Fast Outlier Detection Despite Duplicates

Jay-Yoon Lee[1] **Danai Koutra**[1] **U Kang**[2] **Christos Faloutsos**[1]

[1]*Carnegie Mellon University* [2]*Korea Advanced Institute of Science and Technology*

School of Computer Science
Carnegie Mellon University

Problem Definition

Given a cloud of multi-dimensional points, detect outliers (i.e. points that differ significantly from the rest) in a scalable way taking care of the major problem of duplicate points.

Motivation

many traditional outlier detection methods are slow due to the big number of duplicate points in datasets (overplotting is a main problem especially in graphs with power-law degree distribution)

Reasons duplicates were not handled

(a) they dealt with relatively small amount of data with few –if any- duplicates, and
(b) most were developed to work on Geographical Information System (GIS) data which do not have many duplicates.

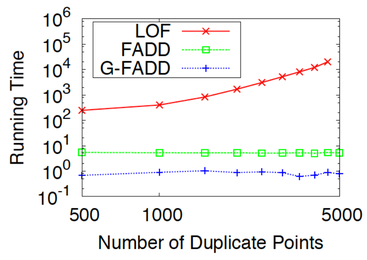
Baseline Approach & our Methods

Baseline method:
LOF: based on the kNN method
Main idea: a point is outlier if its local density is different from the density of its neighbors.
Problems with many duplicate points:
1.Runtime is $O(\max(c_i^2))$, where c_i is count of duplicates for unique element u_i
2.It is not well defined due to division by 0.

	Stack Overflow		US Patent	
Top5	*count*	$count^2$	*count*	$count^2$
1	4221	17.8 M	60598	3.7 B
2	3799	14.4 M	59744	3.6 B
3	3147	9.9 M	56191	3.2 B
4	2844	8.1 M	49192	2.4 B
5	2374	5.6 M	41929	1.8 B
sum	16385 (6.70%)	55.8 M (61.7%)	267654 (12.9%)	14.7 B (79.5%)

FADD
-Main Idea 1: identical coordinates in n-dimensional space are treated as a super node with their duplicate count information, c_i
-Main Idea 2: the distance between duplicate points is set to some small value ε (division by ε instead of 0).

G-FADD
To avoid bias toward highcreate n-dimensional boxes with each dimension -density regions, we introduce a grid-based method:
1.equal to l
2. count the number of points in each box
2a. if #points > k+1, continue with next box
2b. else, apply FADD in this box



Datasets

- real-world networks

Data	# Dimensions	# Points	Description
Twitter 2009	3	39,972,230	degree - PageRank - triangle
US Patent	2	2,076,613	degree - PageRank
Weibo	5	2,158,558	tweets - followees - at - retweets - comments
Stack Overflow	2	243,776	degree - PageRank

