



## AN ENHANCED AND RELIABLE FRAUD DETECTION BY USING TRIANGLE THEORY

<sup>1</sup> M. Tech., Dept of CSE, Krishna University. AP, India, [vijayakumari28@gmail.com](mailto:vijayakumari28@gmail.com)

<sup>2</sup> M. Tech., Dept of CSE, Krishna University. AP, India, [paaru2945@gmail.com](mailto:paaru2945@gmail.com)

**Abstract:-** Financial fraud is commonly represented by the use of illegal practices where they can intervene from senior managers until payroll employees, becoming a crime punishable by law. There are many techniques developed to analyze, detect and prevent this behavior, being the most important the fraud triangle theory associated with the classic financial audit model. In order to perform this research, a survey of the related works in the existing literature was carried out, with the purpose of establishing our own framework. In this context, this paper presents Fraud Find, a conceptual framework that allows identifying and outlining a group of people inside a banking organization who commit fraud, supported by the fraud triangle theory. Fraud Find works in the approach of continuous audit that will be in charge of collecting information for agents installed in user's equipment. It is based on semantic techniques applied through the collection of phrases typed by the users under study for later being transferred to a repository for later analysis. This proposal encourages contributing with the field of cyber security, in the reduction of cases of financial fraud

*Keywords-Triangletheory,fraudfind,Knn-Algorithm,Human-behaviour.*

### I. INTRODUCTION:-

Users often have difficulties in expressing their web search needs; they may not know the keywords that can retrieve the information they require [1]. Keyword suggestion (also known as query suggestion), which has become one of the most fundamental features of commercial Web search engines, helps in this direction. After submitting keyword query, the user may not be satisfied with the results, so the keyword suggestion module of the search engine recommends a set of  $m$  keyword queries that are most likely to refine the user's search in the right direction. Effective keyword suggestion methods are based on click information from query logs [2], [3], [4], [5], [6], [7], [8] and query session data [9], [10], [11], or query topic models [12]. New keyword suggestions can be determined according to their semantic relevance to the original keyword query. The semantic relevance between two keyword queries can be determined (i) based on the overlap of their clicked URLs in a query log [2], [3], [4], (ii) by their proximity in a bipartite graph that connects keyword queries and their clicked URLs in the query log [5], [6], [7], [8], (iii) according to their

cooccurrences in query sessions [13], and (iv) based on their similarity in the topic distribution space [12]. However, none of the existing methods provide location aware keyword query suggestion, such that the suggested keyword queries can retrieve documents not only related to the user information needs but also located near the user location. This requirement emerges due to the popularity of spatial keyword search [14], [15], [16], [17], [18] that takes a user location and user-supplied keyword query as arguments and returns objects that are spatially close and textually relevant to these arguments. Google processed a daily average of 4.7 billion queries in 2011, a substantial fraction of which have local intent and target spatial web objects (i.e., points of interest with a web presence having locations as well as text descriptions) or geo-documents (i.e., documents associated with geo-locations). Furthermore, 53% of Bing's mobile searches in 2011 were found to have a local intent. To fill this gap, we propose a Location-aware Keyword query Suggestion (LKS) framework. We illustrate the benefit of LKS using a toy example. Consider five geo-documents  $d_1$ – $d_5$  as listed in Figure 1(a). Each document  $d_i$  is associated

