Fast and Accurate Group Outlier Detection for Trajectory Data

Youcef Djenouri[1], Kjetil Nørvåg[2], Heri Ramampiaro[2], Jerry Chun-Wei Li[3]

[1] Dept. of Mathematics and Cybernetics, SINTEF Digital, Oslo, Norway
[2] Dept. of Computer Science, NTNU, Trondheim, Norway
[3] Dept. of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway
youcef.djenouri@sintef.no,{noervaag,heri}@ntnu.no, jerrylin@ieee.org

Abstract. Previous approaches to solve the trajectory outlier detection problem exclusively examine single outliers. However, anomalies in trajectory data may often occur in groups. This paper introduces a new problem, *group trajectory outlier detection* (GTOD) and proposes a novel algorithm, named, CDkNN-GTOD (**C**losed **D**BSCAN **kN**earest **N**eighbors for **G**roup **T**rajectory **O**utlier **D**etection). The process starts by determining micro clusters using the DBSCAN algorithm. Next, a pruning strategy using kNN is performed for each micro cluster. Finally, an efficient pattern mining algorithm is applied to the resulting subsets of group of trajectory candidates to determine the group of trajectory outliers. We performed a comparative study using real trajectory databases to evaluate the proposed approach. The results have shown the efficiency and effectiveness of CDkNN-GTOD.

Keywords: Group Trajectory Outlier Detection · Pattern Mining · Clustering.

1 Introduction

The proliferation of GPS devices has resulted in countless of sequence points representing trajectories being generated, stored, and analyzed in the context of urban data [4]. Without loss of generality, in the context of intelligent transportation, the data analyst is faced with a myriad of trajectories derived from the mobility of people, cars, buses, taxis, among others. Previous approaches to solve the trajectory outlier detection have solely considered *individual* outliers. In real-world applications, however, trajectory outliers often appear in groups, e.g., a group of bikes that deviates to the usual trajectory due to the maintenance of streets. This paper presents a new problem of trajectory outlier detection called *Group Trajectory Outlier Detection* (GTOD), which the goal is to identify group of anomalous behaviours from trajectory data.

Motivation and idea. Consider the example of taxi trajectories, each trajectory is mapped to the road map network. Traditional trajectory outlier detection algorithms, e.g., [1], may detect individual outliers. However, these algorithms

cannot identify outliers, where a group of taxis deviate from the usual trajectory. Detecting such trajectory outliers, could help (taxi) planners to study the different correlations between these trajectories to deduce useful information. For example, a group of taxi trajectory outliers could indicate that the taxis are partners in a taxi fraud. However, by observing only *individual* deviations, such a possible fraud would be hard to reveal. Motivated by the limitation of solely identifying individual outliers, we focus on studying, and determining group of trajectory outliers. To do so, we first define a new problem called *Group Trajectory Outlier Detection*, and then propose a novel approach for finding these kind of anomalies. The process starts by determining the micro clusters, using DBSCAN, each micro cluster is considered as a candidate group of trajectory outliers. Each group contains several individual trajectory outliers that are close to each other. Note that the groups may contain normal trajectories as well. Such trajectories can generally be considered as noises. To remove such noises, the set of group of trajectory outliers are pruned using the kNN algorithm. Finally, we run a pattern mining algorithm to explore the correlation among the pruned groups of trajectory outliers. The discovered frequent patterns are thereafter considered as the final groups of trajectory outliers.

Contribution. This paper presents a new problem called *Group Trajectory Outlier Detection* (GTOD for short), which allows to identify groups of trajectory outliers. The main contributions of the presented work can be summarized as follows. i) We introduce and formulate a new problem called *GTOD: Group Trajectory Outlier Detection* to enable to identify group of trajectory outliers. ii) We propose a new technique, named CDkNN-GTOD (**C**losed **D**BSCAN **kN**earest **N**eighbors for **G**roup **T**rajectory **O**utlier **D**etection), which explores the DBSCAN algorithm for determining candidate outliers represented as micro clusters, kNN algorithm for pruning the micro clusters, and a pattern mining process for discovering the group of trajectory outliers. iii) We demonstrate the performance of the proposed algorithm using different real trajectory databases. The results of experiments reveal that CDkNN-GTOD outperforms the baseline algorithms for group outlier detection.

