Locating Data Flow Anomaly With Statistical Distance

Ruoyu Wang

***Abstract*—[abstract]**

I. Introduction

Big data industry has been blooming within this decade, reshaping the form of life and work across the globe. According to [17], 1.2ZB data had been produced from 2012 to 2014. And the amount doubles every two years. Currently, there are totally 2.7ZB data in the digital universe [1]. Big data analysis has been widely adopted in scientific experiments [20], electric business [2, 29, 9], healthcare [11], governments [16] and many other fields.

However, it will be harder in the future to harness the exploding volumes of data since problems have already appeared in data management and engineering, threatening trustworthiness and reliability of data flows inside working systems. Data error rate in enterprises is approximately 1% to 5%, and for some, even above 30% [23].

Those data anomalies may arise due to both internal and external reasons with respect to a certain system. From one hand, components inside the system may generate problematic source data. For example, in a sensor network, some sensors may generate erroneous data when it experiences power failure or other extreme conditions [22]. Data packages will be lost if sensor nodes fail to connect to network or some sensor hub goes down [12]. Also, human operators act as a heavily vulnerable part to bugs and mistakes. Some even deliberately modify system configurations for malicious compromises [24]. A study [31] found that 65% of organizations state that human errors are the main cause of data problems.

On the other hand, data manipulation [14] from outside hackers composes another potential threat of data quality and reliability. Taken Apache Hadoop as an example. It's security issues has long been discussed within communities and industry [26, 30, 15, 27]. As is shown in Figure 1, the two basic vulnerabilities: *lack of access control* and the *absence of encryption* expose the entire cluster to dangerous threats. Data flows can be intercepted and modified; services can also be altered and blocked [13]. Although there are several frameworks(e.g. Kerberos, Sentry, Knox etc.) and algorithms providing basic protection [39, 28, 33, 37, 10], clever attackers can always bypass the barriers and sneak into the core of data pipelines. Several approaches are developed as sentinels to detect probable infiltrations. However, these approaches are not able to locate corrupted data under carefully planned manipulations. Nor can they figure out the exact reasons and recover the original records. To locate and diagnose anomalies

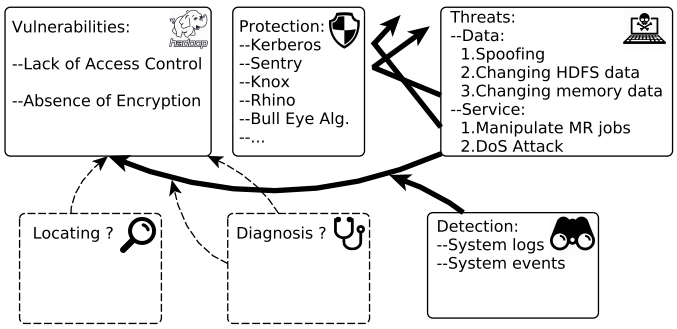

Fig. 1. Security Issues of Hadoop Clusters

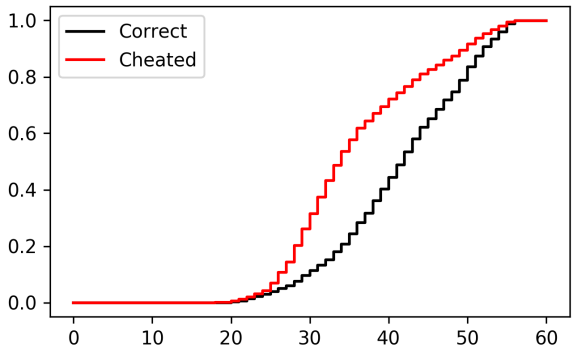

Fig. 2. Example Cumulative Distribution Function of Original And Modified Data

in data pipelines under carefully planned and disguised data manipulations are still on demand by industry and academia.

According to our observation, typical data manipulations on numerical data will lead to the drift of its distribution. For example, a Taobao online seller's transaction records can be "click farmed" to increase the volume of sales. Fig. 2 shows the sales distribution within one day. The curve for cheated data is emulated according to a popular method for click farming. It can be observed that there exists clearly a gap between these two distributions. These anomalies inside data pipelines will severely affect mining and learning algorithms and further change the final decision given by the entire system.

In order to address issues resulted from data manipulation and pipeline errors, we propose a novel mechanism to detect and locate corrupted data within a data pipeline via statistical

distance. As far as we know, this algorithm is the first attempt against data anomaly via statistical distance. Evaluations are performed on synthetic and real-world data sets, demonstrating the correctness and effectiveness of the mechanism.

