N

$$\chi^2 = \frac{(P+N)(p_v(N-n_v)-n_v(P-p_v))^2}{PN(p_v+n_v)(P+N-p_v-n_v)}. \quad (5)$$

DOI:10.32377/cvrjst0814

Integrating Patterns and Page Counts

Algorithm 3.2: GETFEATUREVECTOR(A, B)

comment: Given a word-pair A, B get its feature vector F.

$D \leftarrow$ GetSnippets("A B")
$N \leftarrow null$
for each snippet$d \in D$
do $N \leftarrow N +$ GetNgrams(d, A, B)
$SelPats \leftarrow$ SelectPatterns$(N, GoodPats)$
$PF \leftarrow$ Normalize$(SelPats)$
$F \leftarrow [PF, WebJaccard, WebOverlap, WebDice, WebPMI]$
return (F)

Figure. 5: Integrating patterns and page counts

VIII. EXPERIMENTAL RESULTS

- WebOverlap (rank=18,weight=2.45)
- Web-Jaccard (rar weight=0.618)
- WebPMI (rank=138,weight=0.0001)

Table 2: Features with the highest SVM linear kernel weights

feature	χ^2	SVM weight
WebDice	N/A	8.19
X/Y	33459	7.53
X, Y :	4089	6.00
X or Y	3574	5.83
X Y for	1089	4.49
X , the Y	1784	2.99
with X (Y	1819	2.85
X=Y	2215	2.74
X and Y are	1343	2.67
X of Y	2472	2.56

Table3: Semantic Similarity of Human Ratings and Baselines on Miller-Charles' Dataset

Word Pair	Miller-Charles'	Web Jaccard	Web Dice	Web Overlap	Web PMI	Sahami [36]	CODC [6]	Proposed SemSim
cord-smile	0.13	0.102	0.108	0.036	0.207	0.090	0	0
rooster-voyage	0.08	0.011	0.012	0.021	0.228	0.197	0	0.017
noon-string	0.08	0.126	0.133	0.060	0.101	0.082	0	0.018
glass-magician	0.11	0.117	0.124	0.408	0.598	0.143	0	0.180
monk-slave	0.55	0.181	0.191	0.067	0.610	0.095	0	0.375
coast-forest	0.42	0.862	0.870	0.310	0.417	0.248	0	0.405
monk-oracle	1.1	0.016	0.017	0.023	0	0.045	0	0.328
lad-wizard	0.42	0.072	0.077	0.070	0.426	0.149	0	0.220
forest-graveyard	0.84	0.068	0.072	0.246	0.494	0	0	0.547
food-rooster	0.89	0.012	0.013	0.425	0.207	0.075	0	0.060
coast-hill	0.87	0.963	0.965	0.279	0.350	0.293	0	0.874
car-journey	1.16	0.444	0.460	0.378	0.204	0.189	0.290	0.286
crane-implement	1.68	0.071	0.076	0.119	0.193	0.152	0	0.133
brother-lad	1.66	0.189	0.199	0.369	0.644	0.236	0.379	0.344
bird-crane	2.97	0.235	0.247	0.226	0.515	0.223	0	0.879
bird-cock	3.05	0.153	0.162	0.162	0.428	0.058	0.502	0.593
food-fruit	3.08	0.753	0.765	1	0.448	0.181	0.338	0.998
brother-monk	2.82	0.261	0.274	0.340	0.622	0.267	0.547	0.377
asylum-madhouse	3.61	0.024	0.025	0.102	0.813	0.212	0	0.773
furnace-stove	3.11	0.401	0.417	0.118	1	0.310	0.928	0.889
magician-wizard	3.5	0.295	0.309	0.383	0.863	0.233	0.671	1
journey-voyage	3.84	0.415	0.431	0.182	0.467	0.524	0.417	0.996
coast-shore	3.7	0.786	0.796	0.521	0.561	0.381	0.518	0.945
implement-tool	2.95	1	1	0.517	0.296	0.419	0.419	0.684
boy-lad	3.76	0.186	0.196	0.601	0.631	0.471	0	0.974
automobile-car	3.92	0.654	0.668	0.834	0.427	1	0.686	0.980
midday-noon	3.42	0.106	0.112	0.135	0.586	0.289	0.856	0.819
gem-jewel	3.84	0.295	0.309	0.094	0.687	0.211	1	0.686
Correlation	1	0.259	0.267	0.382	0.548	0.579	0.693	0.834

IX. CONCLUSIONS

In this work, the authors discussed the problem of semantic similarity measure for words based on both page counts and text snippets which are extracted from a web. Four different word co-occurrence measures were computed using page counts.

