VNSOptClust: a variable neighborhood search based approach for unsupervised anomaly detection

Wang, Qian; Belacel, Nabil

Second International Conference on Modelling, Computation and Optimization in Information Systems and Management Sciences (MCO 2008) [Proceedings], 2008

National Research Council Canada
Conseil national de recherches Canada

Institute for Information Technology
Institut de technologie de l'information

NRC · CNRC

*VNSOptClust: Variable Neighborhood Search based approach for Unsupervised Anomaly Detection **

Wang, C. Belacel, N.
September 2008

* published in The Second International Conference on Modelling, Computation and Optimization in Information Systems and Management Sciences (MCO 2008). Metz, France. September 8-10, 2008. NRC 50406.

Canada

VNSOptClust: a Variable Neighborhood Search Based Approach for Unsupervised Anomaly Detection

Christa Wang and Nabil Belacel

National Research Council Canada, IIT-Knowledge Discovery Group,
55 Crowley Farm Road, Suite 1100
Moncton, NB E1A 7R1, Canada
{christa.wang,nabil.belacel}@nrc-cnrc.gc.ca

Abstract. In this paper, we present a new algorithm, VNSOptClust, for automatic clustering. The VNSOptClust algorithm exploits the basic Variable Neighborhood Search metaheuristic to allow clustering solutions to get out of local optimality with a poor value; it considers the statistic nature of data distribution to find an optimal solution with no dependency on the initial partition; it utilizes a cluster validity index as an objective function to obtain a compact and well-separated clustering result. As an application for unsupervised Anomaly Detection, our experiments show that (i) VNSOptClust has obtained an average detection rate of 71.2% with an acceptably low false positive rate of 0.9%; (ii) VNSOptClust can detect the majority of unknown attacks from each attack category, especially, it can detect 84% of the DOS attacks. It appears that VNSOptClust is a promising clustering method in automatically detecting unknown intrusions.

Key words: Unsupervised Learning; Automatic Partitional Clustering; Variable Neighborhood Search; Unsupervised Anomaly Detection.

1 Introduction

Network intrusion attacks pose a serious security threat in a network environment. A wide range of attacks include attempts to destabilize the whole network, to gain unauthorized access to file or privileges, and to prevent legitimate users from using a service [24]. The goals of Intrusion Detection Systems (IDSs) are to automatically detect intrusion attacks from the audit data, and to protect vulnerable network systems with cooperation of static defense mechanisms such as firewalls and software updates [22].

Given the significance of the intrusion detection problems, a number of intrusion detection approaches have been proposed. However, traditional signature-based IDSs suffer from the following drawbacks: first, known signature patterns have to be hand-coded into the systems; secondly, only known attacks that have characteristic signatures can be detected [20]. Data mining based IDSs [4], [5], [16], [19], [20] require precisely labeled data or purely normal data in order to

perform misuse detection or anomaly detection [24], [25]. In practice, neither precisely labeled data nor purely normal data is readily available.

To solve the above problems, Portnoy et al. [24] proposed the concept of the unsupervised anomaly detection clustering. The proposed method takes a set of unlabeled data as input and the clustering is performed to separate intrusions and normal instances using a distance-based metric. Once the data is clustered, the normal instances form large clusters while anomalies appear in small clusters. The main advantage of this unsupervised anomaly detection clustering algorithm is the ability to process unlabeled data and automatically detect unknown intrusions.

In this paper, we present a new Variable Neighborhood Search (VNS) based clustering algorithm, VNSOptClust, for solving the unsupervised anomaly detection problem. The VNSOptClust algorithm adopts the basic VNS principle to allow clustering solutions to get out of local optimality to reach a near-global optimum; it considers the statistical nature of data to find a near-globally optimal solution with no dependency on the initial partition status; it utilizes a cluster validity index as an objective function to obtain a compact, well-separated partition. Based on the two assumptions[1] in [24], VNSOptClust can automatically detect intrusion attacks by clustering the unlabeled data and labeling the large clusters as normal and small clusters as abnormal, respectively. The simulation on the subsets of KDD-99 Cup dataset suggests that VNSOptClust is effective in distinguishing anomalies in the dataset from the normal ones. It has obtained an average detection rate of 71.2% with an acceptably low false positive rate of 0.9%. In addition, VNSOptClust can detect the majority of unknown attacks from each attack category. Especially, it can detect 84% of the DOS attacks. Therefore, it appears that VNSOptClust is a promising clustering method in automatically detecting unknown intrusions.

