International Journal for Modern Trends in Science and Technology
Volume: 03, Issue No: 09, September 2017
ISSN: 2455-3778
http://www.ijmtst.com

Enriched Semantic Similar Frequent Patterns using SSFPOA Neighborhood Ranking Algorithm

S.Jayaprada[1] | P.Bala Krishna Prasad[2] | R.Satya Prasad[3]

[1]Sr. Assistant Professor, Department of CSE, V R Siddhartha Engineering College, Vijayawada, India.
[2]Principal, Department of CSE, Eluru College of Engineering, Eluru, India.
[3]Professor, Department of CSE, Acharya Nagarjuna University, Guntur, India.

To Cite this Article
S.Jayaprada, P.Bala Krishna Prasad and R.Satya Prasad, "Enriched Semantic Similar Frequent Patterns using SSFPOA Neighborhood Ranking Algorithm", *International Journal for Modern Trends in Science and Technology*, Vol. 03, Issue 09, September 2017, pp.-137-142.

ABSTRACT

Semantic frequent pattern is the important sub domain in the data mining. Semantic similar frequent pattern mining is one of the sub domains in the data mining. It is very important to find the accurate semantic similar frequent patterns in data mining. The fundamental task of the data mining is discovery of semantic frequent patterns. Various researches have been done on Semantic similar frequent patterns and implemented many algorithms to the better results. In this paper, an ensemble algorithm is used to discover the original and quality of the sets of semantic similar frequent patterns using SSFPOA Neighborhood Ranking Algorithm improved and also discovers the performance of the pattern formation and focus on better semantic frequent patterns. Our proposed algorithm can implement on various datasets such as gene/protein and supermarket datasets.

Keywords: *Mining, textual content, term based techniques.*

I. INTRODUCTION

Nowadays, with the rapid development of Internet, there are a huge number of documents in many different fields such as science, technology, medicine, literature, etc. Everyone can easily find documents they need. However, it has negative points as well. Many students misused these documents without customizing or writing authors. Because of these problems, measuring the similarity of two documents is very necessary and it is fundamental to detect the plagiarism of many different documents. Comparing the similarity between documents has many different purposes such as checking plagiarism, classifying text, information retrieval, automatic essay scoring.

Detecting the similarity from documents is not new field now. There are many researches about this subject with lots of different algorithms. The methods can be divided into

String-based, Corpus-based and Knowledge-based Similarities [1]. String-based measures determines the similarity by operating on string sequences and character composition. String-based method is divided into character-based and terms-based approaches. Algorithms of character-based Manuscript received April, 2015. Khuat Thanh Tung, DATIC Laboratory, The University of Danang, University of Science and Technology, Danang, Vietnam. Nguyen

Duc Hung,DATIC Laboratory, The University of Danang, University of Science and Technology, Danang, Vietnam. Le Thi My Hanh, DATIC

Laboratory, The University of Danang, University of Science and Technology, Danang, Vietnam. similarity measurement consist of Smith-Waterman, N-gram. Damerau–Levenshtein, Jaro–Winkler, Needleman–Wunsch, Jaro, and Longest Common Substring (LCS). Algorithms of term-based similarity measurement include Block Distance, Cosine similarity, Dice's coefficient, Euclidean distance, Jaccard similarity, Matching Coefficient and Overlap coefficient [1]. Corpus-based measure specifies the similarity between words according to information gained from large corpora. It contains on approaches such as Latent Semantic Analysis (LSA), Generalized Latent Semantic Analysis (GLSA), Explicit Semantic Analysis (ESA), the cross-language explicit semantic analysis (CL-ESA) [1]. Knowledge-based measure relies on identifying the degree of similarity between words using information derived from semantic networks [1]. This paper focuses on two string-based approaches which are character-based and term-based algorithms. In term-based method utilizes the cosine similarity measure [2], [3]. The character-based measure uses n-gram which is a sub-string sequence in order to find fingerprint based on two algorithms: fingerprint and winnowing [4], [5], [6], [7].