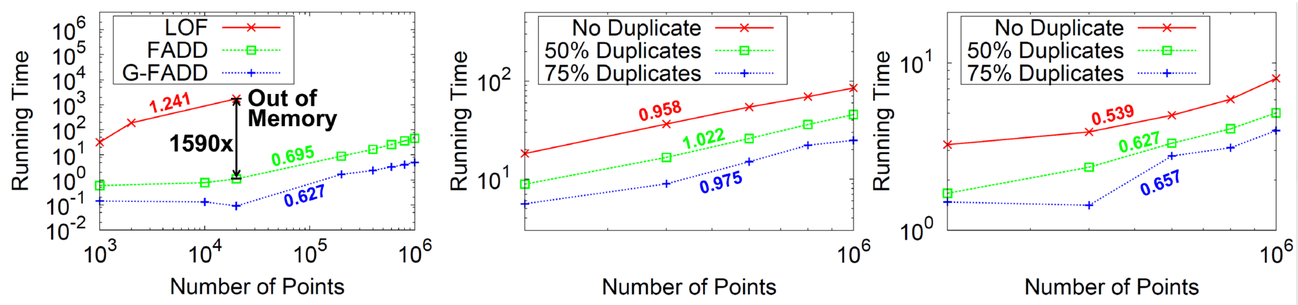
References

C. C. Aggarwal and P. S. Yu. Outlier detection for high dimensional data. In *SIGMOD,* pages 37–46, 2001.
H.-P. Kriegel, M. Schubert, and A. Zimek. Angle-based outlier detection in high-dimensional data. In KDD, pages 444–452, 2008.
L. Akoglu, M. McGlohon, and C. Faloutsos. OddBall: Spotting Anomalies in Weighted Graphs. In PAKDD, 2010.
S. D. Bay and M. Schwabacher. Mining distance-based outliers in near linear time with randomization and a simple pruning rule. In *KDD,* pages 29–38, 2003.
M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. Lof: Identifying density-based local outliers. In SIGMOD, 2000.
G. H. Orair, C. Teixeira, Y. Wang, W. M. Jr., and S. Parthasarathy. Distance-based outlier detection: Conso-lidation and renewed bearing. PVLDB, 2010.
V.Chandola,A.Banerjee,andV.Kumar.Anomalydetection:Asurvey.ACMComput.Surv., 41(3), 2009.

Full paper: http://reports-archive.adm.cs.cmu.edu/anon/2012/abstracts/12-146.html

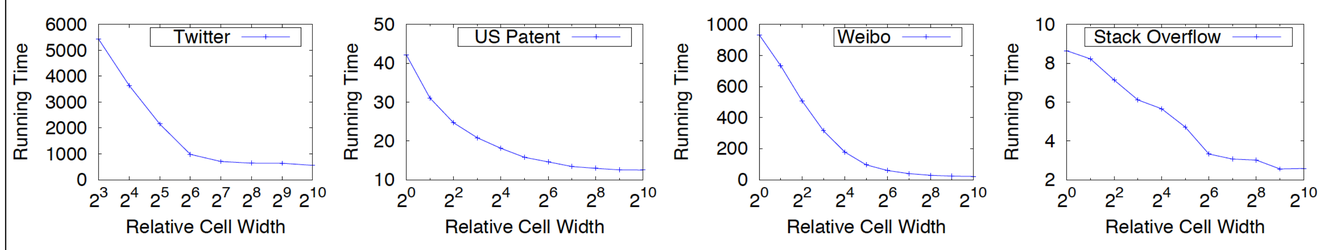
Experimental Results

Running Times



(a) FADD and G-FADD vs. LOF (b) FADD (c) G-FADD

(a) Comparison between LOF, FADD, and G-FADD.
(b) Runtime of FADD with different ratio of duplicate points.
(c) Runtime of G-FADD with different ratio of duplicate points.

