Detecting data semantic: A data leakage prevention approach

Sultan Alneyadi, Elankayer Sithirasenan, Vallipuram Muthukkumarasamy
School of Information and Communication Technology, Griffith University
Gold Coast, Australia
sultan.alneyadi2@griffithuni.edu.au, {e.sithirasenan, v.muthu}@griffith.edu.au

***Abstract*—data leakage prevention systems (DLPSs) are increasingly being implemented by organizations. Unlike standard security mechanisms such as firewalls and intrusion detection systems, DLPSs are designated systems used to protect in use, at rest and in transit data. DLPSs analytically use the content and surrounding context of confidential data to detect and prevent unauthorized access to confidential data. DLPSs that use content analysis techniques are largely dependent upon data fingerprinting, regular expressions, and statistical analysis to detect data leaks. Given that data is susceptible to change, data fingerprinting and regular expressions suffer from shortcomings in detecting the semantics of evolved confidential data. However, statistical analysis can manage any data that appears fuzzy in nature or has other variations. Thus, DLPSs with statistical analysis capabilities can approximate the presence of data semantics.**

In this paper, a statistical data leakage prevention (DLP) model is presented to classify data on the basis of semantics. This study contributes to the data leakage prevention field by using data statistical analysis to detect evolved confidential data. The approach was based on using the well-known information retrieval function Term Frequency-Inverse Document Frequency (TF-IDF) to classify documents under certain topics. A Singular Value Decomposition (SVD) matrix was also used to visualize the classification results. The results showed that the proposed statistical DLP approach could correctly classify documents even in cases of extreme modification. It also had a high level of precision and recall scores.

Keywords—Data leakage prevention; Data semantics; Statistical analysis; Singular Value Decomposition

I. INTRODUCTION

The leaking of confidential data to unauthorized entities can result in various problems for organizations and individuals. Similar to trade secrets, health records and banking details, leaks can affect the competitiveness of companies, the privacy of patients, and the security of accounts. According to recent reports, such leaks are incremental in relation to their size and impact [1]; for example, one hacker leaked the account details of over 77 million PlayStation network subscribers, resulting in a total shutdown of PlayStation network services for several weeks and a public apology from the CEO of Sony [2]. Similarly, the giant online shopping website eBay suffered from one of the biggest recorded leaks in history when the names, emails addresses, and personal information of more than 145 million customers were stolen. This disrupted the website's operations and forced customers to initiate a major account password reset process [3]. Further, in 2010 hundreds of thousands of top secret United States diplomatic cables and military reports were released to the public by an insider and posted on Wikileaks.org. Significant conflicts arose between the United States (as the affected party) and media freedom advocates in relation to the leaks [4]. Additionally, websites such as Pastebin and 4chan allowed confidential data, including stolen credit cards details and personal photos to be anonymously spread once leaked [5] [6]. Such data leakage incidents can negatively affect governments, organizations, and individuals.

To mitigate these problems, academics and practitioners have developed methods and techniques dedicated to the protection of confidential data. These methods and techniques are increasingly and independently being described as Data Leakage Prevention Systems (DLPSs). Unlike standard security mechanisms such as Firewalls and Intrusion Detections/Prevention Systems, DLPSs provide continuous monitoring of confidential data everywhere. Whether the data is "*in use"* (i.e., being accessed and processed by users), *"in transit"* (i.e., travelling internally or externally between network nodes) or *"at rest"* (i.e., being stored in data repositories), DLPSs are constantly tracking confidential data. According to [7] a DLPS is: "*a system that is designed to detect and prevent the unauthorized access, use, or transmission of confidential information*." Also, in [8] DLPSs are described as: "*Products that, based on central policies, identify, monitor, and protect data at rest, in motion, and in use, through deep content analysis"*. Depending on the state of the data, DLPSs perform different tasks; for example, a DLP agent can disable the use of USB ports, screen shots and printing documents for "*in use"* data, act like a proxy and inspect all traffic for any potential leaks for "*in transit*" data, and monitor and audit all types of action done to confidential data for "*at rest*" data. Fig. 1 shows a typical DLP deployment within a network.

In any such detection task, DLPSs must perform two main analyzes; that is, the content and context analyzes of the confidential data. Contextual analysis includes studying attributes such as the size, time, format, sender, and recipient of data. Content analysis includes using regular expressions of data fingerprinting and the statistical analysis of content. In these protection tasks, DLPSs use remedial actions such as audits, alerts, blocks, quarantines, and encryptions [9].