The remainder of the paper is organized as follows: In section 2, the related work in the cluster analysis is reviewed. We confine our discussion on the partitional clustering methods for unsupervised anomaly detection. In Section 3, the VNSOptClust algorithm is introduced in detail. In Section 4, experimental results are reported. Finally, the conclusion is drawn.

2 Related Work in Cluster Analysis

Clustering is a discipline aimed at automatically revealing and describing homogeneous groups or clusters in a dataset. The objective of clustering is that the objects within a group be similar to each other and different from the objects in other groups. In general, the clustering methods can be broadly classified into two categories: hierarchical clustering and partitional clustering. Hierarchical clustering methods build a tree structure for a nested sequence of partitions

[1] Two assumptions: First, the number of normal instances is overwhelmingly larger than the number of intrusions; second, the intrusive instances are qualitatively different from the normal ones.

whereas Partitional clustering methods produce a single partition. In this paper, we will confine our discussion on partitional clustering problems.

The most popular partitional methods are K-means and its variants. K-means is an iterative hill-climbing algorithm and the solution obtained depends on the initial partition status (initial number of clusters with initial centroid seeds). In order to detect the optimal number of initial clusters, an expensive fine-tuning process is necessary. In addition, K-means is often stuck in a local optimum with a poor value and fails to converge to a global optimum. To tackle the shortcomings of K-means, a number of clustering methods have been proposed [8], [11], [24]. The H-Means+ algorithm, an improved version of K-means, eliminates the farthest point that currently contributes most to the total Sum of Squared Errors to improve the clustering performance [11]. In [24], the authors proposed an algorithm for automatic clustering. The algorithm uses a single-linkage clustering, which starts with an empty set of clusters and updates it iteratively. For each data instance, if its distance to the centroid of the selected cluster is less than predefined constant(Cluster Width) then this data instance is assigned to that cluster. Otherwise, a new cluster is created. However, Portnoy's algorithm still requires the proper values of to be predefined manually for each given dataset. To perform automatic clustering without predefining any constants, Guan et al. [8] introduced the Y-means algorithm. Y-means applies postprocessing strategies to adjust the initial number of clusters so that the initial number of clusters and initial centroid seeds are not crucial to the clustering solutions. However, Y-means still belongs to the category of local search heuristics. It often terminates at a local optimum with no guarantee convergence to a global optimum.

To improve the convergence of the clustering performance, several metaheuristic-based optimization methods have been introduced to solve the global optimization problem. The philosophy of such metaheuristic methods is to efficiently explore the search space, to escape from local optima, and to find a near-optimal solution[2]. Among them, Simulated Annealing (SA) [26], Tabu Search (TS) [1], and Genetic Algorithms (GAs) [2], [9], [18], [22], [29] are the commonly-used methods in solving the global optimization problem. However, the main drawbacks of such metaheuristic-based clustering algorithms are parameter selection and high computational complexity [32]. An ideal clustering algorithm should be able to automatically detect near-globally optimal clusters in reasonable time with no dependency on the initial number of clusters and the initial centroid seeds and no need of critical parameter selection.

Variable Neighborhood Search (VNS) is a newly proposed metaheuristic method for solving combinatorial and global optimization problems [12]. The basic principle of VNS is to proceed to a systematic change of neighborhoods within a local search routine. In comparison with other metaheuristics, VNS has several advantages [14]: (i) In VNS, there are no critical parameters to be defined

[2] Since finding the exact global solutions of the clustering problem in a reasonable amount of computational time is an NP-hard problem [27], the goals of solving the global optimization problem are to allow clustering solutions to get out of local optima and to provide near-optimal solutions in reasonable time.

while retaining its efficiency and effectiveness. (ii) VNS can provide near-optimal solutions in moderate computing time.

Inspired by the successful applications of VNS (e.g., Traveling Salesman Problem [13], p- median Problem [10], Minimum Sum-of-Squares Clustering Problem [11], Multi-source Weber Problem [6], [7], and Fuzzy Clustering Problem [3]), we have developed a VNS-based clustering algorithm, VNSOptClust, in automatically searching optimal clusters [31]. In this paper, we will apply VNSOptClust to solve the Unsupervised Anomaly Detection problem.