So as to remedy the above paradox, this paper offers an effective sample discovery method, which first calculates located specificities of patterns after which evaluates term weights consistent with the distribution of phrases in the observed patterns rather than the distribution in files for fixing the misinterpretation problem. It also considers the have an impact on of styles from the poor education examples to locate ambiguous (noisy) patterns and try and lessen their impact for the low-frequency problem.

The manner of updating ambiguous patterns can be referred as sample evolution. The proposed approach can enhance the accuracy of comparing term weights because found styles are more precise than complete documents.

Cosine similarity:

Cosine similarity is a measure of similarity between two non-zero vectors of an inner product space that measures the cosine of the angle between them. The cosine of 0° is 1, and it is less than 1 for any other angle. It is thus a judgment of orientation and not magnitude: two vectors with the same orientation have a cosine similarity of 1, two vectors at 90° have a similarity of 0, and two vectors diametrically opposed have a similarity of -1, independent of their magnitude. Cosine similarity is particularly used in positive space, where the outcome is neatly bounded in [0,1]. The name derives from the term "direction cosine": in this case, note that unit vectors are maximally "similar" if they're parallel and maximally "dissimilar" if they're orthogonal (perpendicular). This is analogous to the cosine, which is unity (maximum value) when the segments subtend a zero angle and zero (uncorrelated) when the segments are perpendicular.

Note that these bounds apply for any number of dimensions, and cosine similarity is most commonly used in high-dimensional positive spaces. For example, in information retrieval and text mining, each term is notionally assigned a different dimension and a document is characterised by a vector where the value of each dimension corresponds to the number of times that term appears in the document. Cosine similarity then gives a useful measure of how similar two documents are likely to be in terms of their subject matter.[1]

The technique is also used to measure cohesion within clusters in the field of data mining.[2]

Cosine distance is a term often used for the complement in positive space, that is: {\displaystyle D_{C}(A,B)=1-S_{C}(A,B),} {\displaystyle D_{C}(A,B)=1-S_{C}(A,B),} where {\displaystyle D_{C}} D_C is the cosine distance and {\displaystyle S_{C}} S_{C} is the cosine similarity. It is important to note, however, that this is not a proper distance metric as it does not have the triangle inequality property—or, more formally, the Schwarz inequality—and it violates the coincidence axiom; to repair the triangle inequality property while maintaining the same ordering, it is necessary to convert to angular distance (see below.)

One of the reasons for the popularity of cosine similarity is that it is very efficient to evaluate, especially for sparse vectors, as only the non-zero dimensions need to be considered.

Given two vectors of attributes, A and B, the cosine similarity, cos(θ), is represented using a dot product and magnitude as

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|_2 \|\mathbf{B}\|_2} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Where A_i and B_i are components of vector A and B respectively.

Metric similarity:

To qualify as a metric, a measure d must satisfy the following four conditions: Let x and y be any two objects in a set and d(x, y) be the distance between x and y [14].

• The distance between any two points must be nonnegative, that is, $d(x, y) \geq 0$.

• The distance between two objects must be zero if and only if the two objects are identical, that is, $d(x, y) = 0$ if and only if $x = y$.

• Distance must be symmetric, that is, distance from x to y is the same as the distance from y to x, ie. $d(x, y) = d(y, x)$.

• The measure must satisfy the triangle inequality, which is $d(x, z) \leq d(x, y) + d(y, z)$.

II. Related Work

This paper [7] uses 3 phase architecture for extracting Synonymous gene and protein names from biomedical text documents. In the first phase, pre-processing is carried out to prepare text document suitable for subsequent stages with the help of NLP techniques. An extended improved porter stemmer algorithm is proposed to find root words. In the second phase, we find synonyms for the extracted gene and protein names using SSFPOA measure. Finally in the third phase we construct/ update Gene database depending on its presence. Extracted synonyms will be validated and verified using performance measures such as precision, recall and F-measure.