978-1-4673-7952-6/15 $31.00 © 2015 IEEE
DOI 10.1109/Trustcom.2015.464

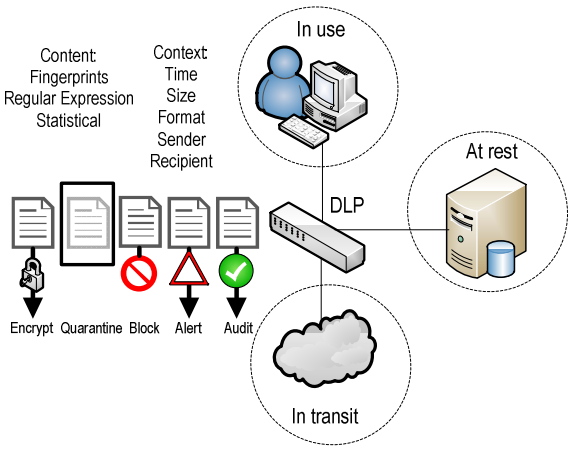

Figure. 1. Data Leakage Prevention deployment within a network showing data states and remedial actions.

The use of content analysis has been widely implemented in many DLPSs that focus on detecting confidential data. Detection occurs when copies or parts of the confidential data are used, accessed, or transmitted without authorization. Regular expression (e.g., dictionary-based), fingerprinting (including full and partial file detection), and statistical analysis are used to identify copies of confidential data. The problem with regular expression is that it has limited coverage and thus is only suitable for the rule-based detection of items such as credit cards and social security numbers. The susceptibility to change of fingerprinting also creates issues, as tiny changes in the original data could result in a totally different hash values. This may occur when standard hashing techniques such as SHA1 and MD5 are used[10].

Some robustness in detecting modified data has been found in more advanced fingerprinting (see [11], [12]). Techniques such as Locality Sensitive Hashing (LSH) and Similarity Digest have been used to detect modified versions of confidential data; however, these techniques are ineffective against extreme data obfuscation. Conversely, statistical analysis can be used to approximate the amount of confidential data within captured data traffic. Statistical techniques like N-gram analysis and term weighting can help to identify data semantics even where extreme data modification has occurred [13], [14].

This paper examines the effectiveness of using statistical analysis in detecting confidential data semantics. A DLP model is proposed based on term weighting function Term Frequency-Inverse Document Frequency (TF-IDF) classification and powered by Singular Value Decomposition (SVD) abstraction representation. The proposed method is shown to be robust against various levels of data obfuscation, including multi-level document spinning.

The paper is divided as follows: Section II discusses related research; Section III describes the proposed DLP approach; Section IV details all the experiments conducted; Section V analyzes the findings of the research; and Section VI draws conclusions and proposes future directions.

II. Related Work

Within the literature there are various approaches directed at detecting data semantics using advanced types of fingerprinting and statistical analysis; for example, advanced fingerprinting [15]. In this paper modified full data fingerprinting is presented to overcome shortcomings in fingerprints produced by ordinary data hashing. Ordinary fingerprints are vulnerable and can be bypassed even with minor modifications to the original data; thus, *k*-skip-*n*-grams are used to produce modified fingerprints. The *k*-skip-*n*-grams introduce a robust method for identifying the original data even after data modification (i.e., addition, subtraction, word synonyms). Both confidential and non-confidential documents are processed to produce fingerprints in this method, where non-confidential documents produce non-confidential *k*-skip-*n*-grams. The non-confidential *k*-skip-*n*-grams help to eliminate unnecessary n-grams in the confidential documents. This proposed method outperformed ordinary full fingerprinting methods in almost all the experiment scenarios. However, intensive indexing is required for all confidential and non-confidential documents. Thus, the extra storage and processing capabilities required are a major drawback for this method.

Hart et al. [16] introduced a method based on machine learning to classify enterprise documents as confidential and non-confidential. This approach uses a Support Vector Machines (SVMs) algorithm to classify three types of data: enterprise private, enterprise public, and non-enterprise. The data were represented by the most frequent binary weighted unigrams found across all corpora. The method was able to identify 97 percent of data leaks with a false negative (FN) rate of 3.0 percent. Unfortunately, this method can only classify data as public or private and ignores more flexible classification levels such as top secret, secret, and confidential. Thus, this method could obstruct the work process making the enforcement of security polices difficult.

Another form of LSH called TrendMicro Locality Sensitive Hashing (TLSH) was introduced in [10]. To improve data detection accuracy, this method uses quartiles to track the counting bucket heights and has a sliding window of five bytes. At a certain window place, the five bytes form a trigram that is finally mapped into a counting bucket using Pearson hash [17]. TLSH was tested against inserting, deleting, swapping, and replacing words. It outperformed previous methods such as Sdhash and Ssdeep on detecting text mutation. Further, a semantic hashing approach to achieve higher precision and recall than TF-IDF or LSH was presented in [18]. The method is based on using binary codes as memory addresses to find semantically similar documents and was able to retrieve similar

documents in a small fraction of the time taken by LSH. The only notable limitation of this method is that it was only able to retrieve documents based on very general topics such as government borrowing and disaster and accidents. Thus, trying to retrieve specific sub-topics within a general topic could be challenging.