The rest of the paper is organised as follow: Section II introduces preliminaries and recent works on data anomaly detection and statistical distance. Algorithm details are proposed in section III. Section IV presents evaluation results and further findings of the algorithm. And all contents are concluded in section V.

II. Preliminaries and Related Work

A. Data Anomaly Detection

Anomaly detection, also known as outlier detection, has been studied for a long time and discussed in diverse research domains, such as fraud detection, intrusion detection, system monitoring, fault detection and event detection in sensor networks. According to a systematic classification in [8], anomaly detection algorithms deal with input data in the form of points(or records), sequences, graphs and spatial relationships, where point data is the simplest and well studied, others are attracting more attention in new studies.

Prevalent anomalies can be classified into *Point Anomalies*, *Contextual(or Conditional) Anomalies* and *Collective Anomalies*. Point anomalies refers to an individual data instance that is considered anomalous with respect to others. But if it is anomalous only in certain circumstances or a specific context, the instance is regarded as contextual anomaly. If a group of related data(e.g. a segment of sequence) instances is anomalous with respect to other groups in the data set(e.g. the entire sequence), it is called a collective anomaly.

Detection approaches can be categorized into three types according to whether data is labeled: *Supervised*, *Semi-Supervised* and *Unsupervised* anomaly detection. As the name suggests, supervised detection methods train models on completely labeled data while unsupervised detection leverages data without any labeling. Semi-supervised detection approaches train model on data that has labeled instances for only the normal class. Supervised detection is commonly applied when both normal and anomalous data can be obtained. When it comes to the circumstances that anomalous data is hard to obtain or there exist too many diverse types of anomalies to enumerate, semi-supervised or unsupervised approaches are usually taken into consideration.

To make the final decision, detection algorithms mostly yield a score from each input instance, denoting how likely it is anomalous. The algorithm then selects top few as anomalies or compare the score with a threshold. Or, detection algorithms output a label on each instance, then decide whether each label belongs to the normal class.

Currently, distance based [5, 4] and feature evolving algorithms [19, 18, 25] algorithms seize most attention. Others adopted tree isolation [38], model based [35] and statistical methods [40] in certain applications.

To detect collective anomalies, [6] adopted the *ART(Adoptive Resonance Theory)* neural networks to detect time-series anomalies. *Box Modeling* is proposed in [7]. And *Longest Common Subsequence* was leveraged in [3] as similarity metric for symbolic sequence. Markovian modeling techniques are also popular in this domain[34, 32, 21]. [36] depicted groups in social media as combinations of different "roles" and compare groups according to the proportion of each role within each group.

B. Click Farming Detection

C. Statistical Divergence

Statistical divergence, also called statistical distance, is a function which describes the "distance" of one probability distribution to the other on a statistical manifold. Suppose S is a space of probability distributions, then a divergence is a function from S to non-negative real numbers:

$$D(\cdot||\cdot) : S \times S \to \mathbb{R}^+ \tag{1}$$

Divergence between two distributions P and Q, written as $D(P||Q)$, satisfies:

1) $D(P||Q) \geq 0, \forall P, Q \in S$
2) $D(P||Q) = 0$, if and only if $P = Q$

There are many ways to calculate divergence, such as f-divergences, M-divergences and S-divergences. Some of them provides better properties which brings conveniences to the design and implementation of our approach.

1) Kullback-Leibler Divergence: P, Q are discrete probability distributions, $Q(i) = 0$ implies $P(i) = 0$ for $\forall i$, the *Kullback-Leibler Divergence* from Q to P is defined to be:

$$KLD(P||Q) = \sum_{Q(i)\neq 0} P(i) log\Big(\frac{P(i)}{Q(i)}\Big) \tag{2}$$

2) Jensen-Shannon Divergence: P, Q are discrete probability distributions, *Jensen-Shannon Divergence* between P and Q is defined to be:

$$JSD(P||Q) = \frac{1}{2}KLD(P||M) + \frac{1}{2}KLD(Q||M) \tag{3}$$

where $M = \frac{1}{2}(P + Q)$.

A more generalized form is defined to be:

$$JSD_{\pi_1,\ldots,\pi_n}(P_1, \ldots, P_n) = \sum_{i=1}^{n} \frac{1}{\pi_i} KLD(P_i||M) \tag{4}$$

where $M = \sum_{i=1}^{n} \frac{1}{\pi_i} P_i$ and $\sum_{i=1}^{n} \frac{1}{\pi_i} = 1$.

