Highly Discriminative Features for Phishing Email Classification by SVD

Masoumeh Zareapoor, Pourya Shamsolmoali and M. Afshar Alam

Abstract Unstructured text documents have drawn recently more attention, because with growing amount of text documents, there is a need to classify them automatically. But an important problem in field of text categorization is the huge dimensional and very sparse dataset which hurts generalization performance of classifiers. This paper presents a Singular Value Decomposition (SVD) technique to email classification, in order to compress optimally only the kind of documents (in our experiments email classes) and to retain the most informative and discriminate features from an email document. The performance evaluation is performed on email dataset which is publicly available to demonstrate the benefit of the LSA.

Keywords Data mining · Dimension reduction · Email classification · Feature extraction

1 Introduction

In data mining technique, where the aim is to "find unknown and potentially interesting patterns in large databases a common task is automatic classification" [1]. Text classification has been an important application due to the very large amount of text documents that we have to deal with daily. Several popular techniques have been used for text categorization. These techniques are based on the "vector space" model for representing each document as vector [2]. One of the important examples of text which most of people deal with it is email. In recent

M. Zareapoor · P. Shamsolmoali (✉) · M. Afshar Alam
Department of Computer Science, Jamia Hamdard University, New Delhi, India
e-mail: pshams@jamiahamdard.ac.in

M. Zareapoor
e-mail: mzarea@jamiahamdard.ac.in

M. Afshar Alam
e-mail: aalam@jamiahamdard.ac.in

J.K. Mandal et al. (eds.), *Information Systems Design and Intelligent Applications*,
Advances in Intelligent Systems and Computing 339,
DOI 10.1007/978-81-322-2250-7_65

years, e-mails have become a common medium of communication for most internet users. When classifying the emails, often the data contained in emails are very complex, multidimensional [3]. Then, the uses of dimensionality reduction techniques are useful in the "classification task in order to avoid the curse of dimensionality". Generally an e-mails can be categorized into three [4]—"Ham, Spam and Phishing". Ham is legitimate e-mail while spam is "an unsolicited email". On the other hand phishing is an unsolicited, deceitful, and potentially harmful email. Generally phishing emails, "depend on forged email that pretence from a legitimate company or financial institution". Then, through a link within the email, the phisher attempts to forward users to fake Websites. These fake Web sites are designed to "deceptively obtain financial data (usernames, passwords, credit card numbers, and personal information, etc.) from genuine users" [5]. Victims of e-banking phishing email expose their bank account number, password, credit card number, and other important information needed for financial transaction to the attacker. "The attacker then misuses this information to make transactions from the victims account. This issue not only affects normal users of the internet, but also causes a big problem for companies and organizations those are misused by the attackers". In our experiments, we use 10-fold cross validation technique. In order to have a better overview of the performance of the PCAI and LSAI, we present a comparison with the SMO classifier, a popular Support Vector Machine (SVM) with good behavior in text document classification.

2 Related Work

Numerous techniques have been developed "to overcome the phishing attack problem". They include "black listing and white listing [6], network and content based filtering [7], client and server side tool bars [3, 7]". The first technique consists of lists of "malicious phishing websites (the black list) and lists of legitimate non-malicious websites (white list), where each link in a message must be checked in both lists". PhishTank [1] is a corpus of "URLs of suspected websites that has been reported as phishing attack which is commonly used by the researchers". Email providers block phishing emails if the message body contains of PhishTank URLs. Network level protection is usually achieved by blocking a series of IP addresses or set of domains from entering the network [8]. In all these research works, one of the main problem of email classification is highly dimensionality of features, because texts are often represented by a large vocabulary of individual terms. Thus dimensionality reduction has been popular since the early 90 s in text processing tasks [2, 9] like, the technique of latent semantic analysis (LSA) [10]. LSA is an application of "principal component analysis" (PCA) where a document is represented along its "semantic axes". In a text categorization task, documents are represented by a LSA vector model both when training and testing the categorization system. The computation of the latent components that represent correlated features is very valuable.