with a location  $di$  as shown in Figure 1(b). Assume that a user issues a keyword query  $kq = \text{"seafood"}$  at location, shown in Figure 1(b). Note that the relevant documents  $d1-d3$  (containing  $\text{"seafood"}$ ) are far from  $q$ . A location aware suggestion is  $\text{"lobster"}$ , which can retrieve nearby documents  $d4$  and  $d5$  that are also relevant to the user's original search intention. Previous keyword query suggestion models (e.g., [6]) ignore the user location and would  $\text{"sh"}$ , which again fails to retrieve nearby relevant documents. Note that LKS has a different goal and therefore differs from other location-aware recommendation methods (e.g., auto-completion/instant search [19], [20], tag recommendation [21]). The first challenge of our LKS framework is how to effectively measure keyword query similarity while capturing the spatial distance factor. In accordance to previous query suggestion approaches [3], [4], [5], [6], [7], [8], [10], [11], LKS constructs and uses a keyword-document bipartite graph (KD-graph for short), which connects the keyword queries with their relevant documents as shown in Figure 1(c). Different to all previous approaches which ignore locations, LKS adjusts the weights on edges in the KD-graph to capture not only the semantic relevance between keyword queries, but also the spatial distance between the document locations and the query issuer's location  $q$ . We apply a random walk with restart (RWR) process [22] on the KD-graph, starting from the user supplied query  $kq$ , to find the set of  $m$  key-word queries with the highest semantic relevance to  $kq$  and spatial proximity to the user location. RWR on a KD-graph has been considered superior to alternative approaches [7] and has been a standard technique employed in various (location-independent) keyword suggestion studies [5], [6], [7], [8], [10], [11]. The second challenge is to compute the suggestions efficiently on a large dynamic graph. Performing keyword suggestion instantly is important for the applicability of LKS in practice. However, RWR search has a high computational cost on large graphs. Previous work on scaling up RWR search require pre-computation and/or graph segmentation [22], [23], [24], [25], [26]; part of the required Rescores are materialized under the assumption that the transition probabilities between nodes (i.e., the edge

weights) are known beforehand. In addition, RWR search algorithms that do not rely on pre-computation (e.g., [27]) accelerate the computation by pruning nodes based on their lower or upper bound scores and also require the full transition probabilities. However, the edge weights of our KD-graph are unknown in advance, hindering the application of all these approaches. To the best of our knowledge, no existing technique can accelerate RWR when edge weights are unknown a priori (or they are dynamic). To address this issue, we present a novel partition-based algorithm (PA) that greatly reduces the cost of RWR search on such dynamic bipartite graph. In a nutshell, our proposal divides the keyword queries and the documents into partitions and adopts a lazy mechanism that accelerates RWR search. PA and the lazy mechanism are generic techniques for RWR search, orthogonal to LKS, therefore they can be applied to speed up RWR search in other large graphs. In summary, the contributions of this paper are: We design a Location-aware Keyword query Suggestion (LKS) framework, which provides suggestions that are relevant to the user's information needs and can retrieve relevant documents close to the query issuer's location. We extend the state-of-the-art Bookmark Colouring Algorithm (BCA) [28] for RWR search to compute the location-aware suggestions.

## II. Related Work:-

This paper builds on and significantly extends our preliminary work on the evaluation of competitiveness [30]. To the best of our knowledge, our work is the first to address the evaluation of competitiveness via the analysis of large unstructured datasets, without the need for direct comparative evidence. Nonetheless, our work has ties to previous work from various domains. **Managerial Competitor Identification:** The management literature is rich with works that focus on how managers can *manually* identify competitors. Some of these works model competitor identification as a mental categorization process in which managers develop mental representations of competitors and use them to classify candidate firms [3], [6], [31]. Other manual categorization methods are based on market- and resource-based similarities between a firm and candidate

competitors [1], [5], [7]. Finally, managerial competitor identification has also been presented as a sensemaking process in which competitors are identified based on their potential to threaten an organizations identity [4]. **Competitor Mining Algorithms:** Zheng et al. [32] identify key competitive measures (e.g. market share, share of wallet) and showed how a firm can infer the values of these measures for its competitors by mining (i) its own detailed customer transaction data and (ii) aggregate data for each competitor. Contrary to our own methodology, this approach is not appropriate for evaluating the competitiveness between any two items or firms in a given market. Instead, the authors assume that the set of competitors is given and, thus, their goal is to compute the value of the chosen measures for each competitor. In addition, the dependency on transactional data is a limitation we do not have. Doan et al. explore user visitation data, such as the geo-coded data from location-based social networks, as a potential resource for competitor mining [33]. While they report promising results, the dependence on visitation data limits the set of domains that can benefit from this approach. Pant and Sheng hypothesize and verify that competing firms are likely to have similar web footprints, a phenomenon that they refer to as *online isomorphism* [34]. Their study considers different types of isomorphism between two firms, such as the overlap between the in-links and out links of their respective websites, as well as the number of times that they appear together online (e.g. in search results or new articles). Similar to our own methodology, their approach is geared toward pair wise competitiveness. However, the need for isomorphism features limits its applicability to firms and make it unsuitable for items and domains where such features are either not available or extremely sparse, as is typically the case with co-occurrence data. In fact, the sparsity of co-occurrence data is a serious limitation of a significant body of work [8], [10], [11], [35] that focuses on mining competitors based on comparative expressions found in web results and other textual corpora. The intuition is that the frequency of expressions like “Item A is better than Item B”

“or item A Vs. Item B” is indicative of their competitiveness.