2 Related Work

Chalapathy et al. [2] proposed the deep generative model to find out the group outliers on various image applications. The outlierness for each group in the input data was then estimated by group reference function using the backpropagation algorithm. Liang et al. [12] developed a flexible genre model to find specific group outliers. Their main idea was to characterize data groups at both point and group level to detect various types of anomalous groups. Das et al. [3] explored the different correlations between data outliers to detect anomalous patterns using Bayesian network anomaly detection and conditional anomaly detection. Xiong et al. [11] proposed a group outlier detection approach by defining a mixture of Gaussian mixture model. It adopted the likelihood of each group, the marginal likelihood of each observation within a group, and the max-

imum likelihood estimation to learn the hyperparameters of the mixture model. Soleimani et al. [9] developed a supervised learning approach that groups anomalous patterns when memberships are previously unknown. The salient features were extracted from an appropriate training set with discrete data inputs. Li et al. [7] assigned feature weights on each group outlier, and computed chain rule entropy to determine correlation between different feature groups. Toth et al. [10] reviewed both static, and dynamic group anomaly detection solutions. The static group anomaly detection is the process of identifying groups that are not consistent with regular group patterns, while dynamic group change detection assesses significant differences in the state of a group over a period of time. In contrast to this, in this study, we are interested in dealing with static group anomaly detection on the trajectory data. From this brief review, we can conclude that approaches to group outlier detection algorithms are mainly based on some known distributions to find group outliers. In real scenarios, it is hard to fit the data to such distributions. In this paper, we introduce a new problem called group of trajectory outlier detection and propose a new data mining approach, which do not need to know the distribution of the input data to determine the group of trajectory outliers.

3 Problem Statement

Definition 1 (Trajectory Database). *We define a trajectory database* $T = \{T_1, T_2...T_m\}$*, where each raw trajectory* T_i *is a sequence of spatial location points* $(p_{i1}, p_{i2}...p_{in})$*, obtained by localization techniques such as GPS. Each point is represented by the latitude, and the longitude values, respectively.*

Definition 2 (Mapped Trajectory Database). *We define a mapped trajectory database* $\Lambda = \{\Lambda_1, \Lambda_2...\Lambda_m\}$*, where each mapped trajectory* Λ_i *is a sequence of spatial location regions* $(R_{i1}, R_{i2}...R_{in})$*, obtained by mapping each point in* T_i *to the closest region* R_i*. We note* $R = \{R_1, R_2...R_{|R|}\}$*, by the set of all regions.*

Definition 3 (Trajectory Dissimilarity). *We define the distance between two trajectories* $d(\Lambda_i, \Lambda_j)$ *by the number of all regions minus the number of shared regions between the two trajectories* Λ_i*, and* Λ_j*, as*

$$d(\Lambda_i, \Lambda_j) = n - |\{(R_{il}, R_{jl}) | R_{il} = R_{jl}, \forall l \in [1..n]\}| \tag{1}$$

Definition 4 (Group Trajectory Candidate). *We define a group of trajectory candidate* $\mathcal{G}$ *by the set of individual trajectory outliers retrieved from the set of individual trajectory outliers* ITO*, i.e.,*

$$\mathcal{G} = \{\Lambda_i | \Lambda_i \in ITO\} \tag{2}$$

Definition 5 (Density Group). *We define the density of the candidate group trajectory outliers* $\mathcal{G}$ *as*

$$Density(\mathcal{G}) = \frac{|\mathcal{G}|}{|\{R_j | \Lambda_i \in \mathcal{G}, R_j \in \Lambda_i\}|} \tag{3}$$

To normalize the density function, we divide the result by the density of the group having maximum density value, this ensures to obtain values ranged from 0 to 1. We call this function NormalizedDensity.

Definition 6 (Group Trajectory Outlier). *A set of trajectories $\mathcal{G}$ is called a Group Trajectory Outlier if and only if,*

$$\begin{cases} \mathcal{G} \subseteq ITO \\ NormalizedDensity(\mathcal{G}) \geq \gamma \end{cases} \tag{4}$$

Note that γ is the density threshold varied from $[0 \dots 1]$.