Pair wise sequence alignment methods are used to find the best-matching pair wise local or global alignments of two query sequences [8]. Protein sequence alignment is one of the crucial tasks of computational biology which forms the basis of many other tasks like protein structure prediction, protein function prediction and phylogenetic analysis. This paper studied various Pair Wise Local alignment approaches and figured the possible extensions to each of the existing approach. The paper also proposed a new method by using different techniques such as CARD, ALAE, BLASA/Smith waterman algorithm, BLAST, FASTA at various stages to find out the protein sequences similarity. This proves to be an efficient technique to overcome the drawbacks in previous methods. Even though it takes much time, in due course we identify techniques that can reduce time.

The given [9] architecture improved the performance of stemmer algorithm by applying new rules to the original stemmer algorithm. As there is no universally accepted tokenization method for processing text documents, our future work concentrates on improving biomedical tokenization process. Our work should also consider abstracts from other biomedical journals such as Pubmed etc. Also we need to compare Gene database constructed by us with one of the existing databases such as GenBank so as to analyse whether our approach can put update-to-date information or not.

III. Proposed System

• An advanced Semantic similar frequent pattern discovery technique is discovered.

• Appraise specificities of patterns and then appraises term weights according to the distribution of terms in the discovered patterns.

• Solves 99% falsify Problem.

• Training the samples to find the noisy patterns and influence to reduce the low-frequency problem.

• In this pattern evolution, the process of updating ambiguous patterns is referred.

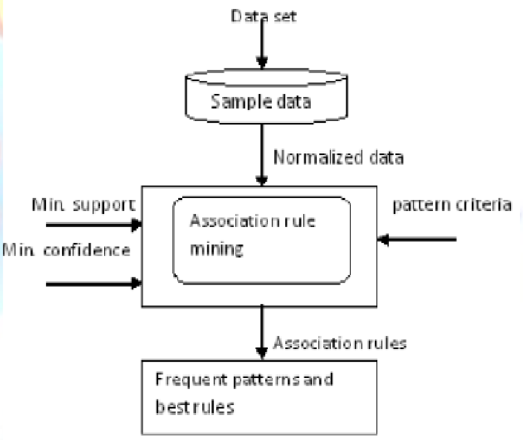


Figure 1: Basic approach for association rule mining.

• We can identify the improvement by using proposed approach by evaluating term weights because discovered patterns are more specific than whole documents.

• There are two modules in this.

• Training and Testing

• In training module, the d-patterns in the positive documents (pd) divide on min sup are identified, and evaluates term supports by deploying d-patterns to terms.

• In testing module, it will test the noise negative documents in D based on experimental coefficient.

• Based on the weights the incoming documents are sorted.

Advantages of Proposed System:

- To improve the performance of the evaluating term weights by using proposed system.
- From all the documents the identified documents are more important.
- To avoiding the issues of phrase-based approach to using the pattern-based approach.
- To find out various text patterns we use pattern mining techniques.

Algorithm:

```
1. Di is a new document. LDi is empty list
3. for each sentence S in Di do
4. for each labeled term in S do
5. if(labeled term already in the list LDi)
6. Increase labeled-term count by 1;
7. else
8. {
9. Add a new node in the list
10. Node->data=labeled-term;
11. Labeled-term count =1 12.
}
13. End for
14. End for
15. SQ is a temporary variable.
16. For each labeled term in LQi do
17. If(labeled-term in LQi==labeled-term in LDi) 18.
{
19. SQ= SQ + Labeled-term count in LDi *
Labeled-term count in LQi;
20. }
21. End for
22. Semantic similarity=SQ/sum of count of all
labeled terms in LDi;
```

Evaluation Results for Medline abstracts

We took 50 MEDLINE Abstracts from [13] http://www.biomedcentral.com/ to construct synonymous gene/protein database. The following table 1 shows the statistics of extracted gene/protein names and their cluster similarities. Netbeans 8.0.2 and IDE 1.8, 4GB RAM, Intel core i5 with 2.4 GHz processor is used for finding clusters. .arff file is created with 11 matcher values and is given to various classification techniques like Bayesnet, LibSVM and Multiplayer Perceptron. The output which represents similarity is not just 0 or 1 but in the range of 0-1 to represent different levels of similarity.