III. Data Leakage Prevention Model

This study's DLP model uses statistical data analysis to detect confidential data semantics. Specifically, it uses the well-known term weighting function TF-IDF to measure the amount of information (i.e., informativeness) in a tested document. It aims to cluster topic-relative documents under a predefined category. Assuming that each category has a secrecy level, documents with restricted secrecy levels can be detected and remedial actions such as blocks, alerts, and quarantines can be undertaken. Further, using statistical analysis to perform the classification task can help the DLP in approximating the semantics of confidential data; for example, DLPSs using regular expressions and data fingerprinting can easily identify document (*D*) if a user (*U*) is trying to maliciously access, use or send it. However, if (*D*) is altered by (*U*) by adding, subtracting, or rephrasing words, lines, or paragraphs, identifying (*D*) will be challenging even with the robust data fingerprinting discussed in section II. Thus, by approximating the presence of (*D*) using a statistical analysis, a DLP can detect existing confidential data. Moreover, newly created documents that might contain confidential data can be identified if they are classified under a restricted category. Fig. 2 shows and an overview of the proposed DLP model.

Specifically, Fig. 2 shows a user attempting to send a document (*D*) over a networked environment. At this stage, (*D*) has no secrecy level; it is either new or has been modified by the user. Before allowing (*D*) to leave the secure premises, a DLPS intercepts and inspects all traffic.

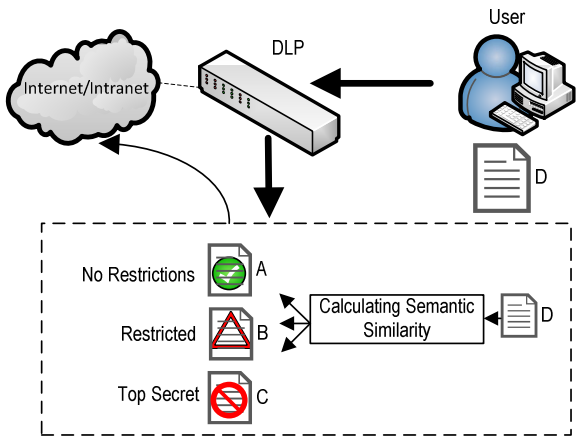

Figure 2. Proposed Data Leakage Prevention model showing document comparison to topics.

It should be noted that any DLPS is either a hardware appliance or a DLP software agent installed on a user's machine. Next, the similarity between (*D*) and other predefined categories is calculated by statistically analyzing the overall term weighting of (*D*) and then comparing it with the overall term weighting of the categories. Upon the selection of one category (topic), (*D*) will automatically inherit the secrecy level and the selected remedial action associated with that category; for example, if (*D*) is classified as topic (*A*) (i.e., no restriction) then no actions will be taken. Conversely, if (*D*) is classified as (*B*) or (*C*) (i.e. restricted or top secret) notification or blocking actions can be applied.

A. Term Weighting Classification

Term weighting is a statistical method that indicates the importance of a word within a document. It is normally used in text classification Vector Space Models (VSM) in which documents are treated as vectors. Salton et al. first introduced this approach in 1975 [19] and used a novel approach based on space density computations. It was shown that separation between documents spaces resulted in better retrieval. Thus, a clustered document space was best considered when related documents were grouped into classes (classes are formed around cluster centroids, which are also formed around a main centroid). Fig. 3 shows an example of a clustered document space.

The TF-IDF was efficient and accurate when used for classification purposes. For many applications, TF-IDF is considered the best well-known term weighting function. The TF-IDF function is illustrated blow:

- Term frequency (*tf*):

$$W_{t,d} = \begin{cases} 1 + log_{10} tf_{t,d} & \text{, if } \quad tf_{t,d} > 0 \\ 0 & \text{, otherwise} \end{cases} \qquad (1)$$

Where, *tf* simply means the frequency of a specific term within one document. The use of a logarithmic representation is important to avoid dealing with huge figures. Also, "1" is added to avoid an undefined result in case a term is not present.

- Inverse document frequency (*idf*):

$$idf_t = log_{10}\left({}^{N}/_{df_t}\right) \qquad (2)$$

This is an inverse measure of the informativeness of *t*. Where, *N* is the total number of documents in a corpus and *dft* means the number documents within a corpus that contain a specific term; for example, if there is a corpus of 1,000 documents and if the term "the" appeared in all documents, then $log_{10}\left({}^{1000}/_{1000}\right)$ would result in zero. Thus, the term "the" has no real value. Conversely, if a term were rarely sighted in the corpus then the *idf* value would be high. The total weight for a term is given by the product value of *tf* and *idf* [20].