Definition 7 (Non-Redundant Group Trajectory Outlier). *A group of trajectory outliers $\mathcal{G}$ is called a Non-Redundant Group Trajectory Outlier if it has no superset of $\mathcal{G}$, that is a group of trajectory outlier.*

Definition 8 (Group Trajectory Outlier Detection Problem). *Group Trajectory Outlier Detection Problem aims to discover from the set of all mapped trajectories, the set of all non-redundant groups of trajectory outliers, denoted by $\mathcal{G}^*$.*

4 CD*k*NN-GTOD Algorithm

This section presents our algorithm CDkNN-GTOD, (Closed DBSCAN k Nearest Neighbors for Group Trajectory Outlier Detection). Our main goal is to efficiently explore the enumeration tree of the trajectory candidates to determine the group of trajectory outliers. In this work, we inspire by the clustering, the neighborhood computation, and the pattern mining algorithms to accurately prune the search space and find the group of trajectory outliers. The process starts by finding the micro clusters using DBSCAN algorithm, the pruning strategy is performed for each micro cluster using the kNN principle. An efficient pattern mining algorithm is then explored on the resulted subset of group of trajectory candidates to determine the groups of trajectory outliers. In the remaining of this section, we show how to use all these concepts in the CDkNN-GTOD framework.

4.1 Clustering

Before presenting the clustering step, we need formally define some basic concepts.

Definition 9 (Trajectory Neighborhoods). *We define the neighborhoods of a trajectory Λ_i, $\mathcal{N}_{\Lambda_i}$, for a given threshold ϵ by*

$$\mathcal{N}_{\Lambda_i} = \{\Lambda_j | d(\Lambda_i \Lambda_j) \leq \epsilon \vee j \neq i\} \tag{5}$$

Definition 10 (Core Trajectory). *A trajectory Λ_i is called core trajectory if there is at least a minimum number of trajectories MinPts such that $|\mathcal{N}_{\Lambda_i}| \geq MinPts$*

Definition 11 (Micro Cluster). *A cluster of trajectories C_i is called a micro cluster if and only if $0 < |C_i| \leq \mu$, where μ is a user threshold.*

This section presents how to use *DBSCAN* algorithm to identify micro clusters, each micro cluster is considered as group of trajectory outlier candidates. The ϵ-neighborhood of each trajectory is computed using Def. 9. The core trajectories are determined using Def. 10. DBSCAN then iteratively collects density-reachable trajectories from these core trajectories directly, which may involve merging a few density-reachable clusters. The process terminates when no new trajectories can be added to any cluster. Initially, the set of trajectories are grouped using *DBSCAN*. This generates several clusters with different sizes. Each micro cluster (See Def. 11) is considered as group candidates. As a result, sets of groups trajectory candidates called $\{\mathcal{G}_i^+\}$ are generated.

4.2 Pruning Strategy

The clustering step returns micro clusters, where each micro cluster forms the groups of trajectory candidates. These groups contain individual trajectory outliers close to each other. However, they may contain normal trajectories. To well prune the groups trajectory candidates, we develop an efficient pruning strategy based on kNN principle. Before presenting the pruning step, we need formally define some basic concepts.

Definition 12 (kNN of a trajectory). *We define kNN of a trajectory Λ_i, denoted by $kNN(\Lambda_i)$ as*

$$kNN(\Lambda_i) = \{\Lambda_j \in \Lambda \setminus \{\Lambda_i\} | d(\Lambda_i, \Lambda_j) \leq k_{dist}(\Lambda_i)\} \tag{6}$$

$k_{dist}(\Lambda_i) = d(\Lambda_i, \Lambda_l)$ is the k-distance of the trajectory Λ_i defined such as it exists k trajectories $\Lambda' \in \Lambda$, it holds that $d(\Lambda_i, \Lambda_l) \geq d(\Lambda_i, \Lambda')$