Evolution of Results for Super Market

We took 5000 transactions super market dataset and implemented with SSFPOA nearest neighborhood ranking algorithm to find the association rules with clusters.

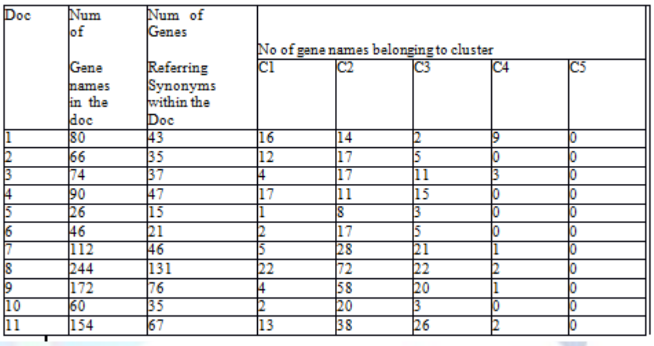
Similarity Library

The similarity is not a general library in the sense that the library is dedicated to specific semantic graph (ontologies, terminologies).

The Similarity Library aims at providing developers with a library for assessing similarity both between words and sentences. This library in an extension of the JWSL (Java WordNet Similarity Library). In the current implementation, there are two categories of similarity measures between words:

- measures exploiting ontologies such as WordNet, MeSH or the Gene Ontology.
- measures exploiting search engines.

Table 1: Semantic Similarity measures applied on Gene Synonyms



Doc	Num of Gene names in the doc	Num of Genes Referring Synonyms within the Doc	No of gene names belonging to cluster				
			C1	C2	C3	C4	C5
1	80	43	16	14	2	9	0
2	66	35	12	17	5	0	0
3	74	37	4	17	11	3	0
4	90	47	17	11	15	0	0
5	26	15	1	8	3	0	0
6	46	21	2	17	5	0	0
7	112	46	5	28	21	1	0
8	244	131	22	72	22	2	0
9	172	76	4	58	20	1	0
10	60	35	2	20	3	0	0
11	154	67	13	38	26	2	0

- Various experimental studies are performed by considering only 1 matcher, 3 matchers,5 matchers, 6 matchers,11 matchers, so as to understand the importance of each matcher in the semantic measure. The following Table-2,3,4 represents the comparison of accuracy in results while using a combination of matchers for various classifiers.

Table 2: Comparison of accuracy when using different matchers (Bayesnet)

Sno	Number of matchers Used	Precision	Recall	F-Measure
1	1-matcher (Exact matcher)	.267	.467	.326
2	3-matchers (Exact, soundex, synonymn matchers)	.258	.478	.341
3	5-matchers matchers (Exact, soundex, synonymn, trigram, boyer-moore matchers)	.587	.661	.601
4	6-matchers(Levenstein Distance, Smith-waterman, Needleman-Wunsch,Monge-Elkan Distance , Stoilos Similarity,Jaro-winkler)	.559	.704	.614
5	11- matchers (Levenstein			

	Distance, Smith-waterman,Needleman-Wunsch, Monge-Elkan measure, Stoilos Similarity,Boyer-Moore, synonymn, Soundex, Tri-Gram , Exact matcher,Jaro-winkler)	0.757	0.8	0.757

- The accuracy is calculated according to Precision,Recall, and Fmeasure .Then compare the results using different
- matchers in SVM classifier is shown in Table-3

Table 3: Comparison of accuracy when using different matchers (SVM)