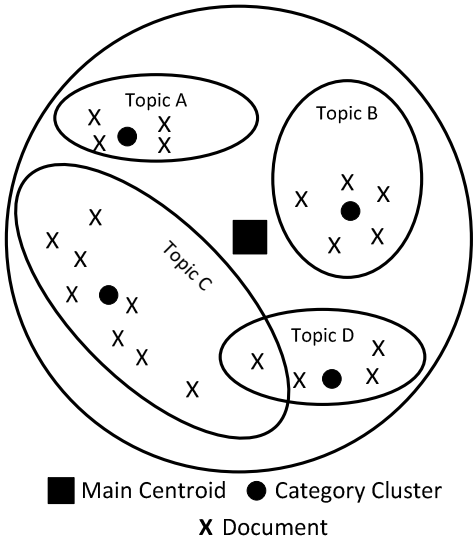

Figure 3. Clustered document space.

In the proposed DLP model, the category centroids form a number of documents reflecting one topic. A centroid source can also be a comprehensive document that contains varieties of relevant terms (e.g., a Wikipedia article). Further, a document is classified under a category by calculating the TF-IDF function for each term within the document and the available category centroids. Then, the following two well-known distance measures are used to measure the similarity: taxicab distance and cosine similarity.

The overall taxicab distance is given as follows:

- $d_1(p,q) = \|p-q\|_1 = \sum_{i=1}^{n}|p_i - q_i|$ (3)

Where, p is the TF-IDF value of the term in the document and q is the TF-IDF value in the category.

The *cosine* similarity is calculated using:

- $cos(\vec{p},\vec{q}) = \vec{p}\cdot\vec{q} = \sum_{i=1}^{|V|} p_i\, q_i$ (4)

Where, p is the normalized TF-IDF value of the term in the document and q is the normalized TF-IDF value in the category. The overall *cosine* similarity value is the sum of all calculated dot products of intersected terms between the document and the categories.

B. Singular Value Decomposition

The SVD is a matrix-based mathematical approach used to reduce dimensionality. The dimension of the greatest variant is used for the linear classification. In this study, SVD breaks down the matrix (*user-defined dimensions* × (*document* + *categories)*) into linearly independent components that form an abstraction separate to the overall semantic weight of the document. Most of the resulting component values are small and can be ignored. The remaining components form the smaller semantic dimension of a topic. Thus, SVD performs some sort of noise reduction. It achieves this by factorizing the main matrix into the three other matrices as follows:

$$A = U\,\Sigma\,V^T$$

Where A is the original (*user-defined dimensions* × (*document* + *categories)*) or ($m \times n$) matrix and:

- U: is an $m \times m$ orthonormal matrix whose columns are the eigenvectors of AA^T
- V: is an $n \times n$ orthonormal matrix whose columns are the eigenvectors of A^TA
- Σ : is an $m \times n$ diagonal matrix, where the diagonal elements are the non-negative square roots of the eigenvalues of A^TA

Multiplying the three matrices can reconstruct the original matrix "A"; however, the main purpose is to use the dimensionally reduced representation to recognize similar terms and documents. The final outcome of using SVD is that relevant components will appear more similar and irrelevant components more dissimilar [21].

IV. Experiments

To test our approach a total of 360 articles were gathered from public sources (e.g., pcmag.com and scmagazine.com) and divided into six categories of 60 topic-related articles. The selected categories were: Antivirus (A), Data Leakage Prevention (D), Intrusion Detection Systems (I), Encryption (E), Firewall (F), and Virtual Private Networks (V). These articles fell under the wider topic of "Information Security." While some available datasets have used academic research (e.g., 20 Newsgroups and Reuters 21578), this study used the self-gathered dataset "Information Security," as its topics were strongly related or overlap, whereas publically available datasets have very distinct categories (e.g., economics and sport). Thus, by testing semantically related topics the robustness of the method was tested. The experiments were divided into two main parts. First, to accurately detect data semantics with or without prior knowledge, the proposed DLP model's ability to classify all documents under relevant topics in three different scenarios (i.e., known documents, partially known documents and unknown document) was tested. Second, to view the document clustering around the category centroid, the SVD function was used to visualize the document classification in the vector space.