3 The VNSOptClust Algorithm

VNSOptClust is developed from the basic VNS principle [12]. The basic idea of VNSOptClust is to proceed to a systematic change of neighborhoods within a local search routine. The search is centered around the current best solution and explored increasingly distant neighborhoods until a better solution is found, and then jumped there. VNSOptClust is an optimization process controlled by a random perturbation routine, in which both descend to local optimal and escape from local optima are reached. In this way, VNSOptClust allows clustering solutions to get out of local optima and converge to a near-global optimum. Moreover, VNSOptClust considers the statistical nature of data distribution, eliminating the effect of outliers in clustering procedures, and handling the appearance of empty clusters. Unlike traditional local search methods, VNSOptClust is not sensitive to the initial number of clusters and initial centroid seeds. The general steps of VNSOptClust can be described as follows:

Step 1: *Initialization*
(1) *Assignment*: Partition the normalized data instances ($I_j, j = 1, 2, ..., n$, n is the total number of data instances in the dataset) into p(arbitrary initial number of clusters, $p \in [2, 3, ..., n]$)clusters.
(2) *Remove Empty Clusters*: For each of p clusters, check for empty clusters. If there are, remove them. The resulting number of clusters is p_1.
(3) *Splitting*: For each cluster $C_i, i = 1, 2, ...p_1$, identify outliers based on the splitting condition[3] and replace them as centroids of new clusters.
(4) Let P_M and f_{opt}[4] be the current incumbent partition and the current objective value for VNS heuristic search; choose stopping condition t_{max}(maximum running time for the VNS heuristic search) and a value for the parameter k_{max}(the maximum number of Neighborhoods to be searched).

Step 2: *Termination (Outer Loop)*
If the stopping condition is met, then stop.

[3] Splitting condition: please refer to Section of Splitting for details.
[4] The Objective function: We have employed Dunn's Index, the Davies-Bouldin Index, and Silhouette Validity Index respectively as an objective function and found the clustering results are irrespective with the index being used.

Step 3: *First Neighborhood around current incumbent solution*
Set $k = 1$, k is the current searching neighborhood.

Step 4: *Inner Loop*
If $k > k_{max}$ or $2k > |c|$, where $|c|$ is the number of clusters in the current solution, then return to Step 2 and stop.

Step 5: *Perturbation*
Randomly choose k pairs of clusters from the current solution , and then merge k pairs of clusters into k clusters; denote the so-obtained partition with P_M^1.

Step 6: *Local Search*
(1) *Merging*: With P_M^1 as the initial solution, merge any two clusters in P_M^1 based on the merging condition[5]. The resulting number of clusters is p_2.
(2) *Assignment*: Partition the normalized data instances into p_2 clusters.
(3) Denote the resulting partition and the objective value with P_M^2 and f_{new} respectively.

Step 7: *Move or Not*
If f_{new} is better than f_{opt}, then recenter the search around the new solution P_M^2: Set $f_{opt} = f_{new}$ and $P_M \leftarrow P_M^2$, and go to Step 3. Otherwise, set $k = k + 1$ and go to Step 4.

It should be noted that VNSOptClust does not require any critical parameters to be defined. Since VNSOptClust can automatically detect optimal clusters with no dependency on the initial number of clusters [31], the value of the initial number of clusters is not sensitive to the clustering result. Parameters t_{max}, k_{max} are defined based on the users' expectation of how much time and how far the VNS heuristic search performs. In our experiment, we used $t_{max} = 2$ seconds and $k_{max} = 10$.

As observed, several strategies have been employed in VNSOptClust. VNSOptClust has taken into consideration the statistical nature of data distribution to identify and remove outliers to improve the clustering performance; its effectiveness has been implemented through the procedures of perturbation and local search. We therefore present those strategies in the remainder of the section.

Splitting. The purpose of the splitting procedure is to identify outliers, to remove outliers from each cluster, and to replace them as centroids of new clusters. As the Euclidean distance is used to measure the similarity between any two data points, outliers can be treated as data points that are far from the cluster centroid. VNSOptClust takes into consideration the statistical nature of data distribution and applies the Chebyshev's Theorem to determine the splitting threshold.

[5] Merging condition: please refer to Section of Local Search for details.

Chebyshev's Theorem
For any data distribution, at least $(1 - 1/n^2)$ *of the observations of any set of data lies within* n *deviations of the mean, where* n *is greater than 1.*[30]

By applying Chebyshev's Theorem, we observe that at least 94% of data objects lie within 4 standard deviations of the mean when $n = 4$. It can be assumed that, given majority of data objects (94%) lie within 4 standard deviations of the cluster centroid, the data objects that stay beyond the threshold 4σ can be identified as outliers. Hence, we can define our splitting condition as follows: given the cluster centroid, if any data point within the cluster whose distance from the cluster centroid is greater than the threshold $d = 4\sigma$, then this data point can be identified as an outlier. VNSOptClust removes the identified outlier from the cluster and replaces it as the centroid of a new cluster. The splitting procedure is repeated until no outliers exist.

