An Efficient Unsavory Data Detection Method for Internet Big Data

Peige Ren[1(✉)], Xiaofeng Wang[1], Hao Sun[1], Fen Xu[2],
Baokang Zhao[1], and Chunqing Wu[1]

[1] College of Computer, National University of Defense Technology,
Changsha 410073, China
renpeige@163.com, sunhao4257@gmail.com,
{xf_wang, bkzhao, chunqingwu}@nudt.edu.cn

[2] China Aerodynamics Research and Development Center,
Hypervelocity Aerodynamics Institute, Mianyang, China
fenxl5@163.com

Abstract. With the explosion of information technologies, the volume and diversity of the data in the cyberspace are growing rapidly; meanwhile the unsavory data are harming the security of Internet. So how to detect the unsavory data from the Internet big data based on their inner semantic information is of growing importance. In this paper, we propose the i-Tree method, an intelligent semantics-based unsavory data detection method for internet big data. Firstly, the internet big data are mapped into a high-dimensional feature space, representing as high-dimensional points in the feature space. Secondly, to solve the "curse of dimensionality" problem of the high-dimensional feature space, the principal component analysis (PCA) method is used to reduce the dimensionality of the feature space. Thirdly, in the new generated feature space, we cluster the data objects, transform the data clusters into regular unit hyper-cubes and create one-dimensional index for data objects based on the idea of multi-dimensional index. Finally, we realize the semantics-based data detection for a given unsavory data object according to similarity search algorithm and the experimental results proved our method can achieve much better efficiency.

Keywords: High-dimensional feature space · Principal component analysis · Multi-dimensional index · Semantics-based similarity search

1 Introduction

In recent years, with the era of Internet big data coming, the volume and diversity of internet data objects in the cyberspace are growing rapidly. Meanwhile, more and more various unsavory data objects are emerging, such as various malwares, violent videos, subversive remarks, pornographic pictures and so on [1–3]. The unsavory data are harming our society and network security, so efficiently detecting the unsavory data objects from the Internet big data is of growing importance. But the traditional accurate matching based data detection methods cannot identify the inner semantic information of various internet data objects and cannot realize intelligent data detection.

I. Khalil et al. (Eds.): ICT-EurAsia 2015 and CONFENIS 2015, LNCS 9357, pp. 213–220, 2015.
DOI: 10.1007/978-3-319-24315-3_21

To realize the intelligent semantics-based data detection, we need to extract the features of internet data collection to construct a high-dimensional feature space [4], and the data objects are expressed as high-dimensional points in the feature space, so we can discover the data objects semantics-similar to a given query data object (unsavory data) based on the distances between high-dimensional points. While the efficiency of semantics-based similarity search in feature space is sensitive to the dimensionality of the space, when the dimensionality is too high, the efficiency of similarity search can be so worse that cannot meet our needs.

In the other hand, when searching the semantics-similar data objects to a given query point in feature space, the multi-dimensional indexes [5] can prune away the data objects semantics-irrelevant (with large distance) to the query point, reducing the searching zone and searching paths and increasing the efficiency of similarity search. While the existing multi-dimensional indexes have several shortcomings for processing the internet big data: Firstly, the multi-dimensional indexes are influenced by the dimensionality of feature space, when the dimensionality is very high, the efficiency of multi-dimensional indexes might become worse than sequential scan; Secondly, most existing multi-dimensional indexes are proposed in some particular situations. For instance, the Pyramid-Technique [6] is efficient at processing uniformly distributed data set but inefficient when data set is irregularly distributed, while the iDistance [7] method is efficient at kNN query but cannot carry out range query; Thirdly, for the semantics-based similarity searching in feature space, the semantic information of data set is usually embedded in a lower-dimensional subspace so the original high-dimensional feature space can be compressed. Besides, there are many correlated features, noise and redundant information in the original feature space, which can impair the efficiency of semantics-based similarity search.