3 Dimensionality Reduction Techniques

In text classification tasks, the documents or examples are represented by thousands of tokens, which make the classification problem very hard for many classifiers. Dimensionality reduction is a typical step in many data mining problems, which transform our data representation into a "shorter", more compact, and more predictive one [2, 11]. The new space is easier to handle because of "its size", and also to carry the most important part of the information needed to distinguish between emails, allowing for the "creation of profiles that describe the data set". In this paper, we are concentrating on binary classification problem, where we want to distinguish phishing emails from legitimate. Our long vector data are represented in "highly discriminative features, which can deal with an amount of noise and heterogeneity" in the data. For these reasons we used two well-known approaches: Principal Component Analysis (PCA) [2, 12] and Latent Semantic Analysis (LSA) [10], which "involves obtaining the principal components into the term-to-document sparse matrix".

3.1 Principal Components Analysis (PCA)

PCA is a well known technique that can reduce the dimensionality of data by "transforming the original attribute space into smaller space". In the other word, the purpose of principle components analysis is to "derive new variables" that are combinations of the original variables and are uncorrelated. This is achieved by transforming the "original variables" Y = [y1, y2, …, yp] (where p is number of original variable) to a "new set of variables", T = [t1, t2, …, tq] (where q is number of new variables), which are combinations of the original variables. Transformed attributes are framed by first; "computing the mean ($\mathcal{M}$) of the dataset, then covariance matrix" of the original attributes is calculated as follow [2, 9]:

$$Covariance = \frac{1}{n(Y - \mathcal{M})^T (Y - \mathcal{M})}$$

And the second step is, "extracting its eigenvectors". The eigenvectors [13] (principal components) introduce as a "linear transformation from the original attribute space to a new space in which attributes are uncorrelated". Afterward, the obtained eigenvectors can be sorted "according to the amount of variation in the original data". The best "n eigenvectors" (those one with highest eigenvalues) are selected as new features while the rest are discarded. A principal component is the "unsupervised method" that is mean it is no use of the class attribute. One of the main pitfalls of standard Principal Components Analysis (PCA) is the "expensive time" which it requires to perform an "eigenvalue decomposition" to find the PCs. But in this paper we use a relation of the covariance matrix with the Singular Value

Decomposition (SVD) [14] instead of compute the eigenvectors directly from Co, since "SVD is less restrictive".

3.2 Latent Semantic Analysis (LSA)

Generally, LSA analyzes "relationships between a term and concepts" which is contained in an unstructured collection of text. It is called Latent Semantic Analysis, because of "its ability to correlate semantically related terms that are latent in a text". LSA produces a set of concepts, "which is smaller in size than the original set, related to documents and terms" [10]. LSA are computed by using "SVD (Singular Value Decomposing)" to identify pattern between the "terms and concepts contained in the text, and find the relationships between documents". The method commonly referred as "concept searches". It has ability to "extract the conceptual content of a body" of text by "establishing associations between those terms that occur in similar contexts". LSA is mostly used for "page retrieval systems and text clustering purposes". LSA overcomes two of the most problematic keyword queries: "multiple words that have similar meanings and words that have more than one meaning".

4 The Classic SVD Method

In this Section, we provide the basic methodology, which is usually followed for text categorization. We defined a dictionary which contains all the unique words of all documents (emails) in the dataset. The value of each dimension in a document's vector is the frequency of a specific word in that document. The words are also called "terms"; the dimensions' values are called "term frequencies". In the following, we show how vector space model is applied in the following three documents [14]: A: "Sun is a star". B: "Earth is a planet". C: "Earth is smaller than the Sun".

So, the dictionary is defined as: "D = [a, Earth, is, planet, smaller, star, Sun, than, the]". As is clear, the length of the vector in above documents is 9. The frequency vectors are:

$$\begin{matrix} A = [1, 0, 1, 0, 0, 1, 1, 0, 0] \\ B = [1, 1, 1, 1, 0, 0, 0, 0, 0] \\ C = [0, 1, 1, 0, 1, 0, 1, 1, 1] \end{matrix} \Rightarrow \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}_{3\times 9}$$

Frequency values are 0 or 1 because the size of our document is very small. Some words or terms, like "a and is, are found not that much useful information that helps the document categorization". If we can remove them from matrix then the categorization will be done more effectively. For this purpose, stop words technique can be used, "which is a list of words that will be ignored during the creation of the dictionary". All the previous methods, they used "stop word removal technique, for

eliminating the noise and redundancy". But in this paper, instead of using stop words, we applied direct, "dimensionality reduction technique" for removing the noise through SVD (Singular Value Decomposition). If applying the SVD technique to "an [r × c] matrix M" [14], it will be analyzed to a "product of three matrices" like: an [r × r] "orthogonal matrix U", a [r × c] "diagonal matrix W" and the transpose of a [c × c] "orthogonal matrix V". The "SVD formula is: $M_{r\times c} = U_{r\times r}W_{r\times c}V^{T}_{c\times c}$".

