The International Journal Of Engineering And Science (IJES)
|| Volume || 3 || Issue || 6 || Pages || 102-107 || 2014 ||
ISSN (e): 2319 – 1813 ISSN (p): 2319 – 1805

Text Based Hybrid Clustering Algorithm Using FAST and Semantic Similarity Algorithm

D. Thamizhmani[1,] Mrs. V. M. Gayathri[2]
[1, 2,] *Department of Computer Science and Engineering, Saveetha School of Engineering Saveetha University, Chennai, India*

----------**ABSTRACT**----------
Efficient estimation of semantic similarity between the words plays an important role in various tasks on the web such as document clustering, community mining, relation extraction, and automatic metadata extraction. An experiential method is proposed to provide a semantic based search that uses the technical English dictionary and the second is page count based metric and a text snippet based metric retrieved from a web search engine for two words. In particular, we characterize a variety of word co-occurrence measures using page counts and assimilate those with lexical patterns extracted from the text snippets. To classify the frequent semantic relations that exist amongst two set of words, we suggest a novel pattern extraction algorithm and a pattern clustering algorithm. The finest grouping of page counts-based co-occurrence measures and lexical pattern clusters is learned using support vector machines. Moreover our experimental results show that the method is reasonable and effective.

KEYWORDS: *Relation extraction, Community mining, Document clustering, Metadata extraction, Semantic similarity, Page count, Snippet*

Date of Submission: 14 May 2014 Date of Publication: 25 June 2014

I. INTRODUCTION

Data warehouses are the principal data sources for decision-making in the information systems of large businesses. Integration of data warehouses is nowadays a hot topic, as the greater amount of data and their greater statistical significance offers a broader base for decision- making and knowledge discovery. Large businesses integrate their separately developed regional warehouses, newly merged companies join their warehouses to enable the business to be run centrally and independent organizations unite their warehouses when this benefits all of them.

Typical schema-related conflicts are due to the use of different names and/or structures to describe the same information (e.g. dimension describing a hospital may be called hospital in one component data warehouse, and clinic in another) or to different specifications of the same structure.

Due to the rapid development of internet technologies, information on the web is vast with lot of veiled information; that information are interconnected with the variety of semantic relations. The semantic similarity among the words plays an important role in various tasks on the web such as relation extraction, document clustering, automatic metadata extraction and community mining. These semantic similarity relations are used to identify the concepts of extracting the useful information from the database. Eternally the semantic similarity between the words changes over the time and also domain, Word Net (a common purpose Ontology) is not effective. Through the involuntary method to guesstimate the semantic similarity by search engines is more capable. The page counts, dictionary based metric and snippets are some types of useful information provided by a search engine.

Page count for a user query is the number of web pages returned by the search engine as a result for the query troubled. And these page counts are provided that the global co-occurrences of the two words on the web search engine. Uncertainty two words require more page count formerly they are considered to be more equivalent. Nevertheless in the page counts only such as an extent of co-occurrence of two words may have lot of problems. The major one is examination of page count ignores the word place in a page. Additionally, a page count for polysemous word (a word with numerous senses) contains a mixture of entirely its senses. In accumulation, incidence of the scale and noise on the web, the words might co-occur on pages starved of being essentially related.

Snippets are the text document with the purpose of showing sample contents to the users regarding the result page of web search engine. That provides a manoeuvre synopsis about the search results and it also provides the valuable information concerning to local context of the query term. Consequently it avoids the necessitate to download the whole source document which is from the web search engine. The dealing out of snippets is the efficient since it obviates the intricacy of downloading the web pages, and which may be time consuming that will be depending on the size of web pages.

On the other hand, a broadly accredited disadvantage of using snippets is, that has the massive scale of web and large number of documents that are in the result set. Only those snippets are used for the top-ranking results and also the query will be processed with more efficiently. Thus, a snippet is resolute by a difficult combination of different factors are distinctive to the primary search engine. So for this reason, it has no guarantee, which means all the information from the web page we have to evaluate semantic similarity among the given pair of words that are enclosed by the top-ranking snippets.

In this paper, an untried method is projected that will be using the technical dictionary which is based on the metric, page counts and lexical syntactic patterns extracted from snippets that are used experimentally. These are used in this paper is to conquer the exceeding mentioned problems. And this method is one of the automatic methods to calculate approximately the semantic similarity among the words or the entities by using of web search engines. The reason that the web search engine used in this is those only having an efficient interface to the internet documents. The page counts and also the snippets are the useful information sources that can be provided only by the web search engines.

The rest of the document is ordered as follows: The section 2 introduces the related work on semantic similarity methods. And in section 3, the approach to estimate semantic similarity among the given words is provided. In section4, it demonstrates an experimental setup for semantic similarity between the words on the web search engine. In section 5, the conclusion and some of the future perspectives are presented.