To realize intelligent and efficient unsavory data detection for internet big data, we proposed the i-Tree method, a semantics-based data detection method. The method firstly utilize the PCA [8] method to reduce the dimensionality of original high-dimensional feature space, eliminating ill effects of “curse of dimensionality” meanwhile diminishing the redundant and noise interference; secondly, we adopt a multi-dimensional index which is robust for arbitrarily distributed data set in the feature space, the index can effectively divide, organize and map multi-dimensional data objects into one-dimensional values; finally, to validate the validity of our method, we realize similarity search algorithm using our method and compare our method with other classic methods. Our method can avoid “curse of dimensionality”, and the method is adaptive for various distributed data set, which can provide inspiration to efficient unsavory data detection for internet big data.

The rest of the paper is organized as follows. Section 2 introduced the related technologies and methods of this paper. Section 3 proposed the semantics-based unsavory data detection method for internet big data based on PCA and multi-dimensional indexes. Section 4 presents the experimental results of our method. Finally we conclude in Sect. 5.

2 Related Work

2.1 Principal Component Analysis

Principal component analysis (PCA) is widely used in analyzing multi-dimensional data set, which can reduce the high dimensionality of original feature space to a lower intrinsic dimensionality, and can realize redundancy removal, noise elimination, data compression, feature extraction, etc. The PCA is widely employed in many actual applications with linear models, such as face recognition, image processing, sex determination, time series prediction, pattern recognition, communications, etc.

The basic idea of PCA is representing the distribution of original date set as precisely as possible using a set of features that containing more amount of information, in other words, it computes an orthogonal subspace with lower dimensionality to represent the original high-dimensional data set.

For an N-dimensional data set X containing M data objects (expressed as N-dimensional vectors): $x_k \in R^{N\times 1}(k = 1, 2, \ldots, M)$, let m represent the mean vector: $\mathrm{m} = \frac{1}{M}\sum_{i=1}^{M} x_k$, and the covariance matrix can be represented as $\mathrm{S}_i = \frac{1}{M}\sum_{k=1}^{M}(x_k - m)(x_k - m)^T \in R^{N\times N}$. Let $\mathrm{Z} = [x_1 - \mathrm{m}, x_2 - \mathrm{m}, \ldots, x_M - \mathrm{m}\ ] \in R^{N\times M}$, then $\mathrm{S}_i = \frac{1}{M}ZZ^T \in R^{N\times N}$.

The optimal projected vectors of PCA are a set of orthonormal vectors (u_1, u_2, …, u_d) when the evaluation function $\mathrm{J(u)} = u^T S_i u$ attends its maximal value, and the retained variance in the projection is maximal. In fact, the vectors u_1, u_2, …, u_d are the corresponding orthonormal eigenvectors of d larger eigenvalues of S_i $(\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_d)$, the vector u_i is also called the i-th principal component. The evaluation function of PCA can also be represented as $\mathrm{J(W)} = tr(W^T S_i W)$, where the optimal projection matrix is $W_{opt} = \arg\max_W J(W) = (u_1, u_2, \ldots, u_d)$.

The contribution rate (energy) of k-th principal component u_k can be defined as: $\frac{\lambda_k}{\lambda_1+\lambda_2+\ldots+\lambda_n}$, that is the rate of the k-th principle component variance in the sum of all principle component variances. Owing to $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$, the contribution rate of anterior principal component is greater than the contribution rate of latter principal component. When the contribution rate sum of d anterior principal components: $\eta_k = \frac{\lambda_1+\lambda_2+\ldots+\lambda_d}{\lambda_1+\lambda_2+\ldots+\lambda_k+\ldots+\lambda_n}$ is large enough (such as ≥90 %), then we can think that the d principal components almost contain all the useful information of original features. When d<< n, we can achieve the goal of dimensionality reduction of original feature space, then we can construct a lower-dimensional feature space using the eigenvectors: u_1, u_2, …, u_d.

2.2 Multi-dimensional Indexes

The multi-dimensional indexes can efficiently divide and organize the data objects in feature space, making sure data objects that close to each other are likely to be stored in the same page, so the useless data zones can be pruned away in advance for processing similarity queries. So far, a series of multi-dimensional indexes have been proposed, the typical ones include Pyramid-Technique, iDistance, and so on.