A. Scenario 1 (Known Data)

In this test it was assumed that all documents were known to the DLP. Thus, every category centroid was constructed solely from the 60 topic-related documents. A total of 360 documents were tested against six categories. Each tested document was processed as follows. First, the document was stripped of unnecessary wording and suffixes. Stop words removal and word stemming algorithms were used to remove stop words (such as "is", "the," "a," and "it") and suffixes (such as "ing," "tion," and

"s"). As results may vary from one scenario to another, this step is optional. Second, the remaining words were sorted by frequency and the TF-IDF weight was calculated for each term. The documents' length normalization was then considered to avoid bias classification. Third, after calculating the distance between intersected terms, the document was classified under one of the six categories. Taxicab distance and cosine similarities were used to find the category with the highest degree of similarity.

The overall classification results when the documents are known are encouraging (see Table 1). All the documents (i.e., 100 percent) were correctly classified using taxicab distance and only one document from category (I) was misclassified when cosine similarity was used. These results were achieved when step one (removing unnecessary wording) was disabled. Removing stop words negatively affected the overall classification. The best correct classification (i.e., 98.6 percent) was achieved by cosine similarity as compared to the taxicab distance result of 96.6 per cent.

B. Scenario 2 (Partially Known Data)

The same testing procedure as used in Scenario 1 was carried used for this test. The only difference was that half of the test documents were assumed to be unknown to the DLP. Thus, each category centroid was constructed by accumulating 30 topic-related documents. The overall classification results are shown in Table 1. Notably, the overall classification was negatively affected by reducing the size of the category centroid. However, the achieved overall classification indicated that up to 80.8 percent of the documents were correctly classified using cosine similarity. Thus, at least 30 percent of the unknown documents were classified under the correct category. The worst classification results of 65.2 percent were achieved for taxicab distance when all stop words were removed.

C. Scenario 3 (Unknown Data)

The same testing procedures used in Scenarios 1 and 2 were used for Scenario 3; however, in this test, to simulate the classification of new documents, it was assumed that all the documents were unknown. Each category centroid was created form a single topic-related Wikipedia article. The classification results varied between a perfect correct classification (i.e., category (A)) and very low correct classification (i.e., category (D)). It is evident that using a single document to construct the category centroid could negatively affect the overall classification. However, using the semantics of a single document (such as a Wikipedia article) resulted in correct classifications of at least 50 percent of the documents. Thus, at least 180 documents were correctly classified. The best classification result was achieved by cosine similarity when all the stop words were removed. The overall classification results are shown in Table 1.

TABLE I. OVERALL CLASSIFICATION RESULTS FOR EACH SCENARIO

CATEGORY	Similarity Measure	Scenario 1	Scenario 2	Scenario 3
A	Taxicab	60	39	60
	Cosine	60	41	60
D	Taxicab	60	38	2
	Cosine	60	46	3
E	Taxicab	60	60	55
	Cosine	60	58	50
F	Taxicab	60	43	18
	Cosine	60	43	23
I	Taxicab	60	52	9
	Cosine	59	56	21
V	Taxicab	60	35	37
	Cosine	60	47	49
Classification %	**Taxicab**	**100**	**74.00**	**50.00**
	Cosine	**99.72**	**80.83**	**57.22**
Classification % without stop words	**Taxicab**	**96.67**	**65.28**	**53.06**
	Cosine	**98.61**	**74.44**	**62.78**

D. Data Visualization Using SVD

After using the TF-IDF term weighting function to calculate the overall weight for each document, the same calculated weight was used in the SVD matrix for visualization purposes. As mentioned previously, SVD was used to reduce the dimensionality and analyze the semantic position documents' in the vector space. The 360 documents were plotted using the TF-IDF term weighting in relation to one main centroid. The centroid contained the top 1,000 terms in the "Information Security" topic based on the TF-IDF rank. As shown in Fig. 4, most of the documents clustered around the center of the "Information Security" topic centroid.

The 3D visualization of the documents resulted in a star shape with six sides. Each side represented one of the six categories. As mentioned in Section III, SVD can create an abstraction of a multi-dimensional dataset. Fig. 4 shows the overlapping representations of document semantics. For classification purposes, particularly in the classification of sub-topics, this representation lacks semantic discrimination.

A better representation of the 360 documents is shown in Fig. 5. In this visualization, SVD was used to represent the documents in relation to six different category centroids. Each centroid represented one of the topics: Antivirus (A), Data Leakage Prevention (D), Intrusion Detection Systems (I), Encryption (E), Firewall (F), and Virtual Private Networks (V). Each category centroid was formed using the top 1,000 TF-IDF terms in each topic. All the related documents formed a cluster around the center of the

category centroids from which it was possible to visually discriminate between different topics. Due to the dimensionality constrains, Fig. 5 shows only one side of the 3D representation; for example, category (A) (in red) has formed a distinctive cluster with no overlapping representation with other categories. Notably, except for a minor overlapping between categories (D) and (I) at the center, there was no overlap between all six categories.