II. RELATED WORK

When taxonomy of words is given, an uncomplicated technique to estimate the semantic similarity among two words is to discover the length of the shortest path that can connect the two words in the given taxonomy. The multiple paths may exist among the two words, if the given word is polysemous. One problem in this approach is, it relies on the notion of that all links in the taxonomy is represent a uniform distance.

Imen Akermi offered a novel semantic similarity measure between the two words is by using an online English dictionary which is provided by the Semantic Atlas project of the French National Centre for Scientific Research and page counts that is resumed by social website Digg.com in which the content can be generated by the users. And in the projected work, the polysemy and the semantic disambiguation problem also has been dealt.

Li et al proposed the fundamental semantic information is by using a nonlinear model from a corpus for the taxonomy of words and the information content. In this the similarity measures uses the shortest path length and depth and the local density in the taxonomy. The proposed work fails to evaluate the similarity process in terms of named entities.

Resnik et al provides another way to compute the similarities among the words which is based on the information content. And in this approach, similarities among the two concepts are built that is based on the range of common information that they share. If the two concepts have more shared information then and there are deliberated as a extremely detailed content. In case of multiple inheritances, Word similarity is also taken into account. The mostly documented problem in this method is that this method did not measure the word sense disambiguation. Consequently it produces the similarity extent for the words on the origin of dissimilar word senses.

III. METHOD

This method for calculation of semantic similarity among the words uses the technical dictionary, page counts and text snippets those can be provided by the most web search engines. And this proposed method has been designed to give great accuracy when measuring the semantic similarity between the words. It has three phases:

Calculation of the similarity among two words based on the technical dictionary (TD).
Calculation of the similarity (SPS) among two words based on the page counts and text snippets retrieved from a web search engine.

Integrating the two similarity measures SD and SPS.

Phase 1: Technical Dictionary based similarity measure

In this phase, technical alternative words for every single word are extracted from the technical English dictionary. The TD differentiates the technical meaning of words from the normal meaning and also it is used to extract the set of technical synonyms for each and every word in the web page. If the two sets of meanings for every word are collected then the degree of similarity will be S (w1, w2) is computed using the Jaccard coefficient:

$$S(w1, w2) = \frac{m_c}{m_{w1} + m_{w2} - m_c}$$

Where,
mc: Number of common words between the two synonyms set
mw1: Number of words contained in the w1 synonym set
mw2: Number of words contained in the w2 synonym set

Uncertainty, the group of synonyms for the words w1unambiguously contains the word w2or vice versa, it allots directly the value of 1 to S(w1,w2).

Phase 2: Page counts and Snippets similarity measure

In this, the four page count based co-occurrences measures the WebJaccard, Web Dice, Web Overlap and WebPMI and those are defined by using the page counts and put together those throughout the lexical patterns that are extracted from the snippets. Lexical pattern extraction algorithm and Sequential pattern clustering algorithm is proposed for mining the lexical patterns. The best possible combination of co-occurrences measures and the lexical pattern clusters is learned by using Support Vector Machines (SVM).

Page Count-Based Co-occurrence Measures

Page count based co-occurrences P and Q for the two words are measured on the web. In this the P AND Q alone can not express the semantic similarity. The four page count based co-occurrences WebJaccard, WebOverlap, WebDice and Web Pointwise mutual information (WebPMI) are computed to calculate the semantic similarity among the words. The *WebJaccard* coefficient among words P and Q will be defined as:

Web Jaccard (P, Q)

$$= \begin{cases} 0, & if H(P \wedge Q) \leq c \\ \dfrac{H(P \wedge Q)}{H(P) + H(Q) - H(P \wedge Q)}, & otherwise \end{cases}$$

Where $P \wedge Q$ denotes the conjunctive query of P AND Q. This is probable that two words may seem taking place on related pages even they are not related owing to occurrence of the scale and noise in the web based content. To modest these dissimilar effects, the WebJaccard coefficient has been set to zero and in this condition the page count of the query $P \wedge Q$ will be lesser than the threshold c. Now set c=5 experimentally.

Then the web dice coefficient is the variant of the Dice coefficient. The WebDice (P, Q) is defined as:

$$\text{Web Dice (P, Q)} = \begin{cases} 0, & if H(P \wedge Q) \leq c \\ \dfrac{2H(P \wedge Q)}{H(P) + H(Q)}, & otherwise \end{cases}$$

The Web Overlap is a usual modification to the Overlap or Simpson coefficient. Web Overlap (P, Q) will be defined as:

Web Overlap(P,Q)

$$= \begin{cases} 0, & if H(P \wedge Q) \leq c \\ \frac{H(P \wedge Q)}{\min (H(P), H(Q))}, & otherwise \end{cases}$$

The Web PMI is the variant of pointwise mutual information by using page counts will be defined as:

Web PMI (P, Q)

$$= \begin{cases} 0, & if H(P \wedge Q) \leq c \\ log_2 = \left[\frac{\frac{H(P \wedge Q)}{N}}{\frac{H(P)}{N} \frac{H(Q)}{N}} \right], & otherwise \end{cases}$$

Where, N is the number of documents indexed by search engine. Then set N=1010 according to the number of indexed pages which is resumed by Google.