Next, we will use SVD to "compress the size of dataset to convert to the small space vector as well as compact one". Each row of the table is a documents or emails. Each column is the unique words or terms. Each cell of this matrix is frequency vector. Based on SVD, we calculate a score from 0 to 100 for each term or word. The lower "the score is the more similar to noise and can be removed from the dataset".

5 Experiments and Results

5.1 Corpora

The public email corpora which we used for performing our tests are (Table 1): SpamAssassin (SA),[1] and the Phishing Corpus (PC).[2]

5.2 Preprocessing

An email consists of two parts, header and body message. The header contains information about the message such as, sender, receiver, subject, servers, etc. The body contains the message and usually is one of two forms: HTML or plain-text [15]. The HTML emails contain a "set of tags to format the text to be displayed on screen". For building the dictionary of the email messages as we explained in Sect. 3, we used SVD technique (instead of stop word removal technique) for removing the words which do not have significant importance in "building the classifiers". Meanwhile, we use the well-known Term Frequency-Inverse Document Frequency (TF-IDF) scheme [9] for creating TDF matrix. At the end we obtain the message matrices $X_{2750\times 2173}$, where each row in the matrix corresponds to a document (e-mail) and each column corresponds to a term (word) in the document. Each cell represents the frequency (number of occurrence) of the corresponding word in the corresponding document. The obtained matrix (X) is the ones used to perform the PCA and LSA based on SVD. The vector that is generated

[1] Available at: http://spamassassin.apache.org/publiccorpus.

[2] Available at: http://monkey.org/~jose/wiki/doku.php?id=PhishingCorpus.

Table 1 Number of emails for per corpus

Corpus	Phishing	Ham	Total
SA		1,800	
PC	950		
			2,750

Table 2 Performance of the different methods over the 10-folds

Methods	Number of features				
PCA I	50	170	500	1,500	2,750
LSA I	50	170	500	1,500	2,750
PCA	50	170	500	1,500	2,750
LSA	50	170	500	1,500	2,750

in this stage is considered as long vector, and decreasing the size of these vectors to short vector with dimensional reduction techniques.

5.3 Classification Model

In this stage, we want to build the suitable classifier, in order to compare the performance of the PCA and LSA techniques which are based on SVD. After several comparison and trail we choose to use SMO classifier [16], a linear SVM which has very good performance in sparse data and which is well suited for text classification. For building SMO implementation we used following settings: a lineal kernel (polynomial with exponent 1); complexity constant equal to 100 (Table 2).

5.4 Evaluation Matrices

In this paper for better overview, the results are presented in the form of the area under the Receiver Operating Characteristic (ROC) curve, which aims at a "high true-positive rate and a low false-positive rate". We also provide results in terms of accuracy of the classification for better understanding. As we can see from the Figs. 1 and 2 the PCA and LSA obtain the good result in detecting phishing email while only using a large numbers of features. In the other hand, they have a good performance only with more features. When we applying the PCA for reducing the dimension in dataset, then the new feature space which are obtained are not that much discriminative for classes. But if we use SVD eigenvalues for both technique (LSAI, PCAI), they need commonly much less features to obtain a good classification. It means that for these proposed techniques choosing more features does not

