Measuring Of Semantic Similarity Between Words Using WebSearch Engine Approach

Prof.P.Pradeep Kumar,Naini.Shekhar Reddy,R.Sai Krishna,Ch.Kishor Kumar,M.Ramesh Kumar

Abstract—
Semantic similarity is the process of identifying the synonyms for a given word. Which returns the one or more words which give the same meaning in context. In dictionary the semantic similarity between words is solved. But when it comes to web, measuring the semantic similarity between words has become the challenging task.Inorder to find the semantic similarity between the words we have proposed a lexical pattern extraction algorithm to find the numerous semantic relations between two words. And also a sequential pattern clustering algorithm was proposed to find the number of lexical patterns that shows the same semantic relations between two words. Page count concurrence measures along with lexical patterns extracted from snippets are used to define features of a word pair. Testing on three benchmark desk by training two class SVM the proposed method outperformed various baselines. And also it also improved the efficiency of community mining.

1 .Introduction

Pattern matching is the concept which reveals/deals with the similarity between words. It is useful in finding the files in a folder or disk Given text in a document etc…. The concept of tries helps to achieve this and the following algorithms are used For example pattern matching algorithms like Brute force Boyer moore Knuth-morris Semantic similarity is the process of identifying the synonyms for a given word. Which returns the one or more words which give the some meaning in context. In dictionary the semantic similarity between words is solved. But when it comes to web, measuring the semantic similarity between words has become the challenging task. This can be achieved by the below mentioned methods**.** Depending upon frequent usage of words we can make the semantic relation between words. Let us consider an example:- Web search engine based approach gives the results for the words like apple and computer. The page count of the query "apple" and "computer" in Google is 2,88,000,000,whereas the same for "banana" and "computer" is only 3,590,000**.**

2 Method
Outline:

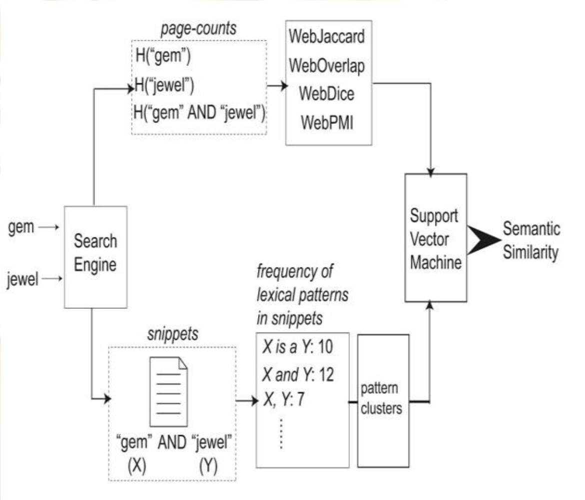

Fig. 1 Out line of the proposed method

Fig.1 illustrates an example of using the proposed method to compute the semantic similarity between two words, gem and jewel. First, we query a web search engine and retrieve page counts for the two words and for their conjunctive (i.e., "gem," "jewel," and "gem AND jewel"). In Section 3.2, we define four similarity scores using page counts. Page counts-based similarity scores consider the global co-occurrences of two words on the web. However, they do not consider the local context in which two words co-occur. On the other hand, snippets returned by a search engine represent the local context in which two words cooccur on the web. Consequently, we find the frequency of numerous lexical syntactic atterns in snippets returned for the conjunctive query of the two words. The lexical patterns we utilize are extracted automatically using the method described in Section 3.3. However, it is noteworthy that a semantic relation can be expressed using more than one lexical pattern. Grouping the different lexical patterns that convey the same semantic relation, enables us to represent a semantic relation between two words accurately. For this purpose, we propose a sequential pattern clustering algorithm in Section 3.4. Both page counts-based similarity scores and lexical pattern clusters are used to define various features that represent the relation between two words. Using this feature representation of word pairs, we train a two-class support vector machine [19].

3 Page Count

Page count of a query is an estimate of the number of pages that contain the query words. In general, page count may not necessarily be equal to the word frequency because the queried word might appear many times on one page. Page count for the query P AND Q can be considered as a global measure of cooccurrence of words P and Q. For example, the page count of the qu ery “apple” AND “computer” in Google is 288,000,000, whereas the same for “banana” AND “computer” is only 3,590,000. The more than 80 times more numerous page counts for “apple” AND “computer” indicate that apple is more semantically similar to computer than is banana. Despite its simplicity, using page counts alone as a measure of co-occurrence of two words presents several drawbacks. First, page count analysis ignores the position of a wordin a page. Therefore, even though two words appear in a page, they might not be actually related. Second, page count of a polysemous word (a word with multiple senses) might contain a combination of all its senses. For example, page counts for apple contain page counts for apple as a fruit and apple as a company. Moreover, given the scale and noise on the web, some words might co-occur on some pages without being actually related [1]. For those reasons, page counts alone are unreliable when measuring semantic similarity. Page count can be measured using four popular co-occurrence measures:- Jaccard Overlap(Simpson) Dice Point wide mutual information (PMI) to compute semantic similarity using page counts.

4 Snippets

It is a brief window of text extracted by a search engine around the query term in a document.It provides useful information regarding the local context of the query term. Snippets, a brief window of text extracted by a search engine around the query term in a document, provide useful information regarding the local context of the query term. Semantic similarity measures defined over snippets,

have been used in query expansion [2], personal name disambiguation [3], and community mining [4]. Processing snippets is also efficient because it obviates the trouble of downloading webpages, which might be time consuming depending on the size of the pages. However, a widely acknowledged drawback of using snippets is that, because of the huge scale of the web and the large number of documents in the result set, only those snippets for the topranking results for a query can be processed efficiently. Ranking of search results, hence snippets, is determined by a complex combination of various factors unique to the underlying search engine. Therefore, no guarantee exists that all the information we need to measure semantic similarity between a given pair of words is contained in the top-ranking snippets.