Sno	Number ofmatchers Used	Precision	Recall	F-Measure
1	1-matcher (Exact matcher)	.285	.488	.347
2	3-matchers (Exact, soundex, synonymn matchers)	.195	.424	.284
3	5-matchers matchers (Exact, soundex, synonymn, trigram, boyer-moore matchers)	.721	.661	.653
4	6-matchers(Levenstein Distance, Smith-waterman, Needleman-Wunsch,Monge-Elkan Distance , Stoilos Similarity,Jaro-winkler)	.683	.768	.707
5	11-matchers (Levenstein Distance, Smith-waterman,Needleman-Wunsch, Monge-Elkan measure, Stoilos Similarity,Boyer-Moore, Soundex, synonymn, Tri-gram , exact matcher,Jaro-winkler)	.646	.774	.694

- The accuracy is calculated according to Precision,Recall, and Fmeasure .Then compare the results using different
- matchers in MultiLayer classifier is shown in Table-4

Table 4: Comparison of accuracy when using different matchers (MultiLayer)

Sno	Number of matchers used	Precision	Recall	F-Measure
1	1-matcher (Exact matcher)	.269	.473	.331
2	3-matchers (Exact, soundex, synonymn matchers)	.182	.409	.270
3	5-matchers matchers (Exact, soundex, synonymn, trigram, boyer-moore matchers)	.662	.625	.601
4	6-matchers(Levenstein Distance, Smith-waterman, Needleman-Wunsch,Monge-Elkan Distance , Stoilos Similarity,Jaro-winkler)	.376	.558	.433
5	11- matchers (Levenstein Distance, Smith-waterman,Needleman-Wunsch, Monge-Elkan measure, Stoilos Similarity,Boyer-Moore, synonymn, Soundex, Tri-gram , exact matcher,Jaro-winkler)	.825	0.9	0.857

One problem noticed while evaluating SSFPOA is some of the Medline abstracts taken by us involve more number of cryptic.Pre-processing phase does not replace these cryptic with corresponding full names such as (LARD - lymphocyte associated receptor of death). This was reflected while evaluating with 1 matcher (exact matcher) as shown in table 2. Mapping cardinality of 1:1 (simple) is used in this paper and have not resolved 1:n (complex) mappings.

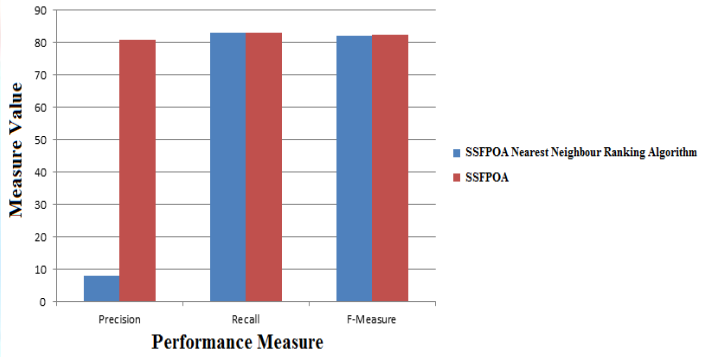


Fig 2: Overall performance of SSFPOA nearest neighbour ranking algorithm for medline abstracts

The following Fig.2 shows a graph of overall performance of SSFPOA on the 10 Medline Abstracts when compared to [12].

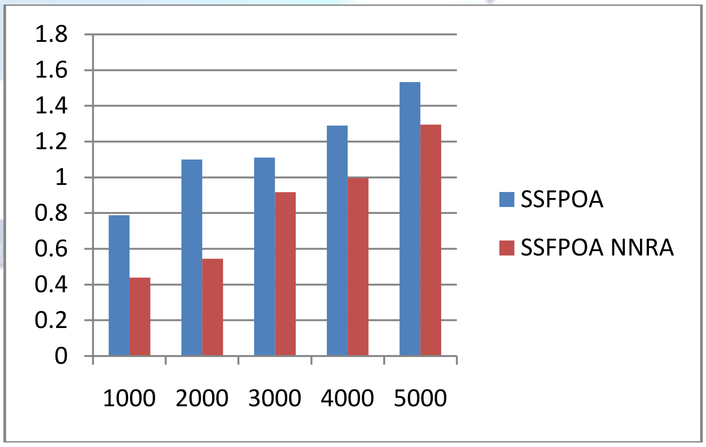


Fig 3: The following fig shows the performance of the proposed system using super market dataset dynamically I,e for every few milli seconds the proposed system forms the association rules forms and over performance of the proposed system shown in above fig.

