

改进的聚类算法在网络异常行为检测中的应用

辛 壮¹ 万 良¹ 李均涛²

(1. 贵州大学 计算机科学与技术学院,贵州 贵阳 550025;

2. 贵州财经大学 信息学院,贵州 贵阳 550025)

摘要: 网络异常行为检测是对大规模网络数据流量进行分析并发现入侵行为的一种方法。针对基于聚类的网络异常行为检测方法不能及时准确地选择初始聚类中心、无法有效地识别非球状簇等问题,提出一种改进的聚类算法应用在网络异常行为检测中。该方法使用最小生成树算法获得初始聚类中心,使用改进的K-means聚类算法区分异常行为与正常行为,通过距离比值判断聚类效果,提高了聚类效果的准确性。通过应用有监督学习的方式对聚类结果进行检测,结果表明改进的聚类算法能够更好地识别初始聚类中心,并进行更加有效的聚类,能够更加准确地检测出网络异常行为。

关键词: K-means; 最小生成树; 网络异常行为; 聚类; 数据挖掘

中图分类号: TP39

文献标识码: A

文章编号: 1673-629X(2019)03-0111-06

doi: 10.3969/j.issn.1673-629X.2019.03.024

Application of Improved Clustering Algorithm in Network Abnormal Behavior Detection

XIN Zhuang¹, WAN Liang¹, LI Jun-tao²

(1. School of Computer Science and Technology, Guizhou University, Guiyang 550025, China;

2. School of Information, Guizhou University of Finance and Economics, Guiyang 550025, China)

Abstract: Network abnormal behavior detection is a method to analyze and discover the intrusion behavior of large-scale network data flow. The anomalous behavior detection method based on clustering cannot timely and correctly select the initial clustering center and is unable to effectively identify the globular clusters. In order to solve these problems, we propose an improved clustering algorithm in the network abnormal behavior detection. This method obtains the initial clustering center by using the minimum spanning tree algorithm, distinguishing the abnormal behavior from the normal behavior by the improved K-means clustering algorithm and judging the clustering effect by the distance ratio, which improves the accuracy of the clustering effect. The results tested by supervised learning show that the improved clustering algorithm can better identify the initial clustering center and make more effective clustering and detect the network anomaly behavior more accurately.

Key words: K-means; minimum spanning tree; network anomalous behavior; clustering; data mining

1 概述

现阶段,计算机网络已经基本实现全球化,庞大的网络体系方便了信息的交流与传递,但是也给网络黑客提供了更多的便利。面对着越来越复杂的网络环境、灵活多变的网络攻击手段,如何快速、准确地识别网络异常行为,并减轻异常行为对网络环境的破坏具有非常重要的意义。对此,专家学者们将数据挖掘技术应用到检测技术中,依靠无监督、半监督的方式快速

准确地识别异常、非异常行为,这也是当前的研究热点之一^[1-2]。

K-means作为传统的聚类算法,能够对大量的数据进行快速聚类,较好地符合了网络异常行为检测的实时性要求。K-means算法能够对无标记的数据样本进行有效聚类,相比于有监督的方法能够大大减少标记时间并且降低数据开销。国内外不断有学者将聚类算法与分类算法相结合的半监督学习方式应用于异

收稿日期: 2018-04-18

修回日期: 2018-08-23

网络出版时间: 2018-12-20

基金项目: 贵州省研究生卓越人才计划项目(黔教研合ZYRCZ字[2014]010号);贵州省科学基金(黔科合J字[2011](2328)黔科合LH字[2014](7634))

作者简介: 辛 壮(1994-),男,研究生,CCF会员(72411G),研究方向为信息安全、入侵检测;万 良,博士,教授,研究方向为信息安全、协议形式化分析;李均涛,博士,副教授,研究方向为密码学与信息安全、协议形式化分析和模型检测。