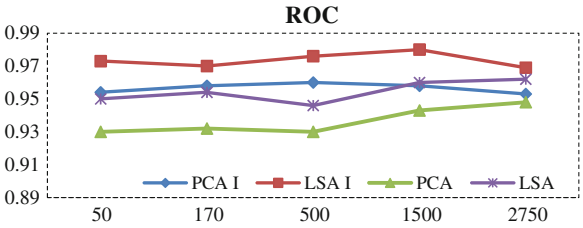
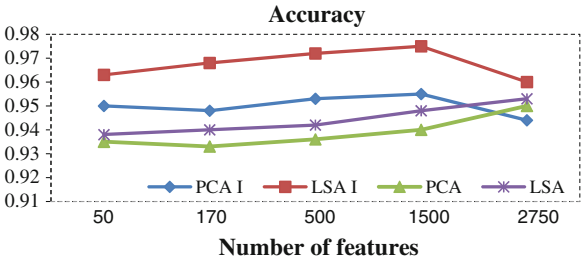


Fig. 1 Performance of LSA and PCA in term of ROC

Fig. 2 Performance of LSA and PCA in term of accuracy



have effect or might degrade the performance of the classifiers. Also from these results, we can observe that the LSAI features extracted techniques are well suited to discriminate between ham and phishing emails.

6 Conclusion

In this paper we presented and evaluated a novel technique based on PCA and LSA which are two well known dimensional reduction technique, and better known in text classification. In our proposed technique, we did not use any traditional technique for removing the useless information from the dataset like stop world removal technique. We used SVD technique for reducing the noise and dimension from original dataset. And from the results, found that PCAI and LSAI are having good performance when the number of feature is less. It means that, the SVD technique can find the very discriminative features from dataset. The results show good classification performance when using the PCA based on SVD techniques 10-fold cross-validation.

References

1. PhishTank: Available from: http://www.phishtank.com (2014). Accessed 14 Jan 2014
2. Gomez, J.C., Moens, M.F.: PCA document reconstruction for email classification. Comput. Stat. Data Anal. **56**, 741–751 (2012)
3. Verbeek, J.J.: Supervised feature extraction for text categorization
4. Basnet, R., Sung, A.H.: classifying phishing emails using confidence-weighted linear classifiers. In: Proceedings of the International Conference on Information Security and Artificial Intelligence (ISAI), pp. 108–112. (2010)
5. Zareapoor, M., Seeja K.R.: Text mining for phishing email classification. In: Intelligent Computing, Communication and Devices, pp. 65–71. Springer, Heidelberg (2015)
6. Huillier, G.L., Weber, R., Figueroa, N.: Online phishing classification using adversarial data mining and signaling games. In: Proceedings of the ACM SIGKDD (2009)
7. Ramanathan, V., Wechsler, H.: Phishing detection and impersonated entity discovery using conditional random field and latent dirichlet allocation. J. Comput. Secur. **34**, 123–139 (2013)
8. Snort: Network intrusion prevention and detection system. Available from: http://www.snort.org/ (2014). Accessed 03 Feb 2014
9. Gomez, J.C., Boiy, E., Moens, M.F.: Highly discriminative statistical features for email classification. Knowl. Inf. Syst. **31**, 23–53 (2012)
10. Huillier, G.L., Hevia, A., Weber, R., Rios, S.: Latent semantic analysis and keyword extraction for phishing classification. Department of Computer Science, University of Chile (2010)
11. Kim, H., Howland, P., Park, H.: Dimension reduction in text classification with support vector machine. J. Mach. Learn. Res. **6**, 37–53 (2005)
12. Biricik, G., Diri, B., Sonmez, A.C.: Abstract feature extraction for text classification. Turk. J. Elec. Eng. Comp. Sci. **20**, 1137–1159 (2012)
13. Tsymbal, A., Puuronen, S., Pechenizkiy, M., Baumgarten, M., Patterson, D.W.: Eigenvector-based feature extraction for classification. In: Haller, S.M., Simmons, G. (eds.) Paper presented at the FLAIRS conference, pp. 354–358. AAAI Press, Menlo Park (2002)
14. Symeonidis, P., Kehayov, I., Manolopoulos, Y.: Text classification by aggregation of SVD eigenvectors. In: Proceedings of the 16th East European Conference on Advances in Databases and Information Systems, pp. 385–398. Springer, Heidelberg (2012)
15. Akinyelu, A.A., Adewumi, A.O.: Classification of phishing email using random forest machine learning technique. J. Appl. Math. (2014)
16. Platt, J.C.: Fast training of support vector machines using sequential minimal optimization, pp. 185–208. MIT Press, Cambridge (1998)