Definition 13 (Outlierness degree of a Trajectory). *We define the outlierness degree of a given trajectory Λ_i, denoted by $\delta(\Lambda_i)$ as*

$$\delta(\Lambda_i) = |\{\Lambda_j | j \neq i \vee \Lambda_j \in (kNN(\Lambda_i) \cap \mathcal{G}^+)\}| \tag{7}$$

In the following, we present an adapted kNN algorithm for pruning the candidate trajectory outliers. The algorithm considers as input the sets of all trajectory candidate $\mathcal{G}^+$. The process aims to reduce the number of candidate trajectory outliers on each micro cluster. For each micro cluster, it first adds the trajectory outlier with highest outlierness degree, Λ_1^+, to the set of candidate trajectory outliers labeled by Λ_1^+, and denoted by $\mathcal{G}_1^+$. It then generates all potential candidates from Λ_1^+. A trajectory t is a potential candidate from Λ_1^+, if and only if, $t \in \mathcal{G}_1^+ \vee t \in k\text{NN}(\Lambda_1^+)$. The same process is recursively applied for all potential candidates added to $\mathcal{G}_1^+$, and the overall process is repeated for all micro clusters.

4.3 Pattern Mining

Consider GTOD problem $< R, \mathcal{G}^+, \mathcal{G}^*, NormalizedDensity(\bullet), \gamma >$, it could be fit to the pattern mining problem [6] represented by the set of all transactions D, the set of items I, the support function $Support$, the minimum support $minsup$, and the set of all returned patterns P, as follows,

$$D = R, I = \mathcal{G}^+, Support(\bullet) = NormalizedDensity(\bullet), minsup = \gamma, \mathcal{G}^* = P$$

Each region is viewed as a transaction, and each trajectory candidate is viewed as an item. A pattern is a subset from $\mathcal{G}^+$ already pruned. The support of the pattern p is equal to the density of the group of trajectories of p. The minimum threshold will be γ threshold. A pattern mining process is applied on the set of transactions D, and the set of items I, with the support function $NormalizedDensity(\bullet)$, and with the minimum support set to γ. Each frequent pattern discovered is considered as a set of group of trajectory outliers. By definition, GTOD problem aims to identify **non-redundant** group of trajectory outliers. If we apply classical pattern mining algorithm [5], redundant patterns may be extracted. To deal with this issue, we aim to discover closed patterns, this ensures non-redundant group of trajectory outliers are derived. In our implementation, we used Closet algorithm [8] to find out the closed patterns. It proceeds in two steps. Initially, all closed frequent patterns of size 1 are mined. Then, new patterns are generated by directly working on the closed frequent patterns of size 1, without mining additional frequent patterns. It used sparse two efficient data structures id-lists and vertical id-lists for fast counting the support of closed frequent patterns patterns and one-step technique to prune the search space and check the closure property.

5 Performance Evaluation

Extensive experiments have been carried out to compare the CDkNN-GTOD algorithm with the state-of-the art group outlier detection algorithms. The evaluation is performed using ROCAUC, which is common measure for the evaluation of outlier detection methods. We perform the experiments using well-known trajectory databases, retrieved from different repositories, consisting of the following: Geolife[4], Manhattan[5], ECML PKDD 2015 competition[6], and big taxi trajectories: taxi 13-1, taxi 13-2, and taxi 15 [13].

5.1 Parameter Settings

The first part of this experiment focuses on tuning the parameters of different stages of CDkNN-GTOD algorithm. It is performed on two parts, the first one

[4] https://www.microsoft.com/en-us/research/publication/geolife-gps-trajectory-dataset-user-guide/
[5] https://lab-work.github.io/data/
[6] http://www.geolink.pt/ecmlpkdd2015-challenge/dataset.html