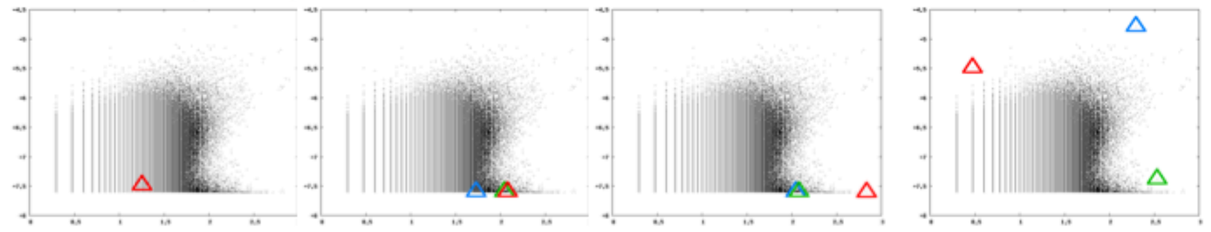


(a) Twitter (b) US Patent (c) Weibo (d) Stack Overflow

G-FADD at Work

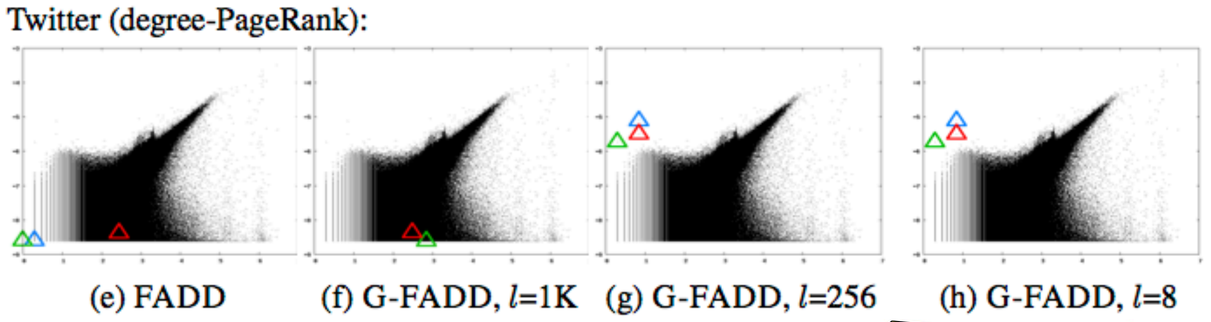
Fine granularity (local outliers) ⟷ *Coarse granularity (global outlier*

US Patent (degree-PageRank):



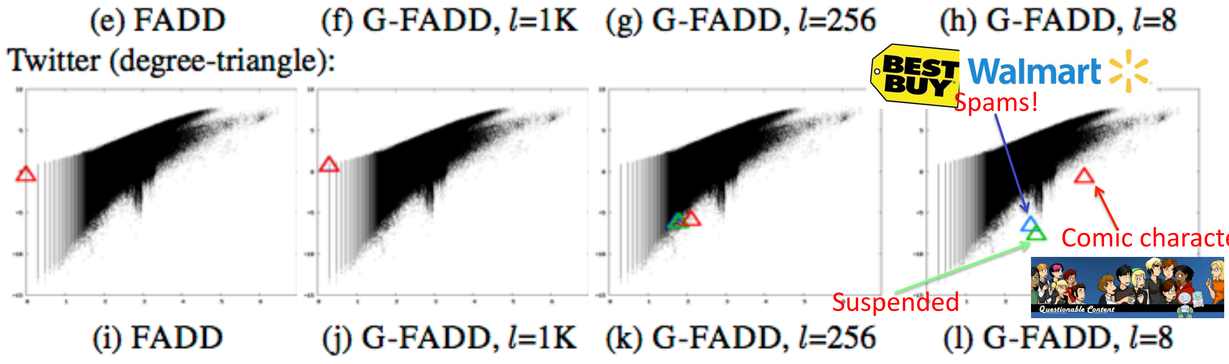
(a) FADD (b) G-FADD, l=1K (c) G-FADD, l=256 (d) G-FADD, l=8

Twitter (degree-PageRank):



(e) FADD (f) G-FADD, l=1K (g) G-FADD, l=256 (h) G-FADD, l=8

Twitter (degree-triangle):



(i) FADD (j) G-FADD, l=1K (k) G-FADD, l=256 (l) G-FADD, l=8

- The blue, green, and red triangles denote the points with the 1st, 2nd, and 3rd largest outlier scores.
- (l) Relatively small number of triangles compared to their neighbors
  - Advertisement spammer (blue): 3 tweets of free gift card of Best Buy and Walmart, 7594 followers, 0 followee.
  - Comic Character (red) : 11207 followers, 6 followees.
- (h) The top 3 outliers are unpopular accounts with very small degrees (7, 2, and 7, respectively), but their neighbors have relatively high degrees: the average degrees of neighbors are 1646, 89, and 343014, respectively. Due to the high-degree neighbors, they have higher PageRanks despite their low degrees.

CONCLUSIONS

1. **No Degeneracy**. We re-design the standard outlier detection algorithm to remove degeneracy that comes from duplicate points in large, real world data.
2. **Running Time**. Near-linear running time compared to the near-quadratic running time of LOF.
3. **Discovery**. We find interesting outliers including Twitter accounts for advertisement spams, and nodes with high PageRanks despite small degrees.

Future research directions: Combining different grid-size results and on-line outlier detection algorithms to handle duplicate points for streaming data.