Jensen-Shannon divergence has some fine properties:

1) $JSD(P||Q) = JSD(Q||P), \forall P, Q \in S$.
2) $0 \leq JSD_{\pi_1,\ldots,\pi_n}(P_1, \ldots, P_n) \leq log_k(n)$. If a k based algorithm is adopted.
3) To calculate $JSD(P||Q)$, it need not necessarily to be true that $Q(i) = 0$ implies $P(i) = 0$.

[If P, Q are continuous distribution?]

III. Algorithm Details

Diverse data sets in the real world show certain structures which may be resulted from hidden patterns or relationships among records in a collection of data. For example, the volume of vehicles in the highway and the business transaction records, they may show a relatively stable distribution in the daily scale. Manipulation on those data(e.g. Fig 2) results in a drift or distortion of the distribution, which can be captured to trigger the alarm. Although the population parameters(e.g. mean, variance, etc.) are unknown and usually impossible to obtain, it can be sampled and estimated according to the central limitation theorem.

A. Technical Points

[Can be divided and settled inside later two subsections]

1) Which classifier should be chosen?
2) How to determine the classifier threshold?**[fixed value, 3σ]**
3) How to locate the compromised component?
4) How to deal with slightly drifting distribution?
5) .[to be continued ...]

B. Basic Algorithm

Suppose data chunks in the given data set S are groups of instances sampled from a population driven by a static distribution. And we are given in advance an evidence set E which contains $n(n \geq 2)$ collections of correct sample data. Then each data collection in S can be checked by the following algorithm.

Algorithm 1 Static Classification

Input: Evidence set $E = \{D_1, \ldots, D_n\}$, new data chunk D'
Output: Whether D' is anomaly

1: **for** $i \leftarrow 1$ to n do **do**
2: $P_i \leftarrow$ the distribution of D_i
3: **end for**
4: $M \leftarrow \frac{1}{n} \sum_{i=1}^{n} P_i$
5: **for** $i \leftarrow 1$ to n do **do**
6: $J_i \leftarrow JSD(P_i||M)$
7: **end for**
8: $N \leftarrow$ normal distribution estimated from $J_1, \ldots, J_n$
9: $P' \leftarrow$ distribution of D'
10: $J' \leftarrow JSD(P'||M)$
11: $p \leftarrow$ probability density of J' in N
12: **if** $p <$ threshold T **then**
13: **Return** True
14: **else**
15: **Return** False
16: **end if**

As shown in Algorithm 1, n evidence collections are used to estimate the ground truth population distribution M. Then n evidence divergences are calculated, composing a gaussian classifier to classify the new distribution sample. Although it is convenient to compute $JSD(P_1, \ldots, P_n, P')$ instead of $JSD(P_1||M), \ldots, JSD(P_n||M), JSD(P'||M)$. It is not suitable for classification. Jensen-Shannon divergence of $n+1$ distributions will dilute the affection of the abnormal one, in which case the difference between P' being normal and anomalous will become subtle when n goes larger.

Similar to the fact that sampling values around a certain parameter will yield a gaussian distribution, sampling divergences around a certain population distribution yields a gaussian distribution $N(\mu, \sigma)$ where μ is a value slightly larger than zero. μ can not be zero according to line 4. M takes into consideration all existing values in every distribution sample and averages corresponding probabilities. Thus M may consist entries that does not exist in P_i and probability in certain entries in M may vary from that in P_i. For example, suppose $P_1(1) = 0.5, P_1(2) = 0.3, P_1(3) = 0.2$; $P_2(1) = 0.3, P_2(2) = 0.4, P_2(3) = 0.3$; $P_3(1) = 0.3, P_3(3) = 0.5, P_3(4) = 0.2$, then $M(1) = \frac{11}{3}, M(2) = \frac{7}{3}, M(3) = \frac{10}{3}, M(4) = \frac{2}{3}$. None of P_1, P_2, P_3 is the same with M. Although the distance cannot be negative values, normal distribution is the closest to the distribution of all JSD values.

C. Histogram

It is possible, in some specific circumstances, to assign a continuous function as the model of a collection of sampled data, which will get more accurate results. However, in the consideration of generality and computational cost, a discrete approximation of the distribution sample will be adopted. It means that every data collection will be counted into a histogram according to a certain order, yielding a discrete probability distribution.