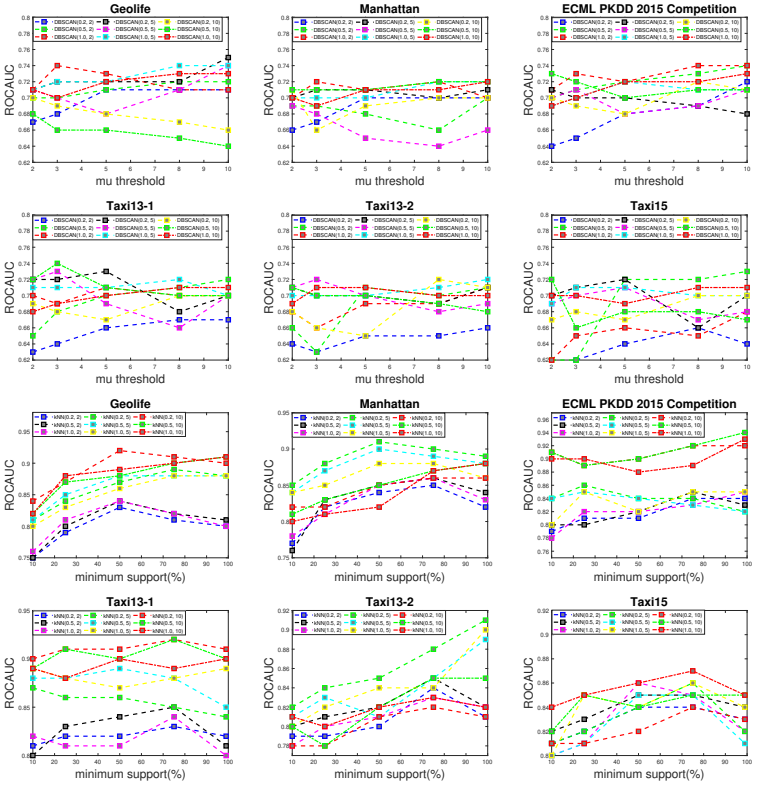

Fig. 1. The parameter setting of the CDkNN-GTOD

is to tune the parameters of the clustering step represented by the DBSCAN parameters (ϵ, and MinPts), μ for determining the micro clusters, the second one is to tune the parameters of kNN represented by the number of neighborhood, k, and the density threshold γ, and the parameter of the pattern mining step represented by the minimum support threshold, minsup. Figure 1 shows the first part of the parameters setting, by considering the micro clusters retrieved in the clustering step as group of trajectory outliers, and ignoring the pruning and the pattern mining processes. Several tests have been performed using different trajectory databases by varying the DBSCAN parameters, ϵ from 0.2 to 1.0, and MinPts from 2 to 10, the μ parameter for determining the micro clusters from 2 to 10. Whatever the trajectory database used as input, the accuracy determined by the ROCAUC value exceeds 0.72, however does not go up 0.75. These results are explained by the fact that the idea of the micro clusters is able to identify the group of trajectory outliers but not in an optimal way. Therefore, in the next experimentation, we tune the parameters of the pruning and the pattern mining processes, by fixing the best parameters of the clustering step for each trajectory database found in this part. The results of the second part is

highlighted in Figure 1, we varied the number of neighborhood from 2 to 10, the density threshold values from 0.2 to 1.0, and the minimum support values from 10% to 99%. The results reveal that the pruning and the pattern mining steps improve the accuracy of the proposed algorithm. This is explained by the fact that kNN strategy allows to prune the search and keep only the most neighbors of trajectory outliers in the micro clusters. Moreover, the pattern mining process further reduces the search space by exploring the frequent patterns among the group of trajectory outliers in the micro clusters. Table 1 summarizes the best parameters values of the CDkNN-GTOD algorithm, which will be used in the remaining of the experiments.

Table 1. Best parameters of CDkNN-GTOD.

Database	ϵ	MinPts	μ	k	γ	minsup
Geolife	0.2	5	10	10	0.2	50
Manhattan	0.5	10	8	5	0.2	50
ECML PKDD 2015 Competition	1.0	10	8	10	0.5	99
Taxi13-1	0.5	10	3	10	0.5	75
Taxi13-2	0.5	5	3	10	0.5	99
Taxi15	0.5	10	10	10	1.0	75