Perturbation. The objective of the perturbation stage in the VNS heuristic search is to provide a good start for the local search heuristic. To implement the diversification of the VNS heuristic search, the perturbation step randomly selects starting points from the increasingly distant neighborhoods of the current best solution. The process of changing neighborhoods with increasing cardinality in case of no improvements yields a progressive diversification. Perturbation is critical for the VNS heuristic search since choosing random starting points in the neighborhoods of the current best solution is likely to produce a solution that maintains some good features of the current best one.

In VNSOptClust, the local search routine employs the idea of merging two closest clusters. To implement the diversification of the VNS heuristic search, the perturbation step in VNSOptClust randomly select a starting point from the neighborhoods of the current best solution by arbitrarily choosing k pairs of clusters (start with $k = 1$) and merging these k pairs of clusters into k single clusters. If there is no improvement in the VNS heuristic search, VNSOptClust generates a progressive diversification process, in which k is incremented while changing the neighborhoods, and a new perturbation step starts using a different neighborhood.

Local Search. The random solution generated from the procedure of Perturbation becomes the starting point of the local search. To address the issue of dependency on the initial partition status, VNSOptClust applies the Chebyshev's Theorem in the cluster-merging step within the local search routine. According to Chebyshev's Theorem, we observe that when $n = \sqrt{2}$, at least 50% of objects are within 1.414 standard deviations of the mean. Therefore, it can be assumed that, given two adjacent clusters, whose overlap is over the threshold $d = 1.414(\sigma_1 + \sigma_2)$ at least 50% of the data points from these two adjacent clusters are similar to each other. We can say these two adjacent clusters are close enough to be merged. The merging procedure within the local search routine is to create a compact, well-separated partition. After the merging procedure,

VNSOptClust can perform the assignment step to assign data objects to these refined clusters. At the end of the Local Search process, a new partition and new objective value are obtained. The new solution is compared with the current best one and a decision whether to replace the current incumbent solution with the new solution is made during the Move-or-Not stage.

4 Experimental Results

As an application to intrusion detection, VNSOptClust is tested on subsets of the KDD Cup 1999 dataset [17]. We have compared VNSOptClust with one automatic, local search based clustering method (Y-means)[6] [8] and one metaheuristic based clustering method (IDBGC)[7] [22]. The strategy for this comparison study is to create the same experimental environment as mentioned in [22]. Five datasets are exacted from the KDD Cup 1999 dataset. The statistical distribution of attack categories in each dataset is detailed in Fig. 1. Both Y-means and VNSOptClust are coded in Java, and tested on these five datasets. All experiments run on Dell-Intel (R) Pentium (R) M CPU 1.8GHz, 1.00GB of RAM.

To evaluate the performance of the clustering algorithms, we are interested in two indicators: the Detection Rate (DR) and the False Positive Rate (FPR). DR is defined as the number of intrusion instances detected by the algorithm divided by the total number of intrusion instances present in the dataset, whereas FPR equals the number of normal instances incorrectly classified by the algorithm as intrusion divided by the number of normal instances in the dataset [24].

The comparative results of Y-means, VNSOptClust, and IDBGC are displayed in Table 1. VNSOptClust has achieved an average detection rate of 71.2% with a low false positive rate of 0.9%. As noted, the average FPR of VNSOptClust is a bit higher than that of IDBGC, but within an tolerably low value according to the definition in [24]. Hence, we can conclude that VNSOptClust is effective in unsupervised anomaly detection.

The results in Table 2 suggest that under the condition of unsupervised anomaly detection, VNSOptClust is able to detect the majority of unknown attacks for each attack category. In particular, it can detect 84% of the DOS attacks. Therefore, VNSOptClust is effective in automatically detecting unknown intrusion attacks.

---

[6] Y-means is a good representative of automatic local search based clustering method. In [8], it has been applied for intrusion detection. Its performance has been compared with H-means+, an improved version of K-means. It also has a better intrusion detection rate than Portnoy's algorithm [24].

[7] In the literature, there are not many metaheuristic based clustering algorithms available for solving intrusion detection problems. The best solution of intrusion detection based metaheuristic algorithms is taken from [22].