### III. Literature Survey:-

#### 1) A technique for computer detection and correction of spelling errors **AUTHORS:** F. J. Damerou

The method described assumes that a word which cannot be found in a dictionary has at most one error, which might be a wrong, missing or extra letter or a single transposition. The unidentified input word is compared to the dictionary again, testing each time to see if the words match—assuming one of these errors occurred. During a test run on garbled text, correct identifications were made for over 95 percent of these error types.

#### 2) LIBSVM: A library for support vector machines

**AUTHORS:** C.-C. Chang and C.-J. Lin

LIBSVM is a library for Support Vector Machines (SVMs). We have been actively developing this package since the year 2000. The goal is to help users to easily apply SVM to their applications. LIBSVM has gained wide popularity in machine learning and many other areas. In this article, we present all implementation details of LIBSVM. Issues such as solving SVM optimization problems theoretical convergence multiclass classification probability estimates and parameter selection are discussed in detail.

#### 3) Beyond blacklists: Learning to detect malicious Web sites from suspicious URLs

**AUTHORS:** J. Ma, L. K. Saul, S. Savage, and G. M. Voelker

Malicious Web sites are a cornerstone of Internet criminal activities. As a result, there has been broad interest in developing systems to prevent the end user from visiting such sites. In this paper, we describe an approach to this problem based on automated URL classification, using statistical methods to discover the tell-tale lexical and host-based properties of malicious Web site URLs. These methods are able to learn highly predictive models by extracting and automatically analyzing tens of thousands of features potentially indicative of suspicious URLs. The resulting classifiers obtain 95-99% accuracy, detecting large numbers of malicious Web sites from their URLs, with only modest false positives.

#### 4) Design and evaluation of a real-time URL spam filtering service

**AUTHORS:** K. Thomas, C. Grier, J. Ma, V. Paxson, and D. Song

On the heels of the widespread adoption of web services such as social networks and URL shorteners, scams, phishing, and malware have become regular threats. Despite extensive research, email-based spam filtering techniques generally fall short for protecting other web services. To better address this need, we present Monarch, a real-time system that crawls URLs as they are submitted to web services and determines whether the URLs direct to spam. We evaluate the viability of Monarch and the fundamental challenges that arise due to the diversity of web service spam. We show that Monarch can provide accurate, real-time protection, but that the underlying characteristics of spam do not generalize across web services. In particular, we find that spam targeting email qualitatively differs in significant ways from spam campaigns targeting Twitter. We explore the distinctions between email and Twitter spam, including the abuse of public web hosting and redirector services. Finally, we demonstrate Monarch's scalability, showing our system could protect a service such as Twitter--which needs to process 15 million URLs/day--for a bit under \$800/day.

### 5) Detecting spammers on social networks

**AUTHORS:** G. Stringhini, C. Kruegel, and G.

Social networking has become a popular way for users to meet and interact online. Users spend a significant amount of time on popular social network platforms (such as Facebook, MySpace, or Twitter), storing and sharing a wealth of personal information. This information, as well as the possibility of contacting thousands of users, also attracts the interest of cybercriminals. For example, cybercriminals might exploit the implicit trust relationships between users in order to lure victims to malicious websites. As another example, cybercriminals might find personal information valuable for identity theft or to drive targeted spam campaigns. In this paper, we analyze to which extent spam The present work proposes Fraud Find, a conceptual framework to detect financial fraud supported by the fraud triangle factors which, compared to the classic audit analysis, makes a significant contribution to the early detection of

has entered social networks. More precisely, we analyze how spammers who target social networking sites operate. To collect the data about spamming activity, we created a large and diverse set of "honey-profiles" on three large social networking sites, and logged the kind of contacts and messages that they received. We then analyzed the collected data and identified anomalous behavior of users who contacted our profiles. Based on the analysis of this behavior, we developed techniques to detect spammers in social networks, and we aggregated their messages in large spam campaigns. Our results show that it is possible to automatically identify the accounts used by spammers, and our analysis was used for take-down efforts in a real-world social network. More precisely, during this study, we collaborated with Twitter and correctly detected and deleted 15,857 spam profiles.