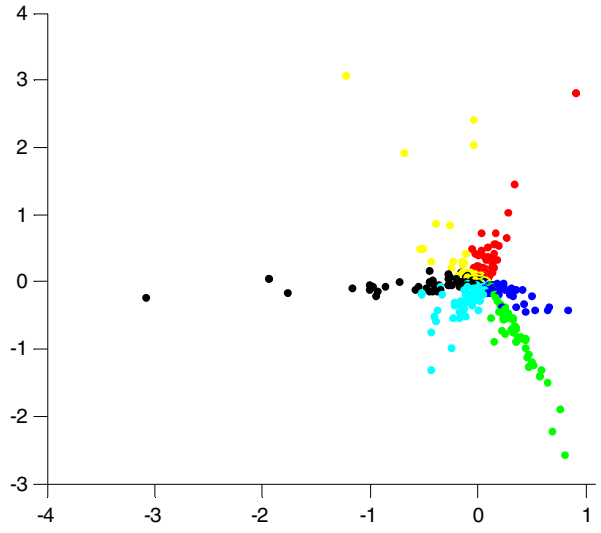

Figure 4. A Singular Value Decomposition representation of documents clustering around the "Information Security" centroid.

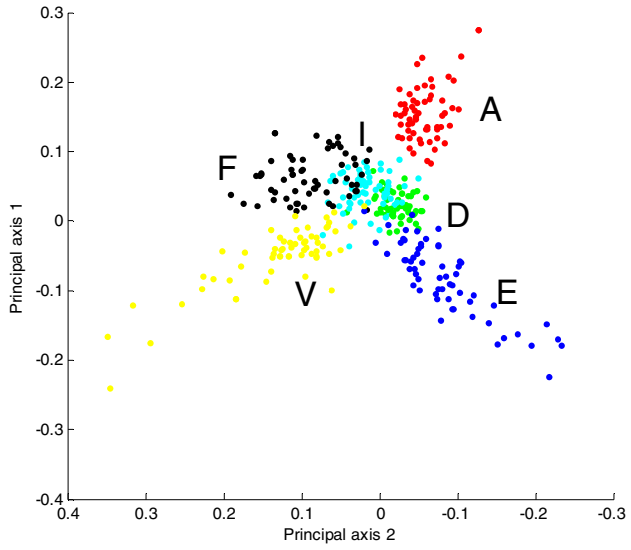

Figure 5. A Singular Value Decomposition representation of the six category clusters in which it is possible to visually distinguish between different topics.

V. Analysis

In this section, findings from some previous experiments are analyzed and the strengths and limitations of the proposed DLP model are identified.

A. Overall Classification

In the main experiment, in which 360 documents were tested against six categories, all documents were classified under their correct categories. In almost all cases, cosine similarity achieved the highest correct classification, especially in Scenarios 2 and 3. Taxicab distance achieved its best results in Scenario 1 where all the documents were known. Thus, for a dynamic envi ronment where new data are continuously created, cosine similarity achieves a better classification and identification of data. Unlike other distance measures such as taxicab, dice coefficient, and Euclidian distances, cosine similarity works better with vector length normalization. A ditionally, removing stop words did not benefit the overa ll classification except in Scenario 3. As the IDF function was used to demote too frequent terms (i.e., stop words), removing stop words led to an improper distribution of the IDF weight.

B. Precision and Recall

Depending on the system that performs the classification, the requirement for precision and recall measures may vary. While both measures are important, some systems require more focus on achieving better precision or better recall; for example, for a DLPS it is more important to recall all possible confidential documents than deal with false positives due to low precision. In this work, precision was defined as the ratio between correctly classified articles and the number of overall detected articles. Recall was defined as the ratio between correctly classified articles and th e number of desired detected articles.

Based on Scenario 1, in which all the documents were known, the precision and recall measures for each category were analyzed. The classification results show an average of 0.99 for both precision and recall when using cosine similarity, an average of 0.97 precision, and 0.96 recall when using taxicab distance (see Table 2). Thus, in the best case scenario, less than four of 360 documents were misclassified; this might be considered a data leakage threat. Additionally, individual cate gories (such as categories (D) and (F)) had perfect precision and recall when using cosine similarity. The worst recall result among all categories when using both distance measures was 0.95; however, even in this instance only three of 60 documents of each category were misclassified.