The partitioning strategies of multi-dimensional indexes can be divided into space-based partitioning and data-based partitioning. The Pyramid-Technique is based on the space-based partitioning strategy. It firstly divides the d-dimensional feature space into 2d subspaces such that the resulting subspaces are shaped liked pyramids with the center-point of the feature space as their common top, and then every subspace is cut into slices that are parallel to the pyramid's basis to form data pages. The technique defines a pyramid number for each subspace. For a d-dimensional data object, the technique determines the pyramid number i in which the data object is located and computes the height h of the data object to the pyramid top. So we can obtain the one-dimensional mapping value of the data object through adding the pyramid number i and the height h of the data object.

The partitioning strategy of the iDistance method can be space-based partitioning strategy or data-based partitioning strategy. The iDistance firstly divides the feature space into subspaces equally or according to the distribution of the data objects and determines a subspace number i for each subspace; secondly, selects a reference point for each subspace and computes the distance d of a given data object p to its nearest reference point; finally, the data object can be mapped into a one-dimensional value y based on the formula: $y = i \times c + d$, where c is a constant to make sure that the data objects in different subspaces are mapped into different one-dimensional intervals. Finally the iDistance uses a B+-tree to index the resulting one-dimensional space.

3 The Overview of Our Method

To realize intelligent semantics-based unsavory data detection, we proposed the i-Tree method, and our method consists of the following three phases:

- Dimensionality reduction of feature space based on PCA;
- Adaptive multi-dimensional index for data distribution;
- Semantics-based similarity search.

3.1 Dimensionality Reduction of Feature Space Based on PCA

The dimensionality of internet big data's feature space is usually too high to easily cause the "curse of dimensionality". In this section we realize the dimensionality reduction of the original feature space using the PCA method, eliminating the ill effects of "curse of dimensionality" and the redundant and noise interference.

Firstly we use the PCA to compute the features of a new feature space, the features (vectors) of the new space are in the direction of the largest variance of the original internet data. Then we use the new features to construct a lower-dimensional feature space, and the internet data set are expressed in this new feature space by their feature coefficients (weights). A user query is also projected into the feature space generated by PCA, and we can find semantics-similar internet data by searching the internet data near it; if the query is not projected into the feature space, we can conclude that there is no semantics-similar internet data to the query.

By means of the PCA method, we can realize the dimensionality reduction of the original feature space, which can eliminate the impact of “curse of dimensionality”, remove the noisy and redundancy information during the similarity search and reduce the complexity of computation and storage for further data processing.

3.2 Adaptive Multi-dimensional Index for Data Distribution

In this section we divide and manage the data objects in feature space based on the idea of multi-dimensional indexes. For the internet data objects irregularly distributed in feature space, we proposed an adaptive multi-dimensional index. Our index can be realized by the following three steps: 1, partitioning the data set according to the data distribution to form a series of data clusters; 2, transforming the data clusters into regular-shaped data subspaces; 3, mapping the high-dimensional data objects into one-dimensional values and index them using a B+-tree.

Firstly, we partition the data set according to their distribution to form data clusters. The data objects distribute irregularly in feature space, usually semantics-similar data objects gather together. So we here employ the data-based partitioning strategy to partition the feature space. Specifically, we utilize the K-means clustering algorithm to cluster the data objects into a series of data clusters, the data objects in a same cluster are near to each other, having similar semantic information.

Secondly, we transform the data clusters into regular-shaped subspaces. To utilize the Pyramid-Technique to process each data cluster, we transform the data clusters into unit high-dimensional hyper-cube shaped subspaces and move the cluster centers to the centers of the unit subspaces. For the data objects in each data clusters, the transformation is a one-to-one mapping [6], so we can perform similarity search algorithm based on the Pyramid-Technique. Figure 1 shows the process of data clustering and transforming.