### Proposed Algorithm:-

#### Algorithm 2 PyramidFinder

---

**Input:** Set of items  $\mathcal{I}$   
**Output:** Dominance Pyramid  $\mathcal{D}_{\mathcal{I}}$

- 1:  $\mathcal{D}_{\mathcal{I}}[0] \leftarrow \text{Sky}(\mathcal{I})$
- 2:  $\mathcal{Z} \leftarrow \mathcal{I} \setminus \text{Skyline}(\mathcal{I})$
- 3:  $level \leftarrow 1$ .
- 4: **while**  $\mathcal{Z}$  is not empty **do**
- 5:    $\mathcal{D}_{\mathcal{I}}[level] \leftarrow \text{Sky}(\mathcal{Z})$
- 6:   **for every** item  $j \in \mathcal{D}_{\mathcal{I}}[level]$  **do**
- 7:     **for every** item  $i \in \mathcal{D}_{\mathcal{I}}[level - 1]$  **do**
- 8:       **if**  $i$  dominates  $j$  **then**
- 9:         Add a link  $i \rightarrow j$
- 10:       **break**
- 11:     **end if**
- 12:   **end for**
- 13: **end for**
- 14:  $\mathcal{Z} \leftarrow \mathcal{Z} \setminus \text{skyline}(\mathcal{Z})$
- 15:  $level \leftarrow level + 1$
- 16: **end while**

---

### IV. Conclusion:-

fraud within an organization. Taking into account human behavior factors, it is possible to detect unusual transactions that would have not been considered using traditional audit methods. These patterns of behavior can be found in the

information that users generate when using the different applications on a workstation. The collected data is examined using data mining techniques to obtain patterns of suspicious behavior evidencing possible fraudulent behavior. Nevertheless, the legal framework and the different regulations that are applied in public and private institutions of a particular region represent a high risk for the non-implementation of this architecture as an alternative solution. Future work will have as its main objective the implementation and evaluation of the framework as a tool for continuous auditing within an organization.

## V. REFERENCES

- [1] "ACFE Asociaci' on de Examinadores de FraudesCertificados," (Date last accessed 15-July-2014). [Online]. Available: <http://www.acfe.com/uploadedfiles/acfewebsite/content/documents/rttm-2010.pdf>
- [2] "PwC," (Date last accessed 15-July-2014). [Online]. Available: <https://www.pwc.com/gx/en/economic-crime-survey/pdf/GlobalEconomicCrimeSurvey2016.pdf>
- [3] N. B. Omar and H. F. M. Din, "Fraud diamond risk indicator: An assessment of its importance and usage," in 2010 International Conference on Science and Social Research (CSSR 2010). IEEE, dec 2010.
- [4] "Lynx," (Date last accessed 15-July-2014). [Online]. Available: <http://www.iic.uam.es/soluciones/banca/lynx/>
- [5] "Ibm," (Date last accessed 15-July-2014). [Online]. Available: <https://www.ibm.com/developerworks/ssa/local/analytics/prevencion-de-fraude/index.html>
- [6] C. Holton, "Identifying disgruntled employee systems fraud risk through text mining: A simple solution for a multi-billion dollar problem," *Decision Support Systems*, vol. 46, no. 4, pp. 853–864, mar 2009.
- [7] S. Hoyer, H. Zakhariya, T. Sandner, and M. H. Breitner, "Fraud prediction and the human factor: An approach to include human behavior in an automated fraud audit," in 2012 45th Hawaii International Conference on System Sciences. IEEE, jan 2012.
- [8] M. Kandias, A. Mylonas, N. Virvilis, M. Theoharidou, and D. Gritzalis, "An insider threat prediction model," in *Trust, Privacy and Security in Digital Business*. Springer Berlin Heidelberg, 2010, pp. 26–37.
- [9] M. Jans, N. Lybaert, and K. Vanhoof, "Internal fraud risk reduction: Results of a data mining case study," *International Journal of Accounting Information Systems*, vol. 11, no. 1, pp. 17–41, mar 2010.
- [10] P. K. Panigrahi, "A framework for discovering internal financial fraud using analytics," in 2011 International Conference on Communication Systems and Network Technologies, June 2011, pp. 323–327.
- [11] D. Yue, X. Wu, Y. Wang, Y. Li, and C. H.