TABLE II. PRECISION AND RECALL MEASURES FOR EACH CATEGORY

	Taxicab		Cosine	
Category	**Precision**	**Recall**	**Precision**	**Recall**
Antivirus	0.96	0.95	0.97	1
DLP	0.98	0.98	1	1
Encryption	0.89	0.98	0.95	1
Firewall	1	1	1	1
IDS	1	0.95	1	0.95
VPN	1	0.95	1	0.97
Average	0.97	0.96	0.99	0.99

C. Irrelevant Topics

All the previous experiments were based on the general topic "Information Security" and its sub-topics. It was evident that it is possible to semantically classify the sub-topics with high precision and recall scores using distance measures. However, the visualization of this classification using SVD was not distinctive. Indeed, some documents from one category were closer to other categories' centroids than the correct one (see Fig. 5). In this case, using the distance measures could result in misclassification as occurred in categories (D) and (I). This problem was expected, as the two categories were semantically related.

The proposed method's ability to distinguish between general topics was also tested. Thus, on top of the six previous categories we added two more categories "economy" and "sport." Each new category contained 10 articles gathered form CNN.com. Two new category centroids were also constructed as discussed in Scenario 1.

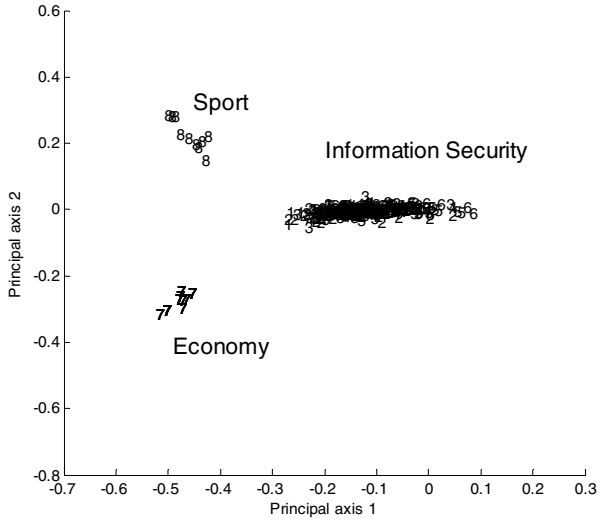

Figure 6. The Singular Value Decomposition visualization of the three general topics.

The experimental results showed a 100 per cent correct classification of new documents under the relevant category. Also, the more general topics appeared distinctive when visualized using the SVD. All the previous categories (i.e., numbers one to six) appeared in the main cluster "Information Security" (see Fig. 6). Additionally, the two new categories "economy" and "sport" (i.e., numbers seven and eight) appeared farther away from each other and form "Information Security."

D. Classifying Modified Documents

Based on Scenario 1, where all the documents were known, a test was run to identify the semantics of the documents after modifications. To achieve this the article spinning tool (BFS Pro) was used [22] to change the wording of documents by changing words to its synonyms. In this test, a situation was simulated where known confidential documents were modified to evade DLP detection. The document spinning process in this test included spinning every possible word, spinning every other word, spinning every third word and spinning every forth word.

Table 3 shows the overall classification results for the two distance measures with/without the stop words. Spinning every possible word resulted in a 59.44 percent correct classification rate using taxicab distance with approximately 40 percent accuracy deterioration. Thus, approximately 212 of 360 documents were correctly classified after extreme modification. Similarly, cosine similarity gave the encouraging result of 63.33 percent after the spinning of every possible word. Thus, approximately 227 of 360 documents were correctly classified after extensive modification. This is a strong indication that this method can preserve some documents' semantics even after extreme modification. Further, spinning every other, third and fourth word also affected the overall classification, but with less negative impact on the overall classification. It is also evident that, as compared to removing stop words, retaining stop words improves overall classification after document spinning.

TABLE III. THE EFFECT OF SPINNING DOCUMENTS ON THE OVERALL CLASSIFICATION RESULTS FOR THE TWO DISTANCE MEASURES

	With stop words		Without stop words	
Case	**Taxicab**	**Cosine**	**Taxicab**	**Cosine**
Every Possible	59.44	63.33	58.06	62.78
Every Other	95.56	95.00	88.61	92.22
Every Third	97.22	97.50	90.28	94.17
Every Forth	99.17	99.44	93.33	97.22
No Spins	100%	99.72%	96.67%	98.61%

VI. Conclusion and Future Work

In this paper, the effectiveness of using statistical analysis techniques to detect confidential data semantics was studied. A DLP classification model was proposed based on the well-known information retrieval function TF-IDF to define terms weights. The classification was based on measuring the similarity between the documents and the category centroids. This model was tested against different scenarios in which the DLPS dealt with known, partially known, and unknown data. The overall classification indicates encouraging outcomes across all scenarios. Further, a graphical representation of the classification results was applied using SVD abstraction. The visualization provided a very useful analytical tool for studying the semantics of documents in relation to category centroids. Further, the proposed model achieved a high score of 0.99 for both precision and recall. Finally, the method was tested against modified documents. These results were also very encouraging; more that 60 percent of the modified documents were able to be identified using cosine similarity.