网络出版地址: <http://kns.cnki.net/kcms/detail/61.1450.TP.20181219.1542.064.html>

常行为检测,例如 Zhang 等^[3]分析了半监督学习方式对于聚类效果的影响,Erman 等^[4]将半监督学习方式应用到流量分类中,并且验证了半监督学习方式对于网络流量分类的有效性。

传统的 K-means 算法虽然简单有效,但需要事先确定 k 值,并且算法对于噪声敏感。为了克服 K-means 算法的不足,许多专家学者都曾试图从各个角度对 K-means 算法进行改进与优化。文献[5]提出了一种快速的最小生成树算法,该算法与 K-means 算法结合使用能够快速处理大规模数据集。把精确的最小生成树算法应用到通过 K-means 算法得到的 K 个聚类之中,根据提出的准则连接每个处理之后的聚类,形成近似的最小生成树。再将聚类产生的相邻对的相邻边界划分为一个聚类,并构建另一个近似的最小生成树。通过合并操作,将两个近似最小生成树合并成一个图形,从而生成更精确的最小生成树,以提高算法的运行效率,使结果更加有效。文献[6]提出了一种能够快速对多维数据进行 K-means 聚类的方法。通过将多维数据分组,不同的组对应不同的带权值的非空单元格,K-means 聚类对象不再作用于单个的多维对象而是直接对非空单元格的哑元点进行操作,并且在算法的迭代过程中不再重复计算哑元点与不同聚类中心之间的聚类。这种改进能够对大量的多维数据进行快速有效的聚类。文献[7]提出一种新的聚类算法,将最小生成树算法与 K-means 算法结合使用,通过基于质心的最近邻规则来划分局部邻域图以代替传统的构造完全图的方法。该算法能够在保证簇质量的前提下降低计算时间。文献[8]提出一种异常检测技术,通过 K-means 算法将数据集聚类并构建最小生成树,将树中的最长边删除形成新的聚类,将小聚类中的数据点作为异常值的候选点并进行分析,从而找出真正的异常点。文献[9]提出一种无参数最小生成树算法来自动地确定簇数,通过计算簇间与簇内距离的比例,对 K-means 聚类结果进行测试。结果表明,改进算法有更好的聚类效果。文献[10]提出使用模糊聚类算法,将聚类边缘的网络数据特征值再次模糊聚类的异常行为判断方法。

上述改进虽然能够在聚类前事先确定 k 值,也能减少噪声点对聚类效果的影响,但是对于准确选择初始聚类中心和有效识别非球状簇这两个问题并没有得到改善。对此,文中提出一种改进的聚类算法应用在网络异常行为检测中,该算法的优越性表现在:融合最小生成树算法,能够更佳有效地判断聚类中心,避免了传统 K-means 算法依靠随机生成聚类中心从而造成聚类结果不准确的情况;使用重心算法更新聚类中心,能够更好地识别非球状簇;通过距离比值在聚类的迭

代过程中判断聚类效果的优劣,使得聚类结果始终保持在最优状态。

2 传统 K-means 算法

K-means 算法是经典的聚类算法,应用在异常行为检测中能很好地对训练数据集进行训练从而构建分类器。检测模块根据分类器来判断是否发生异常行为。算法应用如图 1 所示。

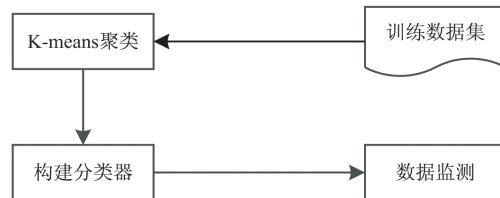


图 1 聚类算法在网络异常行为检测中的应用模型

算法能够根据预先给定的 k 值,随机地在样本集中选取 k 个聚类中心,通过计算欧氏距离与预先设置好的阈值进行对比,把相似度高的样本分配到同一个簇中。之后不断的进行迭代,更新聚类中心,直到目标函数收敛,最终把样本集聚类成 k 个类^[11]。算法步骤如下:

设聚类样本集为 $X = \{x_1, x_2, \dots, x_n\}$,其中 x 为样本数据, n 为样本总数;聚类类别为 $Z_k = \{z_1, z_2, \dots, z_k\}$,其中 k 为类别数目, n_k 表示第 k 类的样本数目; $\{c_1, c_2, \dots, c_k\}$ 为 k 个聚类中心。

定义样本数据平均值:

$$z = \frac{1}{n_k} \sum_{x \in Z_k} x \quad (1)$$

定义目标函数:

$$J = \sum_{i=1}^{n_k} \sum_{j=1}^k d_{ij}(x_i, c_j) \quad (2)$$

目标函数 J 为样本数据集每个类中所有点到聚类中心的距离平方和,当 J 最小即得到最小均方差时,迭代完成。

- (1) 预先指定聚类个数 k ;
- (2) 随机选取 k 个样本数据 c_1, c_2, \dots, c_k 作为初始聚类中心;
- (3) 根据公式 $d(x_i, c_j)$ 计算每个点与聚类中心的距离,根据欧氏距离对样本数据进行聚类,每个样本数据总是聚类到与包含样本点最近的聚类中心的簇中;
- (4) 根据式 1 重新计算聚类中心 c_1, c_2, \dots, c_k ;
- (5) 若式 2 收敛,则聚类完成,若不收敛,则重复执行步骤 3~4。

K-means 算法作为传统的聚类算法,能够应付大规模的数据处理,并且聚类速度快,但当 K-means 算法应用在网络异常行为检测领域时也有一定的缺点:

- (1) 要人为确定 k 值,这需要进行大量的实验和具

备一定的学术经验才能对 k 值进行准确判断,不同的 k 值对聚类结果影响巨大,一个不好的 k 值往往会降低聚类效果;

(2) K-means 算法中初始聚类中心的确定是随机的;

(3) 对离群值与噪声数据敏感;

(4) 对非球状簇的聚类效果不明显;

(5) 容易陷入局部最优解^[12-13]。

K-means 聚类算法也有要遵循的前提条件,这也是所有聚类算法要遵循的几点假设。

(1) 在聚类过程中,正常网络数据流量的数目要占多数,远远大于非正常网络数据流量的数目;

(2) 正常网络数据流量要与非正常网络数据流量特征有明显的差异。

3 改进的聚类算法

根据对 K-means 聚类算法的大量研究,以 K-means 聚类算法为基础,添加最小生成树概念,在已经确定好 k 值的情况下,通过融合最小生成树算法,将训练数据集进行合并和分裂操作来确定训练样本关键点。把关键点映射成训练数据集的初始聚类中心,在数据迭代过程中,通过重心算法重新计算聚类中心,通过对距离比值聚类结果的优劣,直到收敛函数收敛,得到最终的聚类结果^[14-17]。

3.1 相关概念

(1) 重心。

传统的 K-means 算法由于算法限制及依据数据点距离平均值选取聚类中心的方法,使其对非球状簇的识别效果并不理想。通过计算聚类重心来更新聚类中心的方法,能够使得聚类中心不会偏离聚类本身,使得算法在识别非球状簇时也能有很好的聚类效果。公式如下:

$$g = (w_1 + w_2 + \cdots + w_n) / n \quad (3)$$

(2) 类内距离。

设 $c_j (j=1, 2, \dots, k)$ 为聚类中心, x_i 是以 c_j 为中心的聚类中的任意数据。类内距离 DWC 为类内任意数据点到聚类中心距离的平均值,即:

$$DWC = \sum_{i=1}^{n_j} \frac{1}{n_j} d(c_j, x_i) \quad (4)$$

其中, n_j 为某一类中的数据个数。

(3) 类间距离。

设 $c_j (j=1, 2, \dots, k)$ 为聚类中心, x_i' 为不以 c_j 为聚类中心的其他类中的数据点,则类间距离 DBC 为某一聚类中心到其他聚类中数据点的距离的平均值,即:

$$DBC = \sum_{j=1}^k \sum_{i=1}^{n_j} \frac{1}{(k-1)n_j} d(c_j, x_i') \quad (5)$$