Chu, "A review of data mining-based financial fraud detection research," in 2007 International Conference on Wireless Communications, Networking and Mobile Computing, Sept 2007, pp. 5519– 5522.

[12] M. Ahmed and A. N. Mahmood, "A novel approach for outlier detection and clustering improvement," in 2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA), June 2013, pp. 577–582. [13] A. Vikram, S. Chennuru, H. R. Rao, and S. Upadhyaya, "A solution architecture for financial institutions to handle illegal activities: a neural networks approach," in 37th Annual Hawaii International Conference on System Sciences, 2004. Proceedings of the, Jan 2004, pp. 181–190.

[14] H. Li and M. L. Wong, "Financial fraud detection by using grammar-based multi-objective genetic programming with ensemble learning," in 2015 IEEE Congress on Evolutionary Computation (CEC), May 2015, pp. 1113–1120.

[15] D. Cappelli, A. Moore, R. Trzeciak, and T. J. Shimeall, "Common sense guide to prevention and detection of insider threats 3rd edition–version 3.1," Published by CERT, Software Engineering Institute, Carnegie Mellon University, <http://www.cert.org>, 2009.

[16] P. K. Panigrahi, "A framework for discovering internal financial fraud using analytics," in 2011 International Conference on Communication Systems and Network Technologies. IEEE, jun 2011.

[17] G. Mui and J. Mailley, "A tale of two triangles: comparing the fraud triangle with criminology's crime triangle," Accounting Research Journal, vol. 28, no. 1, pp. 45–58, jul 2015.

[18] D. Al-Jumeily, A. Hussain, MacDermott, H. Tawfik, G. Seeckts, and J. Lunn, "The development of fraud detection systems for detection of potentially fraudulent applications," in 2015 International Conference on Developments of ESystems Engineering (DeSE), Dec 2015, pp. 7–13.

[19] S. GVK and S. R. Dasari, "Big spectrum data analysis in dsa enabled lte-a networks: A system architecture," in 2016 IEEE 6th International Conference on Advanced Computing (IACC), Feb 2016, pp. 655–660.

[20] T. Prakash, M. Kakkar, and K. Patel, "Geo-identification of web users through logs using elk stack," in 2016 6th International Conference - Cloud System and Big Data Engineering (Confluence), Jan 2016, pp. 606–610.

[21] U. Thacker, M. Pandey, and S. S. Rautaray, "Performance of elasticsearch in cloud environment with ngram and nonngram indexing," in 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), March 2016, pp. 3624–3628.

[22] D. N. Doan and G. Iuhasz, "Tuning logstash garbage collection for high throughput in a monitoring platform," in Symbolic and Numeric

Algorithms for Scientific Computing (SYNASC), 2016 18th International Symposium on. IEEE, 2016, pp. 359–365.

[23] V. M. Ionescu, “The analysis of the performance of rabbitmq and activemq,” in 2015 14th RoEduNet International Conference - Networking in Education and Research (RoEduNet NER), Sept 2015, pp. 132–137.

[24] D. N. Doan and G. Iuhasz, “Tuning logstash garbage collection for high throughput in a monitoring platform,” in 2016 18th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), Sept 2016, pp. 359–365.

[25] X. M. Li and Y. Y. Wang, “Design and implementation of an indexing method based on fields for elasticsearch,” in 2015 Fifth

International Conference on Instrumentation and Measurement, Computer, Communication and Control (IMCCC), Sept 2015, pp. 626–630.

#### **AUTHORS:-**



**Dr.R.Vijayakumari** received Ph.D Degree in Computer Science and Engineering from Acharya Nagarjuna University, Guntur. She is working presently as Assistant Professor in Department of Computer Science, Krishna University, Machilipatnam. She has published research papers in various National and International journals. Her Research interests are Graphical Password Authentication, Cloud Computing, Parallel Computing, and Big Data Analytics



**Gorla siva parvathi** is a student of krishna University, Machilipatnam Present she is Studying M.tech[computer science & Engineering ] and she received B.tech Degree from JNTUK University