In the future, the use advanced data discrimination techniques such as Linear Discrimination Analysis (LDA) will be considered. Unlike SVD, which provides linear dimension reduction, LDA can deal with bending curves to solve the problem of unevenly distributed data. Further, as the results of the unsupervised techniques were encouraging, the use of a supervised classification algorithm such as SVM will be considered [23]. SVM is a well-known classifier that is widely implemented in text classifications problems. Testing the performance of the proposed DLP model is another step that will be considered in the future. Classification speed and system overheads are important factors that must be addressed.

References

[1] Datalossdb. (2015). *Data loss statistics*. Available: http://datalossdb.org/

[2] C. Arthur and K. Stuart, "PlayStation Network users fear identity theft after major data leak," *The Guardian*, 2011. Available: http://www.theguardian.com/technology/2011/apr/27/playstation-users-identity-theft-data-leak

[3] J. Wakefield, "eBay faces investigations over massive data breach," *BBC News*, 2014. Available: http://www.bbc.com/news/technology-27539799

[4] P. Karhula, "What is the effect of WikiLeaks for Freedom of Information?," *FAIFE Spotlight [online],* vol. 19, 2011.

[5] S. Matic, A. Fattori, D. Bruschi, and L. Cavallaro, "Peering into the Muddy Waters of Pastebin," *ERCIM News: Special Theme Cybercrime and Privacy Issues,* p. 16, 2012.

[6] M. S. Bernstein, A. Monroy-Hernández, D. Harry, P. André, K. Panovich, and G. G. Vargas, "4chan and/b: An Analysis of Anonymity and Ephemerality in a Large Online Community," in *ICWSM*, 2011.

[7] A. Shabtai, Y. Elovici, and L. Rokach, *A survey of data leakage detection and prevention solutions*. Springer Science & Business Media, 2012.

[8] R. Mogull and L. Securosis, "Understanding and selecting a data loss prevention solution," *Technicalreport, SANS Institute,* 2007.

[9] McAfee. (2015). *McAfee Total Protection for Data Loss Prevention (DLP)*. Available: http://www.mcafee.com/au/products/total-protection-for-data-loss-prevention.aspx

[10] J. Oliver, C. Cheng, and Y. Chen, "TLSH--A Locality Sensitive Hash," in *Cybercrime and Trustworthy Computing Workshop (CTC)*, *2013 Fourth*, 2013, pp. 7–13.

[11] M. Datar, N. Immorlica, P. Indyk, and V. S. Mirrokni, "Locality-sensitive hashing scheme based on p-stable distributions," in *Proceedings of the Twentieth Annual Symposium on Computational Geometry*, 2004, pp. 253–262.

[12] V. Roussev, "Data fingerprinting with similarity digests," in *Advances in Digital Forensics VI*. Springer, 2010, pp. 207–226.

[13] W. B. Cavnar and J. M. Trenkle, "N-gram-based text categorization," *Ann Arbor MI,* vol. 48113, pp. 161–175, 1994.

[14] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information processing & management,* vol. 24, pp. 513–523, 1988.

[15] Y. Shapira, B. Shapira, and A. Shabtai, "Content-based data leakage detection using extended fingerprinting," *arXiv preprint arXiv:1302.2028,* 2013.

[16] M. Hart, P. Manadhata, and R. Johnson, "Text classification for data loss prevention," in *Privacy Enhancing Technologies*, 2011, pp. 18–37.

[17] P. K. Pearson, "Fast hashing of variable-length text strings," *Communications of the ACM,* vol. 33, pp. 677–680, 1990.

[18] R. Salakhutdinov and G. Hinton, "Semantic hashing," *International Journal of Approximate Reasoning,* vol. 50, pp. 969–978, 2009.

[19] G. Salton, A. Wong, and C.-S. Yang, "A vector space model for automatic indexing," *Communications of the ACM,* vol. 18, pp. 613–620, 1975.

[20] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval* vol. 1. Cambridge: Cambridge University Press, 2008.

[21] G. H. Golub and C. Reinsch, "Singular value decomposition and least squares solutions," *Numerische Mathematik,* vol. 14, pp. 403–420, 1970.

[22] BestFreeSpinner.com. (2013). *BFS pro v1.0* Available: http://bestfreespinner.com/

[23] H. Kim, P. Howland, and H. Park, "Dimension reduction in text classification with support vector machines," *Journal of Machine Learning Research*, 2005, pp. 37–53.