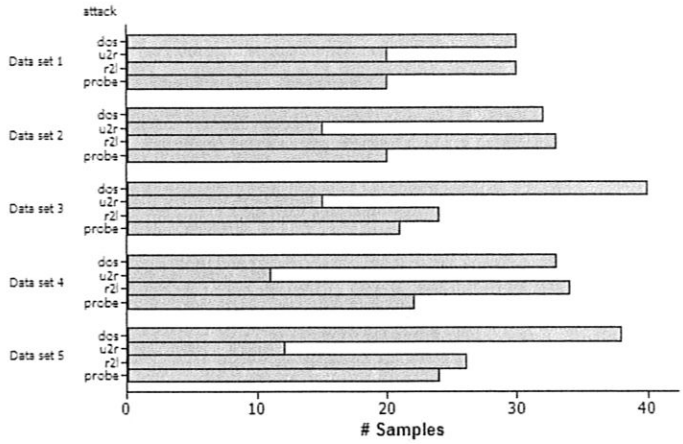

Fig. 1. Attack Distribution in Each Dataset

Table 1. Comparative Results of Y-means, VNSOptClust, and IDBGC

Data set (%)	Y-means DR	Y-means FPR	VNSOptClust DR	VNSOptClust FPR	IDBGC DR	IDBGC FPR
Dataset 1	64	2.0	63	0.4	68	0.8
Dataset 2	55	2.1	81	1.9	33	0.2
Dataset 3	56	1.7	79	0.6	74	0.4
Dataset 4	59	2.2	72	1.0	44	0.3
Dataset 5	52	1.8	64	0.2	79	0.4
Average	57.2	1.96	71.2	0.9	59.6	0.4

Table 2. Detection Percentage of Different Attack Categories

Data set (%)	DOS DR	U2R DR	R2L DR	PROBE DR
Dataset 1	80	70	55	33
Dataset 2	78	80	78	90
Dataset 3	95	67	67	71
Dataset 4	87	72	69	50
Dataset 5	80	50	73	36
Average	84	68	68	56

5 Conclusion

In this paper, we applied a VNS-based clustering algorithm, VNSOptClust, in solving the unsupervised anomaly detection problem. VNSOptClust adopts a VNS metaheuristic procedure to allow clustering solutions to get out of local optimality with a poor value; it considers the statistical nature of data distribution to find a near-optimal solution; it utilizes a cluster validity index as an objective function of the VNS heuristic search to obtain compact, well-separated clusters. Under the condition of unsupervised anomaly detection, VNSOptClust has obtained an average detection rate of 71.2% with an acceptably low false positive rate of 0.9%, and is capable of detecting the majority of unknown attacks for each attack category. Therefore, VNSOptClust is a promising clustering method for unsupervised anomaly detection.

Acknowledgement. This work was partially supported by NSERC discovery grants awarded to Dr. Nabil Belacel.

References

1. Al-Sultan, K.S.: A Tabu Search Approach to the Clustering Problem. Pattern Recognition. 28(9), 1443–1451 (1995).
2. Babu G.P., and Hall, D.: A near-optimal initial seed value selection in K-means algorithm using a genetic algorithm. Pattern Recognition Letters. 14, 763–769 (1993).
3. Belacel, N., Hansen, P., and Mladenovic, N.: Fuzzy J-means: a new heuristic for fuzzy clustering. Pattern Recognition. 35(10), 2193–2200 (2002).
4. Bloedorn, E., Christiansen, A.D., Hill, W., Skorupka, C., Talbot, L.M., and Tivel, J.: Data Mining for Network Intrusion Detection: How to Get Started. MITRE Technical Paper. (2001).
5. Dokas, P., Ertoz, L., Kumar, V., Lazarevic, A., Srivastava, J., and Tan, P.N. : Data Mining for Network Intrusion Detection. In: Proceedings of the NSF Workshop on Next Generation Data Mining, Baltimore, Maryland (2002).
6. Brimberg, J., and Mladenovic, N., A variable neighborhood algorithm for solving the continuous location-allocation problem. Studies in Location Analysis. 10, 1–12 (1996).
7. Brimberg, J., Hansen, P., Mladenovic, N., and Tailard, E.: Improvements and Comparison of heuristics for solving the multisource weber problem. Operations Research. 3, 444-460 (2000).
8. Guan, Y., Ghorbani, A., and Belacel, N.: Y-means: a clustering method for intrusion detection. In: Proceedings of 2003 IEEE Canadian Conference on Electrical and Computer Enginerring, pp. 1083–1086, Montreal (2003).
9. Hall, L.O., Ozyurt, I.B., and Bezdeck, J.C.: Clustering with a Genetically Optimized Approach. IEEE Transaction on Evolutionary Computation. 3(2), 103–112 (1999).
10. Hansen, P., Jaumard, B., Mladenovic, N., and Parreira, A. Variable Neighborhood Search for the - median Location Science. 5(4), 207–226 (1998).
11. Hansen, P., and Mladenovic, N.: J-means: a new local search heuristic for minimum sum of squares clustering. Pattern Recognition. 34, 405–413 (2001).