Lexical Pattern Extraction

Snippets encompass a window of text chosen from the document which contains the queried words. A user can read the snippet text document and can decide whether the particular search result is appropriate, without opening the URL. The web search engine may produces a snippet by selecting the numerous text fragments from contradictory portion in a document. A predefined delimiter will be used to differentiate the different fragments. For example, in Google the delimiter can be used to separate the different fragments in the snippets. On behalf of a snippet can be retrieved for a word pair (P, Q), to exchange the two words P and Q with two variables X and Y. After that replace all the numeric values by D, a marker for the digits. Then, formulate all subsequences of words from δ which satisfies the below subsequent conditions:

a. Subsequence's should contain precisely one occurrence of each X and Y.
b. Maximum length of the subsequence is L words.
c. Subsequence's can omit one or more words. Nevertheless it should not omit more than g number of words repeatedly.
d. All the negation contractions in a context are prolonged.

For example, "didn't" is expanded to "did not". Do not omit the word "not" when generating subsequences. At the end the frequency of all generated subsequences is counted and only the subsequences that occur more than T times are used as a lexical pattern. Then set L=5, g=2, G=4 and T=5 experimentally.

Lexical Clustering Algorithm

Contradictory patterns that articulate the indistinguishable semantic relation are extracted inexorably by using the sequential pattern clustering algorithm. To classify the various patterns that state the identical semantic relation who supports to exemplify the semantic relation between two words precisely.

Algorithm 1: Grouping of different lexical patterns
Input: Set of lexical patterns, threshold
Output: Clustering of patterns

Method: SORT the patterns into descending order of their total occurrences in all word pairs. The total occurrence μ (a) of a pattern "a" is the sum of the frequency of occurrences of the pattern in all word pairs. μ (a) is given by,

$$\mu(a) = \sum_i f(P_i, Q_i, a)$$

1. Initialize the set of clusters, *C* to the empty set
2. The outer for loop, frequently takes a pattern ai from the ordered set of lexical patterns
3. Set max value to ∞
4. Set the most similar cluster *c** to null
5. The inner for loop finds the cluster *c** (∈*C)* that is most similar to the pattern ai
6. First, represent a cluster by the centroid of all word-pair frequency vectors corresponding to the patterns in that cluster to compute the similarity between the pattern and a cluster. Next, compute the cosine similarity between the cluster centroid (cj) and the word-pair frequency vector of the pattern (ai)

7. If the similarity between a pattern ai, and its most similar cluster *c**, is greater than the threshold θ, append ai to *c**.
8. After that outline a new cluster {ai} and append it to the set of clusters C, if ai is not similar to any existing clusters beyond the threshold θ.

Measuring semantic similarity

The machine learning method is used to unite both page count based co-occurrence measures and the snippet based measures to erect a stout semantic similarity measure. Given N clusters of lexical patterns, first represent a pair of words (P, Q) by an (N+4) dimensional feature vector fPQ. And the four page counts based co-occurrence method is used as a four different characteristics in fPQ. After that the feature from each of N clusters is computed as follows: at first, a weight wij is assigned to a pattern ai that is in a cluster cj as follows:

$$w_{ij} = \frac{\mu(a_i)}{\sum_{t \epsilon c_j} \mu(t)}$$

Where, μ(*a*) is the total frequency of a pattern *a* in all word pairs. At last, compute the value of jth feature in the feature vector for a word pair (P, Q) as per the equation below:

$$\sum_{a_i \epsilon c_j} w_{ij} \; f(P, Q, a_i)$$

The value of the jth feature of the feature vector fPQ representing a word pair (P, Q) is the weighted sum of all patterns in cluster cj that co-occur with words P and Q. All patterns in a cluster represent the same semantic relation.

Using these features a two class support vector machine (SVM) is accomplished to notice the synonyms and non-synonyms word pair. Training data set S is produced inevitably from WordNetsynsets. Subsequently training a SVM using synonyms and non-synonyms word pairs use is to compute the semantic similarity between two given words. LibSVM is used as the SVM implementation.