V. Conclusion

The proposed work mainly focus on Enriched Semantic similar frequent patterns to discover the accurate matching function. Fingerprint, winnowing algorithms and the cosine similarity were widely used to compare documents because they are easy to understand and use. Still there is a lack (i,e low frequency) of identifying the similar patterns by using above data mining techniques. In this proposed work, we have mainly focus on discover the Enriched Semantic similar frequent pattern SSFPOA nearest neighbour ranking algorithm on dynamic datasets. The proposed system implements two processes, pattern checking and pattern implementation, to extract the efficient discovered patterns in text documents. The experimental results shows the performance of the Enriched Semantic Similar frequent pattern mining algorithm is based on outperforms no longer most effective different natural statistics mining-primarily based strategies and the concept based model.

References

[1] K. Aas and L. Eikvil, “Text Categorisation: A Survey,” Technical Report Raport NR 941, Norwegian Computing Center, 1999.

[2] R. Agrawal and R. Srikant, “Fast Algorithms for Mining Association Rules in Large Databases,” Proc. 20th Int’l Conf. Very Large Data Bases (VLDB ’94), pp. 478-499, 1994.

[3] H. Ahonen, O. Heinonen, M. Klemettinen, and A.I. Verkamo, “Applying Data Mining Techniques for Descriptive Phrase Extraction in Digital Document Collections,” Proc. IEEE Int’l Forum on Research and Technology Advances in Digital Libraries (ADL ’98), pp. 2-11, 1998.

[4] R. Baeza-Yates and B. Ribeiro-Neto, Modern Information Retrieval. Addison Wesley, 1999.

[5] N. Cancedda, N. Cesa-Bianchi, A. Conconi, and C. Gentile, “Kernel Methods for Document Filtering,” TREC, trec.nist.gov/ pubs/trec11/papers/kermit.ps.gz, 2002.

[6] [6] N. Cancedda, E. Gaussier, C. Goutte, and J.-M. Renders, “Word- Sequence Kernels,” J. Machine Learning Research, vol. 3, pp. 1059- 1082, 2003.

[7] B. Vinay Kumar, S. Jayaprada, Dr. S. Vasavi, Dr. P. Bala Krishna , A Study on Constructing Synonymous Gene Database from Biomedical Text Documents, IJCST Vol. 4, Issue 1, Jan - March 2013.

[8] G. Pratyusha, S. Jayaprada, Dr. S. Vasavi , A Study On Pair-Wise Local Alignment Of Protein Sequence For Identifying The Structural Similarity, (IJERT), Vol. 2 Issue 3, March – 2013

[9] M.F. Caropreso, S. Matwin, and F. Sebastiani, “Statistical Phrases in Automated Text Categorization,” Technical Report IEI-B4-07- 2000, Instituto di Elaborazione dell’Informazione, 2000.

[10] C. Cortes and V. Vapnik, “Support-Vector Networks,” Machine Learning, vol. 20, no. 3, pp. 273-297, 1995.

[11] S.T. Dumais, “Improving the Retrieval of Information from External Sources,” Behavior Research Methods, Instruments, and Computers, vol. 23, no. 2, pp. 229-236, 1991.

[12] J. Han and K.C.-C. Chang, “Data Mining for Web Intelligence,” Computer, vol. 35, no. 11, pp. 64-70, Nov. 2002.

[13] A Study on Visualizing Semantically Similar Frequent Patterns in Dynamic Datasets, Y.N.Jyothsna Mallampalli, S.Jayaprada, Dr S.Vasavi, (ijceronline.com) Vol. 3 Issue. 3.