12. Hansen, P., Mladenovic, N.: Variable Neighborhood Search. In Glover, F., and Kochenberger, G.A. (eds.) Handbook of Metaheuristics, pp. 145–184. Kluwer Academic Publishers, Boston (2003).
13. Hansen, P., and Mladenovic, N.: Variable Neighborhood Search: Principle and Applications. European Journal of Operational Research. 34, 405–413 (2001).
14. Hansen, P., and Mladenovic, N.: An Introduction to Variable Neighborhood Search. In Vo, S., Martello, S., Osman, I., and Roucairol, C. (eds) Metaheuristics: Advances and trends in local search paradigms for optimization, pp. 433-458, Kluwer Academic Publishers (1999).
15. Jain, A.K., and Dubes, R.C.: Algorithms for Clustering Data. Prentice-Hall, New Jersey (1988).
16. Julisch, K.: Data Mining for Intrusion Detection: A Critical Review. In Barbara, D., and Jajodia, S. (eds) Applications of data mining in computer security, pp. 1–14. Kluwer Academic Publisher, Boston (2002).
17. KDD Cup 1999 Data, University of California, Irvine, October, 1999. `http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html`.
18. Krishna, K., and Murty, M.: Genetic K-means Algorithm. IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, 29(3), 433–439 (1999).
19. Lee, W., Stolfo, S.J., Chan, P.K., Eskin, E., Fan, W., Miller, M., Hershkop, S., and Zhang, J.: Real time data mining based intrusion detection. In: Proceedings of DARPA Information Survivability Conference & Exposition II (2001).
20. Lee, W., Stolfo, S.J.: Data Mining Approaches for Intrusion Detection. In: Proceedings of the 7th USEUIX Security Symposium, San Antonio, Texas (1998).
21. Lippmann, R.P., Graf, I., and et al.: The 1998 DARPA/AFRL Off-Line Intrusion Detection Evaluation. In: First International Workshop on Recent Advances in Intrusion Detection (RAID), Louvain-la-Neuve, Belgium (1998).
22. Liu, Y., Chen, X., Liao, X., and Zhang, W.: A genetic clustering method for intrusion detection. Pattern Recognition. 37, 927–942 (2004).
23. Mirkin, B.: Clustering for Data Mining: A Data Discovery Approach. CRC Press, Boca Raton, Florida (2005).
24. Portnoy, L., Eskin, E., and Stolfo, S.J.: Intrusion Detection with unlabeled data using clustering. In: Proceedings of ACM CSS Workshop on Data Mining Applied to Security (DMSA-2001), Philadelphia, PA (2001).
25. Rousseeuw, P.J.: Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20, 53–65 (1987).
26. Selim, S.Z., and Alsultan, K.: A Simulated Annealing Algorithm for the Clustering Problem. Pattern Recognition. 24(10), 1003–1008 (1991).
27. Spath, H.: Cluster Analysis Algorithms. Ellis Horward, Chichester, UK (1980).
28. Tan, P., Steinbach, M., and Kumar, V.: Introduction to Data Mining, pp.487-647. Addison-Wesley, Boston, MA (2006).
29. Tseng, L.Y., and Yang, S.B.: A genetic approach to the automatic clustering problem. Pattern Recognition. 34, 415-424 (2001).
30. Walpole, R.E.: Elementary Statistical Concepts, 2nd edition, Macmillan Publishing Co., New York (1983).
31. Wang C.: In Search of Optimal Clusters Using Variable Neighbourhood Search, Master Thesis, Department of Computer Science, University of New Brunswick, Fredericton, Canada (2007).
32. Xu, R., and Wunsch, D.: Survey of Clustering Algorithms, IEEE Transactions on Neural Networks. 16(3), 645–678 (2005).