Phase 3: The overall similarity measure

In this phase, the two similarity measures (technical dictionary based metric, page count and text snippets based metric) are integrated for more accurately measuring the semantic similarity between words. In the proposed work the input given to the machine is a collection of word pairs. The synonymous words are extracted directly from the synsets and the non-synonymous words are generated by a random shuffling technique. Once the machine gets trained it can be used to compute the similarity between the words. After computing the similarity between the words it is used to design a semantic search engine which in turn returns the semantically related results for the user query.

IV. EXPERIMENTAL SETUP

The proposed method has been evaluated against Miller-Charles dataset, a dataset of 30 word pairs. The evaluation of the results of the proposed method with Chen Co-occurrence Double Checking(CODC) measure with Sahami and Heilman metric, Normalised Google Distance(NGD), No Clust that means it does not expenditure any clustering information in the feature vector conception and with four popular co-occurrence measures WebJaccard, WebOverlap, Web Dice, and WebPMI. This measure uses the page counts reverted by the search engine. The suggested method uses the Spearman coefficient and Pearson coefficient for semantic similarity estimation.

The search engine will consists of two following searches. The first one is Normal search that yields all pages for the query word. Next one is semantic search that precedes the pages which are semantically correlated to query keyword. Then the evaluation of result pages resumed by the Semantic search and Normal search is specified in below Figure. From the figure we can conclude that the semantic search provides more accuracy than the normal search.

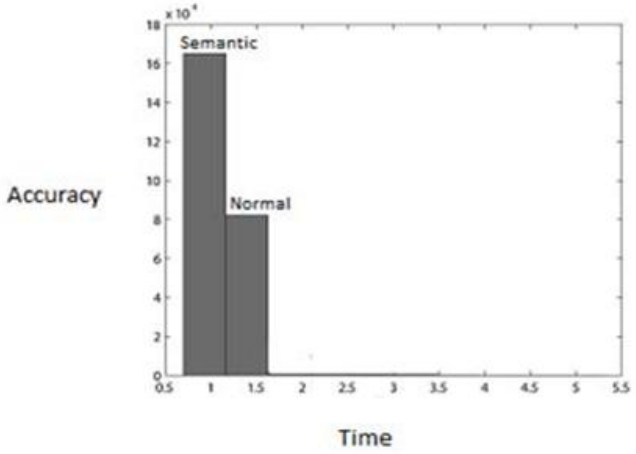

Fig. Comparison of Semantic search with Normal search

V. CONCLUSION & FUTURE WORK

Semantic similarity amongst the words is essential to several areas such as Natural Language Processing, Information Retrieval and Cognitive Science. So, depend on stout semantic similarity measure is decisive. The proposed semantic similarity measures for two words uses the page counts and snippets reclaimed from web search engine. To define the features for a word pair the page counts based co-occurrence measure and lexical pattern clusters were used. The two-class SVM was skilled by using those features that are extracted for nonsynonymous and synonymous word pairs selected from a WordNetsynsets. Experimental grades on three benchmark data sets exhibited that the proposed method outperforms the web based semantic similarity measures and it achieves the highest correlation with the human ratings. The future work includes prolonging this method to other purviews. This method will also be very interesting to covenant by image based information retrieval by using the correlation measures.

REFERENCES

[1] ImenAkermi and Rim Faiz (2012), "*Semantic similarity measure based on multiple resources*", Proceedings of the International Conference on Information Technology and e-Services, pp.546-550.

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

[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] Lapata M and Keller F (2005), "*Web-Based models for natural language processing*," ACM Transaction Speech and Language Processing, vol. 2, no. 1, pp. 1-3.

[5] Mclean D, Li Y, and Bandar Z. A (2003), "*An approach for measuring semantic similarity between words using multiple information sources,*" IEEE Transactions on Knowledge and Data Engineering, vol. 15, Issue 4, pp. 871-882.

[6] Lin D (1998), "*An Information-Theoretic definition of similarity*," Proceedings of the 15th International Conference on Machine Learning (ICML), pp. 296-304.

[7] Matsuo Y, Sakaki T, Uchiyama K and Ishizuka M (2006)," *Graph-based word clustering using web search engine*", Proceedings of EMNLP, pp.523-530.

[8] M. Hearst, "Automatic acquisition of hyponyms from large text corpora," Proc. 14th Conf. Computational Linguistics (COLING), pp. 539-545, 1992.

[9] Pei T, Han J, Mortazavi-Asi B, Wang J, Pinto H, Chen Q, Dayal U, and Hsu M (2004), "*Mining sequential patterns by Pattern-Growth: The Prefixspan Approach*," IEEE Trans. Knowledge and Data Eng., vol. 16, no. 11, pp. 1424-1440.

[10] Pasca M, Lin D, Bigham J, Lifchits A, and Jain A (2006), "*Organizing and searching the world wide web of facts - Step One: The One-Million fact extraction challenge*," Proceedings of National Conference on Artificial Intelligence (AAAI '06).