(4) 距离比值。

距离比值 WB 是类内距离 DWC 与类间距离 DBC 的比值,公式为:

$$WB = \frac{DWC}{DBC} = \frac{\sum_{i=1}^{n_j} \frac{1}{n_j} d(c_j, x_i)}{\sum_{j=1}^k \sum_{i=1}^{n_j} \frac{1}{(k-1)n_j} d(c_j, x_i')} \quad (6)$$

通过 WB 来反映聚类效果的优劣。K-means 算法根据类间分离、类内紧凑的原则进行聚类划分,通过单个数据点进行研究分析,分别用类间距离 DBC 与类内距离 DWC 表示类间分离度、类内紧凑度,计算出的结果能够更加直观准确地对聚类效果进行判断。如图 2 所示,网络数据流量样本被分为四类,分别为 j 、 x 、 y 、 z , j 类中的样本中心点为 i 。根据概念 2 得知,样本中心点 i 到 j 类内其他样本数据距离的平均值为类内距离,反映了类内的紧凑程度,从数值上看,得到的 DWC 数值越小,类内紧凑程度就越大。根据概念 3, j 类样本中心点 i 到其他类 x 、 y 、 z 的样本中心点距离的平均值称为类间距离,反映了类间的离散程度,类间距离越大,DBC 越大,离散程度越高。从类内紧凑程度来看,希望 DWC 越小越好,从类间离散程度来看,希望 DBC 越大越好。因此单一的 DBC 或者 DWC 都无法准确地判断聚类效果的优劣,于是将 WD 作为聚类结果的评价标准。WD 通过类间距离与类内距离的比值作为评判标准,WD 越小说明聚类效果越好,但是并不适用于聚类数为 1 的情况。

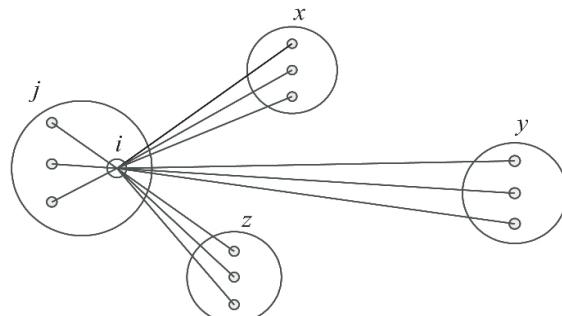


图 2 聚类结构示意

3.2 算法思想

设聚类样本集为 $X = \{x_1, x_2, \dots, x_n\}$, 其中 x 为样本数据, n 为样本总数。设数据集 $E = \{e_1, e_2, \dots, e_n\}$ 为最小生成树边集, 边长度由权重表示, 由欧氏距离公式 $d(x, y) = \sqrt{|x_1 - y_1|^2 + |x_2 - y_2|^2 + \cdots + |x_n - y_n|^2}$ 计算获得。设聚类中心集 $C = \{o_1, o_2, \dots, o_n\}$ 。

(1) 初始聚类中心的选取。

最小生成树是完全图的最小连通子图,包含完全图的所有点。将网络流量数据作为完全图的连接点构建带权完全图 $G(x) = \{V, E, D\}$, 其中 $V (V \in X)$ 为完

全图的顶点集, $E = \{(x_i, x_j) | x_i, x_j \in X\}$ 为完全图的边, 将任意两个连接点 x_i, x_j 之间的距离 $d(x_i, x_j)$ 作为边的权重 D ($D = \{d(x_i, x_j) | x_i, x_j \in X\}$)。通过 Kruskal 算法构建出完全图的最小生成树, 通过最小生成树确定网络数据流量关键点, 其具体过程如下:

首先, 确定网络流量数据关键点的所有候选点。计算完全图中所有边的权值 D , 寻找权值最小边的点, 如 x_i, x_j 是权值最小边的点, 将两点存入集合 B_x , 将边存入集合 E_x 。搜索完全图中所有的边, 此时集合 B_x 与集合 E_x 包含全部的数据点与边, 并且边集 E_x 中的数据是以边权重的大小进行排列的。

遍历边集 E_x , 将最小权值的边添加到最小生成树中。若添加一条边之后, 最小生成树构成回路则删除边集 E_x 、点集 B_x 中对应的数据, 最终构建出最小生成树 $T(E_1, E_2, \dots, E_n)$ 。根据最小生成树, 逐层查找距离最小的两个点, 计算两个点的中心 $O = d(x_i, x_j)/2$ 。用计算出的中心点 O 代替两个连接点的边, 并且将中心点作为父节点处理, 更新边集 E_x 与点集 B_x 。此时两个集合中的数据都将减少 1, 重复执行合并操作, 直到最小生成树只含有 k 个连接点时, 查找结束。将最小生成树剩余的连接点作为网络流量数据的初始聚类中心。

(2) 初始聚类的划分。

将最小生成树算法得到的数据点作为初始聚类中心, 得到聚类中心集 $C = \{o_1, o_2, \dots, o_n\}$ 。取其中一个聚类中心点 o_x 为例, 遍历新数据样本集 X ($X \in \{X - o_x\}$)。根据预先设置的半径 R , 若样本数据集中的数据元素与聚类中心点 o_x 的距离小于半径 R , 则划分到以 o_x 为聚类中心的簇中, 并修改类标识, 在新的样本数据集中删除此样本数据。若距离大于半径 R , 则将此样本数据点与其他聚类中心点进行比较。

(3) 生成新的聚类。

根据式 3 计算新的聚类中心, 得到新的聚类中心集 C 。与初始聚类划分的操作类似, 根据半径 R 进行新聚类的划分。新聚类划分完成之后, 计算每个点的 DWC 与 DBC, 得到距离比值 WB。若比值过大或超过阈值, 则需要重新聚类。

具体的算法实现如下:

步骤 1: 根据公式 $d(x_i, x_j)$ 计算出任意样本数据 x_i ($i = 1, 2, \dots, n$) 与其他样本数据 $x \in X \setminus \{x_i\}$ 的欧氏距离 d , 获得数据集 D , 用来存储样本的数据距离。

步骤 2: 根据 Kruskal 算法, 把样本数据集 D 中的数据作为最小生成树的端点, 将两点之间的距离作为最小生成树的边, 从而构建最小生成树, 将边长存入数据集 E 。从数据集 E 中寻找权重最小的边即欧氏距离最小的两个点, 公式如下:

$$D_{\min} = \min\{d(x_i, x_j)\} \quad (7)$$

其中, x_i, x_j 为训练集中的任意数据, 若其权重最小, 则计算出中心点 d_{ij} , 用计算出的值代替权重最小的边, 更新数据集 E , 直到数据集 E 中的元素个数剩余 k 个。

步骤 3: 以数据集 E 中的元素作为关键点, 映射到样本数据集合 X 中, 以关键点作为初始聚类中心, 获得聚类中心集合 C 。

步骤 4: 以集合 C 中的元素 o_x 为中心, 遍历训练集的样本数据, 根据距离最近的原则将数据点划分成 k 个簇。

步骤 5: 根据式 3 重新计算出新的聚类中心, 根据新的聚类中心更新聚类中心集合 C , 根据数据集中点的相似度进行重新聚类。

步骤 6: 根据式 6 计算出样本的 WB, 与预先设置好的阈值进行比较, 若不满足条件则重新执行步骤 4, 重新聚类, 若满足则执行步骤 7。

步骤 7: 若目标函数(式 2) 收敛, 则输出最终聚类结果, 若不收敛, 则重新聚类。

在聚类过程中会遇到边缘值的问题, 样本数据点会在类的交界处, 这种数据至少拥有几个类的属性特征, 没办法明确地确定这种数据点到底属于哪一个类。通过计算余弦相似度来进行区分, 计算公式如下:

$$\cos(x, y) = \frac{\sum_{k=1}^n x_k y_k}{\sqrt{\sum_{k=1}^n x_k^2} \sqrt{\sum_{k=1}^n y_k^2}} \quad (8)$$

其中, n 表示样本数据维度; x 表示在类交界处的数据点; y 表示处于交界类中的任意数据。改进的 K-means 算法将处于类交界处的数据点划分到余弦相似度最大的类中。

4 实验

采用 KDD Cup99 数据集进行仿真实验, 该数据集是由哥伦比亚大学教授和北卡罗来纳州立大学教授通过数据挖掘技术对 DARPA 数据进行处理形成的。数据集中包括 Normal、DOS、Probe、R2L、U2R 五大类攻击类型的数据, 并且又细分为 22 种攻击行为数据, 其中训练集仅存在 8 种攻击行为数据, 另外 14 种攻击行为数据存在于测试集中。

4.1 数据预处理

数据预处理的目的是使数据能够更加快速有效地进行数据挖掘。首先把离散属性转换为连续属性, 然后再进行数据点标准化与归一化。计算公式如下:

$$X_{ij} = \frac{X_{ij} - \text{AVG}_j}{\text{STAD}_j} \quad (9)$$

$$\text{AVG}_j = \frac{1}{n} (X_{1j} + X_{2j} + \dots + X_{nj}) \quad (10)$$

$$\text{STAD}_j = \frac{1}{n} (|X_{1j} - \text{AVG}_j| + |X_{2j} - \text{AVG}_j| + \dots + |X_{nj} - \text{AVG}_j|) \quad (11)$$

其中, X'_{ij} 为 X_{ij} 标准化后的数值; AVG_j 为数据的平均值; STAD_j 为数据的平均绝对偏差。

数据归一化是把标准化的数据归一到 [0-1] 区间之内, 归一化的公式如下:

$$X''_{ij} = \frac{X'_{ij} - X_{\min}}{X_{\max} - X_{\min}} \quad (12)$$

$$X_{\min} = \min\{X'_{ij}\} \quad (13)$$

$$X_{\max} = \max\{X'_{ij}\} \quad (14)$$

其中, X''_{ij} 为 X'_{ij} 归一化之后的值; X_{\min} 、 X_{\max} 分别为标准化数据中的最小值与最大值。

从 KDD Cup99 数据集中按比例抽取一部分数据作为训练集与测试集, 把训练集和测试集分为四组进行测试。其中每组正常行为数据为 4 890 条, 非正常行为数据为 110 条, 如表 1 所示。

表 1 异常行为检测测试数据集

测试集	记录数	正常	入侵	分类
		记录数	记录数	攻击数
Test1	5 000	4 890	110	4
Test2	5 000	4 890	110	3
Test3	5 000	4 890	110	4
Test4	5 000	4 890	110	2

4.2 实验结果与分析

实验使用检测率与误检率作为网络异常行为检测的评判标准, 公式如下:

$$\text{检测率} = \frac{\text{检测的异常行为数据条数}}{\text{异常行为总条数}} \quad (15)$$

$$\text{误检率} =$$

$$\frac{\text{被检测为异常行为的正常行为的数据条数}}{\text{正常行为总条数}} \quad (16)$$

经过多次实验, 得出能实现较好聚类结果的 k 值, 首先对传统的 K-means 算法与融合高斯随机数的 K-means 算法进行比较, 结果如表 2 所示。

表 2 算法检测结果比较

测试集	传统的 K-means 算法		融合高斯随机数的 K-means 算法	
	检测率/%	误检率/%	检测率/%	误检率/%
	Test1	72.7	9.6	91.5
Test2	79.3	9.1	83.0	6.7
Test3	72.4	9.4	91.0	6.0
Test4	71.7	8.9	93.0	5.4
均值	74.0	9.3	88.7	6.1