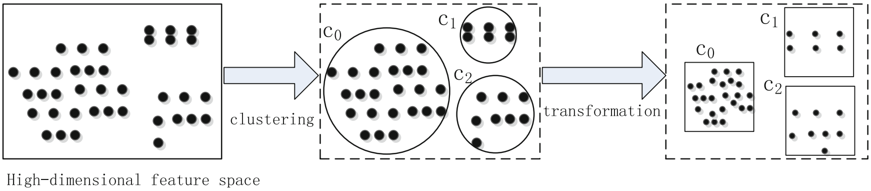

Fig. 1. The process of data clustering and transforming

Thirdly, we map high-dimensional data objects in each subspace into one-dimensional values and index them. For each hyper-cube shaped subspace, we utilize the Pyramid-Technique to map the data objects. We firstly number each subspace with subspace number i. Then we use the Pyramid-Technique to map data objects in each subspace respectively. For an m-dimensional subspace C_i, we partition the subspace into 2 m hyper-pyramids with the subspace center as their common top, and number the pyramids counterclockwise with pyramid-number j. For a high-dimensional data object v in pyramid j of subspace i, we compute the height h_v (to its top) and map v into a one-dimensional value $p_v = i + j + (0.5 - h_v)$. Using the above method, we can map all data objects in feature space into a one-dimensional space. Finally, we index the one-dimensional space using the B+-tree, the high-dimensional data objects and the corresponding one-dimensional keys are stored in the data pages of the B+-tree. The mapping of the data objects is shown in the Fig. 2.

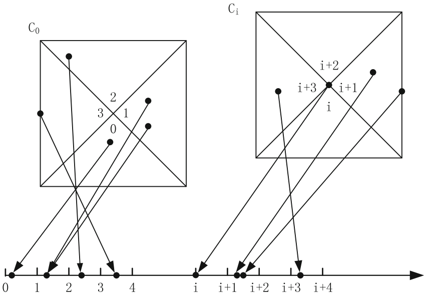

Fig. 2. Mapping of the data objects

3.3 Semantics-Based Similarity Search

In the feature space, the semantics-similar data objects are near to each other, so we can utilize the distance information to discover data objects that are semantics-similar to a given query q. In this section, we mainly study the range query.

The range query is a very popular similarity search algorithm. In this paper, the range query can be realized as follows: firstly extract the features of the query and express it as a multi-dimensional point in feature space; secondly determine the searching spaces and abandon the space that does not intersected with the query range; finally scan the data objects in searching spaces to find the right answers. The range query can be realized by the following algorithm:

Algorithm 1 Semantics-based range search

```
1. Read the range query RangeQuery (D, q, r, M).
2. Map the query range to one-dimensional space.
3. Determine which subspaces are affected with the range query according to the
   intersected zone in one-dimensional space.
4. Determine the searching spaces by reversely mapping the intersected zone to
   multi-dimensional feature space.
5. Scan the data objects in the searching spaces, find the final answers to the query q.
```

4 Performance Evaluation

In this section, we evaluate the effectiveness and efficiency of our method by analyzing the experimental results. We implement the range query based on our method in C, and chose the sequential scan and Pyramid-Technique as the reference algorithms. We choose a computer with Intel(R) Core (TM) 2 Quad CPU Q8300 2.5 GHz and 4 GB RAM, and the operating system is CentOS 5. For each experiment, we run 20 times and computed the average results as the final experimental results. For the input data set, we generate synthetically a series of clustered data sets of different data sizes and different dimensionality. The dimensionalities of the data sets are respectively 16, 32, 64 and 128, and the data size of them varies from 100000 to 2000000. We compare the response time of our method with the two reference algorithms in the same conditions. The experimental results are shown in the Figs. 3 and 4.

As shown in the Fig. 3, the input data set is a clustered 32-dimensional data set with data size varying from 100000 to 2000000, and the data set has 8 natural clusters. We can observe from the Fig. 3 that the response time of the three methods increases with the data size, while the response time of our method is less than the sequential scan and the Pyramid-Technique.

