International Journal of Engineering Science Invention
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org ||Volume 4 Issue 7|| July 2015 || PP.01-04

An Efficient Mining for Sensitive Transactional Data

[1] V.Mathimalar, M.Sc., M.Phil, MBA, [2]M.Manimehalai,
[1]*Assistant Professor, P.G Department of computer applications, Shrimati Indira Gandhi College, Trichy-2.*
[2]*Research scholar,Shrimati Indira Gandhi College,Trichy-2.*

ABSTRACT :*The Existing research on privacy-preserving data publishing focuses on relational data: in this context, the objective is to enforce privacy-preserving paradigms, such as k-anonymity and ℓ-diversity, while minimizing the information loss incurred in the anonymizing process (i.e., maximize data utility). Existing techniques work well for fixed-schema data, with low dimensionality. Nevertheless, certain applications require privacy-preserving publishing of transactional data (or basket data), which involve hundreds or even thousands of dimensions, rendering existing methods unusable. To propose two categories of novel anonymization methods for sparse high-dimensional data. The first category is based on approximate nearest-neighbor (NN) search in high-dimensional spaces, which is efficiently performed through locality-sensitive hashing (LSH). In the second category, To propose two data transformations that capture the correlation in the underlying data: 1) reduction to a band matrix and 2) Gray encoding-based sorting. These representations facilitate the formation of anonymized groups with low information loss, through an efficient linear-time heuristic. To show experimentally, using real-life data sets, that all our methods clearly outperform existing state of the art. Among the proposed techniques, NN-search yields superior data utility compared to the band matrix transformation, but incurs higher computational overhead. The data transformation based on Gray code sorting performs best in terms of both data utility and execution time.*

Key words*: Privacy, Anonymity, Transactional Data*

I. INTRODUCTION

ACCURATELY measuring the semantic similarity between words is an important problem in web mining, information retrieval, and natural language processing. Web mining applications such as, community extraction, relation detection, and entity disambiguation, require the ability to accurately measure the semantic similarity between concepts or entities. In information retrieval, one of the main problems is to retrieve a set of documents that is semantically related to a given user query. Efficient estimation of semantic similarity between words is critical for various natural language processing tasks such as word sense disambiguation (WSD), textual entailment, and automatic text summarization.

Semantically related words of a particular word are listed in manually created general-purpose lexical ontologies such as WordNet.1 In WordNet, a synset contains a set of synonymous words for a particular sense of a word. However, semantic similarity between entities changes over time and across domains. For example, apple is frequently associated with computers on the web. However, this sense of apple is not listed in most general-purpose thesauri or dictionaries. A user who searches for apple on the web, might be interested in this sense of apple and not apple as a fruit. New words are constantly being created as well as new senses are assigned to existing words. Manually maintaining ontologies to capture these new words and senses is costly if not impossible.

To propose an automatic method to estimate the semantic similarity between words or entities using web search engines. Because of the vastly numerous documents and the high growth rate of the web, it is time consuming to analyze each document separately. Web search engines provide an efficient interface to this vast information. Page counts and snippets are two useful information sources provided by most web search engines. 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 co occurrence of words P and Q.

For example, the page count of the query “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 . For those reasons, page counts alone are unreliable when measuring semantic similarity.

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, personal name disambiguation, and community mining. Processing snippets is also efficient because it obviates the trouble of downloading web pages, 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 to pranking 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 need to measure semantic similarity between a given pair of words is contained in the top-ranking snippets

II. PROBLEM ANALYSIS

To propose a method that considers both page counts and lexical syntactic patterns extracted from snippets that show experimentally to overcome the above mentioned problems. For example, let us consider the snippet shown in Fig. 1 retrieved from Google for the query Jaguar AND cat. Here, the phrase is the largest indicates a hypernymic relationship between Jaguar and cat. Phrases such as also known as, is a, part of, is an example of all indicate various semantic relations. Such indicative phrases have been applied to numerous tasks with good results, such as hypernym extraction and fact extraction. From the previous example, form the pattern X is the largest Y, where to replace the two words Jaguar and cat by two variables X and Y

III. METHODOLOGY

3.1 EXISTING METHODOLOGY

Given a taxonomy of words, a straightforward method to calculate similarity between two words is to find the length of the shortest path connecting the two words in the taxonomy . If a word is polysemous, then multiple paths might exist between the two words. In such cases, only the shortest path between any two senses of the words is considered for calculating similarity. A problem that is frequently acknowledged with this approach is that it relies on the notion that all links in the taxonomy represent a uniform distance. Resnik proposed a similarity measure using information content. He defined the similarity between two concepts C1 and C2 in the taxonomy as the maximum of the information content of all concepts C that subsume both C1 and C2. Then, the similarity 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. 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 shortest path length, depth, and local density in a taxonomy. Their experiments reported a high Pearson correlation coefficient of 0.8914 on the Miller and Charles benchmark data set. They did not evaluate their method in terms of similarities among named entities. Lin defined the similarity between two concepts as the information that is in common to both concepts and the information contained in each individual concept. Semantic similarity measures have been used in various applications in natural language processing such as word sense disambiguation , language modeling , synonym extraction , and automatic thesauri extraction . Semantic similarity measures are important in many web related tasks. In query expansion , a user query is modified using synonymous words to improve the relevancy of the search. One method to find appropriate words to include in a query is to compare the previous user queries using semantic similarity measures. If there exists a previous query that is semantically related to the current query, then it can be either suggested to the user, or internally used by the
search engine to modify the original query.