使用融合最小生成树的 K-means 算法对数据集进行检测, 检测结果如表 3 所示。

表 3 融合最小生成树的 K-means 算法检测结果

测试集	检测率/%	误检率/%
Test1	95.9	0.13
Test2	98.2	0.07
Test3	91.3	1.80
Test4	95.4	0.11
均值	95.2	0.53

由对比结果可以看出, 对于网络异常行为检测, 传统的 K-means 算法误检率较高, 容易产生误报, 将正常的网络行为检测成异常网络行为, 误检率平均值为 9.3%。检测率在三种方法中最低, 平均值为 74.0%, 对于异常网络行为, 容易产生漏报。而融合高斯随机数的 K-means 算法对于网络异常行为检测, 检测率相对于传统 K-means 有所提高, 但是检测率波动较大, 具有随机性。

由表 3 可以看出, 融合最小生成树的 K-means 算法, 在网络异常行为检测方面有较好的效果。与传统 K-means 算法、融合高斯随机数的 K-means 算法相比, 检测率有明显的提升, 均值提升到 95.2%, 误检率平均值下降到 1% 以下。

为了更好地检验融合最小生成树的 K-means 算法对于网络异常行为的检测效果, 将其与传统 K-means 算法进行比较, 检测效果如表 4 所示。

表 4 对不同攻击类型检测效果

攻击类型	检测率/%		误检率/%	
	传统的 K-means	改进的 K-means	传统的 K-means	改进的 K-means
U2R	0	95	100	3.2
R2L	80	98	23.7	0.7
DOS	83	97	20.1	2.8
Probe	90	100	26.3	0

传统的 K-means 算法对于 U2R、R2L 的检测效果不是太理想, 这两种攻击类型都是伪装成用户正常行为来实现攻击的, 数据特征与正常行为特征具有相似性, 并且两种攻击类型数据数目较少, 在数据训练时对其处理效果不好, 导致后期检测效果下降。另一方面, 改进后的 K-means 算法对于四种攻击都有很好的检测效果, 尤其是 Probe 类型攻击, 达到了完全检测的效果。对于 U2R 类型攻击, 不可避免地将其聚类到正常网络行为中, 使得其误检率较高。

从实验结果可以看出, 融合最小生成树的 K-means 算法对网络异常行为具有更高的检测效果, 由最小生成树得到数据关键点, 以关键点为初始聚类中

心。相对于传统的 K-means 算法,该算法能够较好地解决初始聚类中心随机选择的问题,克服了聚类容易陷入局部最优的问题。自定义的有效性指标能够判断聚类效果的好坏,确保最佳聚类效果。通过重心选择算法更新聚类中心的方法,使得聚类不再因为聚类簇图像过于狭窄或者非球形而使得聚类中心偏离簇本身,与传统的 K-means 算法相比,检测效果明显提升。

5 结束语

通过仿真实验可以看出,融合最小生成树算法能够使得 K-means 算法在网络异常行为检测中得到更好的应用,并且改进的重心选择算法能够使得 K-means 算法的聚类效果更加准确有效。该算法在 k 值确定的情况下,通过最小生成树算法确定初始聚类中心,克服了初始聚类中心随机性选择的问题,距离比值的存在确保了输出的聚类能够达到最好的效果,在更新聚类中心的过程中,通过重心选择算法选取新的聚类中心,这有别于传统的计算平均值算法,有效解决了聚类中心因为聚类簇太狭窄而使得聚类中心偏移的问题。