When constructing histograms, step size is the most important parameter the algorithm should receive. If the size is too small then the resulting histogram will be over fitting; but if the size is too large then the estimation will be too coarse to depict the original shape. According to statistics theory, when dealing with a sample size of k, a step size of $l = c\sigma k^{-0.2}$ will give a best partition, where c is a constant relative to the shape of distribution(e.g. for normal distribution, $c =$).

D. Threshold

E. Dynamic Algorithm

IV. Evaluation

A. Experiment Environment

B. Methodology

Raise and answer some research questions. Present test background and methods.

C. Experiment on Synthetic Data

Test basic properties of statistical distance and algorithm

D. Experiment on Real World Data

Test performance of algorithm

V. Conclusion

Acknowledgement

References

[1] *Big Data Statistics And Facts for 2017*. 2017. URL: https://www.waterfordtechnologies.com/big-data-interesting-facts/.

[2] Nathan Bronson, Thomas Lento, and Janet L Wiener. "Open data challenges at facebook". In: *Data Engineering (ICDE), 2015 IEEE 31st International Conference on*. IEEE. 2015, pp. 1516–1519.

[3] Suratna Budalakoti et al. "Anomaly detection in large sets of high-dimensional symbol sequences". In: (2006).

[4] Lei Cao et al. "Multi-Tactic Distance-Based Outlier Detection". In: *Data Engineering (ICDE), 2017 IEEE 33rd International Conference on*. IEEE. 2017, pp. 959–970.

[5] Lei Cao et al. "Scalable distance-based outlier detection over high-volume data streams". In: *Data Engineering (ICDE), 2014 IEEE 30th International Conference on*. IEEE. 2014, pp. 76–87.

[6] T Caudell and D Newman. "An adaptive resonance architecture to define normality and detect novelties in time series and databases". In: *IEEE World Congress on Neural Networks, Portland, Oregon*. 1993, pp. 166–176.

[7] Philip K Chan and Matthew V Mahoney. "Modeling multiple time series for anomaly detection". In: *Data Mining, Fifth IEEE International Conference on*. IEEE. 2005, 8–pp.

[8] Varun Chandola, Arindam Banerjee, and Vipin Kumar. "Anomaly detection: A survey". In: *ACM computing surveys (CSUR)* 41.3 (2009), p. 15.

[9] Guoqiang Jerry Chen et al. "Realtime data processing at facebook". In: *Proceedings of the 2016 International Conference on Management of Data*. ACM. 2016, pp. 1087–1098.

[10] Jason C Cohen and Subrata Acharya. "Towards a trusted HDFS storage platform: Mitigating threats to Hadoop infrastructures using hardware-accelerated encryption with TPM-rooted key protection". In: *Journal of Information Security and Applications* 19.3 (2014), pp. 224–244.

[11] Peter Groves et al. "The'big data'revolution in healthcare: Accelerating value and innovation". In: (2016).

[12] Herodotos Herodotou et al. "Scalable near real-time failure localization of data center networks". In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2014, pp. 1689–1698.

[13] Jingwei Huang, David M Nicol, and Roy H Campbell. "Denial-of-service threat to Hadoop/YARN clusters with multi-tenancy". In: *Big Data (BigData Congress), 2014 IEEE International Congress on*. IEEE. 2014, pp. 48–55.

[14] *Is Data Manipulation the Next Step in Cyber Crime*. URL: https://www.cloudmask.com/blog/is-data-manipulation-the-next-step-in-cybercrime.

[15] Masoumeh Rezaei Jam et al. "A survey on security of Hadoop". In: *Computer and Knowledge Engineering (ICCKE), 2014 4th International eConference on*. IEEE. 2014, pp. 716–721.

[16] Gang-Hoon Kim, Silvana Trimi, and Ji-Hyong Chung. "Big-data applications in the government sector". In: *Communications of the ACM* 57.3 (2014), pp. 78–85.

[17] Emmanuel Letouzé and Johannes Jütting. "Official statistics, big data and human development: towards a new conceptual and operational approach". In: *Data Pop Alliance and PARIS21* (2014).

[18] Yaliang Li et al. "On the discovery of evolving truth". In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 2015, pp. 675–684.

[19] Mohammad M Masud et al. "Classification and adaptive novel class detection of feature-evolving data streams". In: *IEEE Transactions on Knowledge and Data Engineering* 25.7 (2013), pp. 1484–1497.

[20] Frank Austin Nothaft et al. "Rethinking data-intensive science using scalable analytics systems". In: *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*. ACM. 2015, pp. 631–646.