3.2 PROPOSED METHODOLOGY

To present an automatically extracted lexical syntactic patterns-based approach to compute the semantic similarity between words or entities using text snippets retrieved from a web search engine. To propose a lexical pattern extraction algorithm that considers word subsequences in text snippets. Moreover, the extracted set of patterns are clustered to identify the different patterns that describe the same semantic relation.

To integrate different web-based similarity measures using a machine learning approach. To extract synonymous word pairs from WordNet synsets as positive training instances and automatically generate negative training instances. Then train a two-class support vector machine (SVM) to classify synonymous and nonsynonymous word pairs. The integrated measure outperforms all existing web based semantic similarity measures on a benchmark data set. To apply the proposed semantic similarity measure to identify relations between entities, in particular people, in a community extraction task. In this experiment, the proposed method outperforms the baselines with statistically significant precision and recall values. The results of the community mining task show the ability of the proposed method to
measure the semantic similarity between not only words, but also between named entities, for which manually created lexical ontologies do not exist or incomplete.

Semantic similarity is a concept whereby a set of documents or terms within term lists are assigned a metric based on the likeness of their meaning.

Snippet is a brief window of text extracted by a search engine around the query term in a document.

Page count is define as the number of pages including the blanks.

Keyword And Text Snippet Insertion

- Snippet is a programming term for a small region of reusable source code.
- It is used to clarify the meaning of an cluttered function or to minimize the use of repeated code.
- Snippet is computationally efficient because it obviates the need to download the documents from the web.

Seacrch Engine Simulation

The user of the search engine can query an keyword in the search text box and the appropriate web pages related to the keyword searched are displayed . the following details are displayed in the webpage for each match

Link - the web page url

Link text – the text associated with the link

Text snippet = an small snippet that is associated with the web page , which describes the content of the web page .

Web page extraction

Any click on the web url will redirect the user to the link . the display of the web search result is based on previously calculated page count and semantic similarity between words

The following are the web pages related to the algorithm applied

Page count calculation

An page count is the click the url has got from the user . the clicks are persistent and are stored for future calculations

- Pagecount is the number of pages which including the blanks.It will increased every time when the user click the link.
- Page counts for the query P AND Q can be considered as an approximation of co-occurrence of two words.
- Page count analysis ignores the position of a word in a page.
- For example, the page count of the query “apple” AND “computer” in Google is 288,000,000, whereas the same for “banana” AND “computer” is only 3,590,000

Sematic similarity

- Pattern matching is the concept which reveals with the similarity between words
- Semantic similarity is measured using the match maker algorithm.
- In this module semantic measure is calculated. For eg when user click car the semantic measure of car become hundred and semantic measure of automobile also increased.

IV. CONCLUSION AND FUTURE ENHANCEMENT

To 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. To 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 non synonymous word pairs selected from Word Net synsets. Experimental 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. Moreover, the proposed method improved the F-score in a community mining task, thereby underlining its usefulness in real-world tasks, that include named entities not adequately covered by manually created resources.

REFERENCES

[1] S. Skiena, *Implementing Discrete Mathematics: Combinatorics and Graph Theory with Mathematica*. Addison-Wesley, 1990.
[2] B.-C. Chen, K. LeFevre, and R. Ramakrishnan, "Privacy skyline: privacy with multidimensional adversarial knowledge," in *Proc. of VLDB*, 2007, pp. 770–781.
[3] R. Agrawal and R. Srikant, "Privacy Preserving Data Mining," in *Proc. of ACM SIGMOD*, 2000, pp. 439–450.
[4] Z. Huang, W. Du, and B. Chen, "Deriving Private Information from Randomized Data," in *Proc. of ACM SIGMOD*, 2005, pp. 37–48.
[5] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan, "Workload-aware Anonymization," in *Proc. of KDD*, 2006, pp. 277–286.
[6] P. Samarati, "Protecting Respondents' Identities in Microdata Release." *IEEE TKDE*, vol. 13, no. 6, pp. 1010–1027, 2001.
[7] G. Aggarwal, T. Feder, K. Kenthapadi, S. Khuller, R. Panigrahy, D. Thomas, and A. Zhu, "Achieving Anonymity via Clustering," in *Proc. of ACM PODS*, 2006, pp. 153–162.
[8] G. Ghinita, P. Karras, P. Kalnis, and N. Mamoulis, "Fast Data Anonymization with Low Information Loss," in *Proc. of VLDB*, 2007, pp. 758–769.
[9] Q. Zhang, N. Koudas, D. Srivastava, and T. Yu, "Aggregate Query Answering on Anonymized Tables," in *Proc. of ICDE*, 2007, pp. 116–125.
[10] M. Atzori, F. Bonchi, F. Giannotti, and D. Pedreschi, "Anonymity Preserving Pattern Discovery," *VLDB Journal*, pp. 703–727, 2008.
[11] V. Verykios, A. Elmagarmid, E. Bertino, Y. Saygin, and E. Dasseni, "Association Rule Hiding," *IEEE TKDE*, vol. 16, no. 4, pp. 434–447, 2004.
[12] C. C. Aggarwal and P. S. Yu, "On Privacy-Preservation of Text and Sparse Binary Data with Sketches," in *SIAM Conference on Data Mining*, 2007.
[13] M. Terrovitis, N. Mamoulis, and P. Kalnis, "Privacy-preserving Anonymization of Set-valued Data," in *Proc. of VLDB*, 2008.
[14] Y. Xu, K. Wang, A. W.-C. Fu, and P. S. Yu, "Anonymizing Transaction Databases for Publication," in *Proc. of SIGKDD*, 2008, pp. 767–775.
[15] D. Richards, "Data Compression and Gray-code Sorting," *Information Processing Letters*, vol. 22, pp. 201–205, 1986.