然而融合最小生成树的 K-means 算法,时间复杂度太高,要迭代数次才能完成数据关键点的确认。并且该算法仍然要事先确定 k 值,还是不能摆脱算法对 k 值的依赖。在今后的工作中仍要对该算法进行进一步优化,在确保检测率提升的同时,降低误检率与时间复杂度。

参考文献:

- [1] 李 乔,何 慧,方滨兴,等. 基于信任的网络群体异常行为发现 [J]. 计算机学报,2014,37(1):1–14.
- [2] 陆 悠,李 伟,罗军舟,等. 一种基于选择性协同学习的网络用户异常行为检测方法 [J]. 计算机学报,2014,37(1):28–40.
- [3] ZHANG Jinyuan, YANG Yan, WANG Hongjun, et al. Semi-supervised clustering ensemble based on collaborative training [C]//Proceedings of the 7th international conference on rough sets and knowledge technology. Chengdu, China [s. n.] 2012: 450–455.
- [4] ERMAN J, MAHANTI A, ARLITT M, et al. Offline/real-time traffic classification using semi-supervised learning [J]. Performance Evaluation, 2007, 64(9–12): 1194–1213.
- [5] ZHONG Caiming, MALINEN M, MIAO Duoqian, et al. A fast minimum spanning tree algorithm based on K-means [J]. Information Sciences, 2015, 295: 1–17.
- [6] 黄震华,向 阳,张 波,等. 一种进行 K-Means 聚类的有效方法 [J]. 模式识别与人工智能,2010,23(4):516–521.
- [7] RAMASAMY J, MOHANTY S K, OJHA A. Fast minimum spanning tree based clustering algorithms on local neighborhood graph [M]//Graph-based representations in pattern recognition. [s. l.]: Springer International Publishing, 2015: 292–301.
- [8] WANG Xiaochun, WANG Xiali, WILKES D M. A minimum spanning tree – inspired clustering – based outlier detection technique [C]//Proceedings of the 12th industrial conference on advances in data mining: applications and theoretical aspects. Berlin, Germany: Springer-Verlag, 2012: 209–223.
- [9] RAJU B H V S R, KUMARI V V. Parameter-free minimum spanning tree (PFMST) based clustering algorithm [J]. Communications in Computer & Information Science, 2011, 203: 552–560.
- [10] 刘 琴. 网络模糊入侵规则下的异常行为判断 [J]. 网络空间安全,2017,8(6–7):51–54.
- [11] 贾洪杰,丁世飞,史忠植. 求解大规模谱聚类的近似加权核 k-means 算法 [J]. 软件学报,2015,26(11):2836–2846.
- [12] 王守强,朱大铭. 基于最小聚类求解 k-means 问题算法 [J]. 通信学报,2010,31(7):46–52.
- [13] 雷小锋,谢昆青,林 帆,等. 一种基于 K-Means 局部最优性的高效聚类算法 [J]. 软件学报,2008,19(7):1683–1692.
- [14] JANA P K, NAIK A. An efficient minimum spanning tree based clustering algorithm [C]//Proceeding of international conference on methods and models in computer science. Delhi, India: IEEE, 2009: 1–5.
- [15] ZHONG Caiming, MALINEN M, MIAO Duoqian, et al. Fast approximate minimum spanning tree algorithm based on K-Means [C]//International conference on computer analysis of images and patterns. Berlin: Springer, 2013: 262–269.
- [16] GRYGORASH O, ZHOU Yan, JORGENSEN Z. Minimum spanning tree based clustering algorithms [C]//IEEE international conference on tools with artificial intelligence. Arlington, VA, USA: IEEE, 2008: 73–81.
- [17] ZHONG Caiming, MIAO Duoqian, FRÄNTI P. Minimum spanning tree based split-and-merge: a hierarchical clustering method [J]. Information Sciences, 2011, 181(16): 3397–3410.