Drawback

Because of the huge scale of the web and the large no. of documents in the results set, only those snippets for the top ranking results for a query can be processed efficiently.

5 Lexical Syntactic Pattern

It is a pattern extracted from snippet. A method has been proposed which considers both page count co-occurrence & lexical syntactic patterns in order to overcome the above mentioned problems. These patterns have been used in various natural language processing tasks.

5.1 Lexical Pattern Extraction & Sequential Pattern Clustering Algorithms

Typically, a semantic relation can be expressed using more than one pattern. For example, consider the two distinct patterns, X is a Y, and X is a large Y. Both these patterns indicate that there exists an is-a relation between X and Y. Identifying the different patterns that express the same semantic relation enables us to represent the relation between two words accurately. According to the distributional hypothesis [29], words that occur in the same context have similar meanings. The distributional hypothesis has been used in various related tasks, such as identifying related words [16], and extracting paraphrases [27]. If we consider the word pairs that satisfy (i.e., co-occur with) a particular lexical pattern as the context of that lexical pair, then from the distributional hypothesis, it follows that the lexical patterns which are similarly distributed over word pairs must be semantically similar. We represent a pattern a by a vector a of word-pair frequencies. We designate a, the word-pair frequency vector of pattern a. It is analogous to the document frequency vector of a word, as used in information retrieval. The value of the element corresponding to a word pair ðPi;QiÞ in a, is the frequency, fðPi;Qi; aÞ, that the pattern a occurs with the word pair ðPi;QiÞ. As demonstrated later, the proposed pattern extraction algorithm typically extracts a large number of lexical patterns. Clustering algorithms based on pairwise comparisons among all patterns are prohibitively time consuming when the patterns are numerous. Next, we present a sequential clustering algorithm to efficiently cluster the extracted patterns.

Algorithm 1. Sequential pattern clustering algorithm.

```
Input: patterns _ ¼ fa1; . . . ; ang, threshold _
Output: clusters C
SORT(_)
C fg
for pattern ai 2 _ do
    max _1
    c_ null
    for cluster cj 2 C do
        sim cosineðai; cjÞ
        if sim > max then
            max sim
            c_ cj
        end if
    end for
    if max > _ then
        c_ c_ _ ai
    else
        C C [ faig
    end if
end for
return C
```

6 SVM (Support Vector Machine):

SVMs are currently among the best performers for a number of classification tasks ranging from text to genomic data SVMs can be applied to complex data types beyond feature vectors (e.g. graphs, sequences, and relational data) by designing kernel functions for such data.SVM was trained using page count co-occurrence measures, lexical pattern clustering & snippets to extract the synonymous & non-synonymous word pairs which give semantic similarity.

To train the two-class SVM described in Section 3.5, we require both synonymous and nonsynonymous word pairs. We use WordNet, a manually created English dictionary, to generate the training data required by the proposed method. For each sense of a word, a set of synonymous words is listed in WordNet synsets. We randomly select 3,000 nouns from WordNet, and extract a pair of synonymous words from a synset of each selected noun. If a selected noun is polysemous, then we consider the synset for the dominant sense. Obtaining a set of nonsynonymous word pairs (negative training instances) is difficult, because there does not exist a large collection of manually created nonsynonymous word pairs. Consequently, to create a set of onsynonymous word pairs, we adopt a random shuffling technique. Specifically, we first rand omly select two synonymous word pairs from the set of synonymous word pairs created above, and exchange two words betwe en word pairs to create two new word pairs. For example, from two synonymous word pairs ðA;BÞ and ðC;DÞ, we generate two new pairs ðA;CÞ and ðB;DÞ. If the newly created word pairs do not appear in any of the word net synsets, we select them as nonsynonymous word pairs. We repeat this process until we create 3,000 nonsynonymous word pairs. Our final training data set contains 6,000 word pairs (i.e., 3,000 synonymous word pairs and 3,000 nonsynonymous word pairs). Next, we use the lexical pattern extraction algorithm described in Section 3.3 to extract numerous lexical patterns for the word pairs in our training data set. We experimentally set the parameters in the pattern extraction algorithm to L ¼ 5, g ¼ 2, G ¼ 4, and T ¼ 5. Table 1 shows the number of patterns extracted for synonymous and nonsynonymous word pairs in the training data set. As can be seen from Table 1, the proposed pattern extraction algorithm typically extracts a large number of lexical patterns. Figs. 5 and 6, respectively, show the distribution of patterns extracted for synonymous and nonsynonymous word pairs. Because of the noise in web snippets such as, ill-formed snippets and misspells, most patterns occur only a few times in the list of extracted patterns. Consequently, we ignore any patterns that occur less than five times. Finally, we deduplicate the patterns that appear for both synonymous and nonsynonymous word pairs to create a final set of 3,02,286 lexical patterns. The remainder of the experiments described in the paper use this set of lexical patterns

Conclusion

We proposed a semantic similarity measure using both page counts and snippets retrieved from a web search engine for two words. Four word co-occurrence measures were computed using page counts. We proposed a lexical pattern extraction algorithm to extract numerous semantic relations that exist between two words. Moreover, a sequential pattern clustering algorithm was proposed to identify different lexical patterns that describe the same semantic relation. Both page counts-based co-occurrence measures and lexical pattern clusters were used to define features for a word pair. A two-class SVM was trained using those features extracted for synonymous and nonsynonymous word pairs selected from WordNet synsets. xperimental results on three benchmark data sets showed that the proposed method outperforms various baselines as well as previously proposed web-based semantic similarity measures, achieving a high correlation with human ratings.

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.