Lexical pattern extraction algorithm is proposed that can extract various semantic relations that exist between two words. Moreover, a sequential pattern clustering algorithm is also proposed in order to identify different lexical patterns that describe the same logical or meaningful relation. To define similarity features for a word pair, both page counts-based co-occurrence measures and lexical pattern clusters were used. A two-class Support Vector Machine is used for the features that extracted for synonymous and non-synonymous word pairs that are selected from WordNet synsets. Experimental results on three referenced or benchmark data sets shows that the proposed method outperforms various baselines as well as previously proposed web-based semantic similarity measures, achieving a high correlation with human ratings. Moreover, the proposed method improved the F-score in a community mining.

REFERENCES

[1] A. Kilgarriff, "Googleology Is Bad Science," Computational Linguistics, vol. 33, pp. 147-151, 2007.

[2] M. Sahami and T. Heilman, "A Web-Based Kernel Function for Measuring the Similarity of Short Text Snippets," Proc. 15th Int'l World Wide Web Conf., 2006.

[3] D. Bollegala, Y. Matsuo, and M. Ishizuka, "Disambiguating Personal Names on the Web Using Automatically Extracted Key Phrases," Proc. 17th European Conf. Artificial Intelligence, pp. 553 557, 2006.

[4] H. Chen, M. Lin, and Y. Wei, "Novel Association Measures Using Web Search with Double Checking," Proc. 21st Int'l Conf. Computational Linguistics and 44th Ann. Meeting of the Assoc. for Computational Linguistics (COLING/ACL '06), pp. 1009-1016, 2006.

[5] M. Hearst, "Automatic Acquisition of Hyponyms from Large Text Corpora," Proc. 14th Conf. Computational Linguistics (COLING), pp. 539-545, 1992.

[6] M. Pasca, D. Lin, J. Bigham, A. Lifchits, and A. Jain, "Organizing and Searching the World Wide Web of Facts - Step One: The One Million Fact Extraction Challenge," Proc. Nat'l Conf. Artificial Intelligence (AAAI '06), 2006.

[7] R. Rada, H. Mili, E. Bichnell, and M. Blettner, "Development and Application of a Metric on Semantic Nets," IEEE Trans. Systems, Man and Cybernetics, vol. 19, no. 1, pp. 17-30, Jan./Feb. 1989.

[8] P. Resnik, "Using Information Content to Evaluate Semantic Similarity in a Taxonomy," Proc. 14th Int'l Joint Conf. Aritificial Intelligence, 1995.

[9] D. Mclean, Y. Li, and Z.A. Bandar, "An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources," IEEE Trans. Knowledge and Data Eng., vol. 15, no. 4, pp. 871-882, July/Aug. 2003.

[10] G. Miller and W. Charles, "Contextual Correlates of Semantic Similarity," Language and Cognitive Processes, vol. 6, no. 1, pp. 1-28, 1998.

[11] D. Lin, "An Information-Theoretic Definition of Similarity," Proc. 15th Int'l Conf. Machine Learning (ICML), pp. 296-304, 1998.

[12] R. Cilibrasi and P. Vitanyi, "The Google Similarity Distance," IEEE Trans. Knowledge and Data Eng., vol. 19, no. 3, pp. 370-383, Mar. 2007.

[13] M. Li, X. Chen, X. Li, B. Ma, and P. Vitanyi, "The Similarity Metric," IEEE Trans. Information Theory, vol. 50, no. 12, pp. 3250-3264, Dec. 2004.

[14] P. Resnik, "Semantic Similarity in a Taxonomy: An Information Based Measure and Its Application to Problems of Ambiguity in Natural Language," J. Artificial Intelligence Research, vol. 11, pp. 95- 130, 1999.

[15] R. Rosenfield, "A Maximum Entropy Approach to Adaptive Statistical Modelling," Computer Speech and Language, vol. 10, pp. 187-228, 1996.

[16] D. Lin, "Automatic Retrieval and Clustering of Similar Words," Proc. 17th Int'l Conf. Computational Linguistics (COLING), pp. 768 774, 1998.

[17] J. Curran, "Ensemble Methods for Automatic Thesaurus Extraction," Proc. ACL-02 Conf. Empirical Methods in Natural Language Processing (EMNLP), 2002.

[18] C. Buckley, G. Salton, J. Allan, and A. Singhal, "Automatic Query Expansion Using Smart: Trec 3," Proc. Third Text REtreival Conf., pp. 69-80, 1994.

[19] V. Vapnik, Statistical Learning Theory. Wiley, 1998.

[20] K. Church and P. Hanks, "Word Association Norms, Mutual Information and Lexicography," Computational Linguistics, vol. 16, pp. 22-29, 1991.