[21] Dmitry Pavlov. "Sequence modeling with mixtures of conditional maximum entropy distributions". In: *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*. IEEE. 2003, pp. 251–258.

[22] Murad A Rassam, Mohd Aizaini Maarof, and Anazida Zainal. "Adaptive and online data anomaly detection for wireless sensor systems". In: *Knowledge-Based Systems* 60 (2014), pp. 44–57.

[23] Barna Saha and Divesh Srivastava. "Data quality: The other face of big data". In: *Data Engineering (ICDE), 2014 IEEE 30th International Conference on*. IEEE. 2014, pp. 1294–1297.

[24] Felix Schuster et al. "VC3: Trustworthy data analytics in the cloud using SGX". In: *Security and Privacy (SP), 2015 IEEE Symposium on*. IEEE. 2015, pp. 38–54.

[25] Junming Shao, Zahra Ahmadi, and Stefan Kramer. "Prototype-based learning on concept-drifting data streams". In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2014, pp. 412–421.

[26] Ather Sharif et al. "Current security threats and prevention measures relating to cloud services, Hadoop concurrent processing, and big data". In: *Big Data (Big Data), 2015 IEEE International Conference on*. IEEE. 2015, pp. 1865–1870.

[27] Priya P Sharma and Chandrakant P Navdeti. "Securing big data hadoop: a review of security issues, threats and solution". In: *Int. J. Comput. Sci. Inf. Technol* 5.2 (2014), pp. 2126–2131.

[28] Biplab Sikdar. "Spatio-Temporal Correlations in Cyber-Physical Systems: A Defense Against Data Availability Attacks". In: *Proceedings of the 3rd ACM Workshop on Cyber-Physical System Security*. ACM. 2017, pp. 103–110.

[29] Roshan Sumbaly, Jay Kreps, and Sam Shah. “The big data ecosystem at linkedin”. In: *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*. ACM. 2013, pp. 1125–1134.

[30] Duygu Sinanc Terzi, Ramazan Terzi, and Seref Sagiroglu. “A survey on security and privacy issues in big data”. In: *Internet Technology and Secured Transactions (ICITST), 2015 10th International Conference for*. IEEE. 2015, pp. 202–207.

[31] TowerData. *4 Steps to Eliminating Human Error in Big Data*. 2013. URL: http://www.towerdata.com/blog/bid/113787/4-Steps-to-Eliminating-Human-Error-in-Big-Data (visited on 06/30/2017).

[32] Christina Warrender, Stephanie Forrest, and Barak Pearlmutter. “Detecting intrusions using system calls: Alternative data models”. In: *Security and Privacy, 1999. Proceedings of the 1999 IEEE Symposium on*. IEEE. 1999, pp. 133–145.

[33] Zhang Xu et al. “High fidelity data reduction for big data security dependency analyses”. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2016, pp. 504–516.

[34] Nong Ye et al. “A markov chain model of temporal behavior for anomaly detection”. In: *Proceedings of the 2000 IEEE Systems, Man, and Cybernetics Information Assurance and Security Workshop*. Vol. 166. West Point, NY. 2000, p. 169.

[35] Jianhua Yin and Jianyong Wang. “A model-based approach for text clustering with outlier detection”. In: *Data Engineering (ICDE), 2016 IEEE 32nd International Conference on*. IEEE. 2016, pp. 625–636.

[36] Rose Yu, Xinran He, and Yan Liu. “Glad: group anomaly detection in social media analysis”. In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 10.2 (2015), p. 18.

[37] Xianqing Yu, Peng Ning, and Mladen A Vouk. “Enhancing security of Hadoop in a public cloud”. In: *Information and Communication Systems (ICICS), 2015 6th International Conference on*. IEEE. 2015, pp. 38–43.

[38] Xuyun Zhang et al. “LSHiForest: A Generic Framework for Fast Tree Isolation Based Ensemble Anomaly Analysis”. In: *Data Engineering (ICDE), 2017 IEEE 33rd International Conference on*. IEEE. 2017, pp. 983–994.

[39] Zhiyuan Zheng and AL Reddy. “Towards Improving Data Validity of Cyber-Physical Systems through Path Redundancy”. In: *Proceedings of the 3rd ACM Workshop on Cyber-Physical System Security*. ACM. 2017, pp. 91–102.

[40] Yunyue Zhu and Dennis Shasha. “Statstream: Statistical monitoring of thousands of data streams in real time”. In: *Proceedings of the 28th international conference on Very Large Data Bases*. VLDB Endowment. 2002, pp. 358–369.