As shown in the Fig. 4, we observe the response time with the dimensionality of feature space varying from 16 to 128. Here we choose the input data set with 1000000 data objects and 8 natural clusters. We can observe that the response time of three methods increases with the dimensionality of feature space, and the sequential scan may be faster than the Pyramid-Technique when the dimensionality is high enough, but our method is more efficient than the other methods. The experimental results from Figs. 3 and 4 show that our method can effectively find the semantics-similar data objects to a given query.

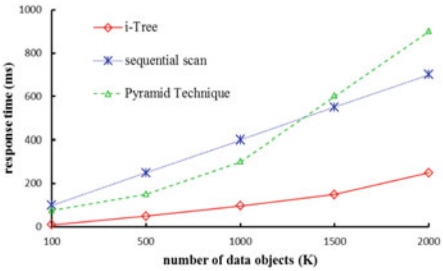

Fig. 3. Effects of data size

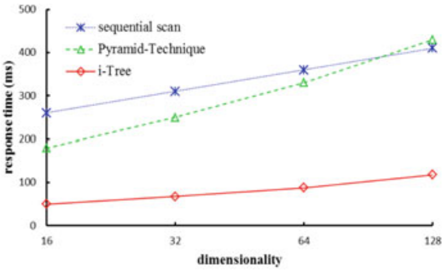

Fig. 4. Effects of dimensionality

5 Conclusion

In this paper, we proposed an efficient unsavory data detection method for Internet big data. To realize semantics-based similarity search of various unsavory data, we express the data objects as high-dimensional points in feature. To solve the problem of "curse of dimensionality" caused by the high dimensionality of feature space, we used the PCA to reduce the dimensionality of feature space. By partitioning the feature space into subspaces and transform them into unit hyper-cubes, we could utilize the Pyramid-Technique to index the data objects and realize efficient semantics-based similarity search. Finally, the performance evaluation results revealed that our method could efficiently discover the semantics-similar data objects to a given query.

Acknowledgment. The work described in this paper is partially supported by the grants of the National Basic Research Program of China (973 project) under Grant No. 2009CB320503, 2012CB315906; the project of National Science Foundation of China under grant No. 61070199, 61103189, 61103194, 61103182, 61202488, 61272482; the National High Technology Research and Development Program of China (863 Program) No. 2011AA01A103, 2012AA01A506, 2013AA013505, the Re-search Fund for the Doctoral Program of Higher Education of China under Grant No. 20114307110006, 20124307120032, the program for Changjiang Scholars and Innovative Research Team in University (No. IRT1012), Science and Technology Innovative Research Team in Higher Educational Institutions of Hunan Province ("network technology").

References

1. Fedorchenko, A., Kotenko, I., Chechulin, A.: Integrated repository of security information for network security evaluation. JoWUA **6**(2), 41–57 (2015)
2. Shahzad, R.K., Lavesson, N.: Comparative analysis of voting schemes for ensemble-based malware detection. JoWUA **4**(1), 98–117 (2013)
3. Skovoroda, A., Gamayunov, D.: Securing mobile devices: malware mitigation methods. JoWUA **6**(2), 78–97 (2015)
4. Zhan, Y., Yin, J., Liu, X.: A convergent solution to matrix bidirectional projection based feature extraction with application to face recognition. Int. J. Comput. Intell. Syst. **4**(5), 863–873 (2011)
5. Bohm, C.: Searching in high-dimensional spaces: index structures for improving the performance of multimedia databases. ACM Comput. Surv. **33**, 322–373 (2001)
6. Zhang, R., Ooi, B.C., Tan, K.L.: Making the pyramid technique robust to query types and workloads. In: Data Engineering 2004, p. 313 (2006)
7. Jagadish, H.V., Ooi, B.C.: iDistance techniques. In: Encyclopedia of GIS, pp. 469–471. Springer, New York (2008)
8. Zhan, Y., Yin, J.: Robust local tangent space alignment via iterative weighted PCA. Neurocomputing **74**(11), 1985–1993 (2011)