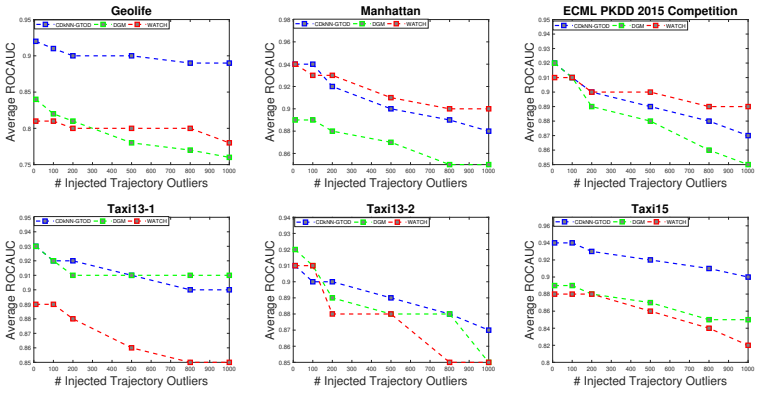

Fig. 2. CDkNN-GTOD vs. state-of-the-art group outlier detection algorithms: accuracy

5.2 CDkNN-GTOD Vs State-of-the-art Group Detection Algorithms

The aim of this experiment is to compare CDkNN-GTOD with the baseline algorithms in terms of accuracy and processing time. To the best of our knowledge,

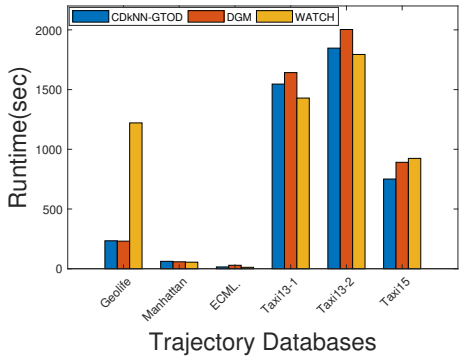

Fig. 3. CDkNN-GTOD vs. state-of-the-art group outlier detection algorithms: runtime

this is the first work which investigates the group outlier detection in trajectory data. Therefore, we adopt two baseline group outlier detection algorithms (DGM [2], and WATCH [7]) to trajectory data for comparison with CDkNN-GTOD. Figure 2 presents the average ROCAUC value of the proposed algorithm CDkNN-GTOD, and the baseline group outlier detection algorithms (DGM and WATCH), using several trajectory databases, and with different number of injected outliers. By varying the number of injected trajectories from 10 to 1000, the CDkNN-GTOD outperforms the other algorithms for almost of cases. Among 36 cases shown, CDkNN-GTOD is the best for 22 cases, DGM for 8 cases, and WATCH for 6 cases. Moreover, when increasing the number of injected trajectory outliers, the accuracy of the CDkNN-GTOD stabilizes and do not go under 0.87, whereas, the accuracy of the baseline algorithm goes under 0.80. This comes from the fact that our approach uses more advanced and recent strategies, based on clustering, neighborhoods, and pattern mining, while the baseline approaches use less advanced concepts of outlier detection based on data distribution. Regarding processing speed, as shown in Figure 3, our approach is very competitive compared to the baseline approaches. This is explained the way we combined the efficient data mining techniques – clustering, kNN, and pattern mining, for finding the groups of trajectory candidates.

6 Conclusion

In this paper, we introduced a new problem that aims at discovering group of trajectory outliers. To solve this problem we proposed to combine clustering, pruning, and pattern mining. More specifically, our approach consisted of three main steps: (1) determination of micro clusters using the DBSCAN algorithm, (2) identification of potential group of trajectory candidates from the micro clusters with kNN, and (3) pruning of the candidates using density computation/pattern mining. Each of these steps are executed in an iterative manner, allowing to extract the group of trajectory outliers in an effective and efficient manner. To

evaluate our approach, we performed our comparative experiments on different real trajectory databases. The experiments showed that our approach achieved good results in terms of both accuracy and processing speed. Overall, the proposed approach is indeed capable of effectively and efficiently solving the GTOD problem, and that it outperforms traditional methods which are based on data distribution.Nevertheless, the combination of the advanced techniques requires high expertise not only in trajectory analysis or outlier detection, but in other sophisticated data mining techniques. In our future work, we will investigate and target new applications of *GTOD*, such as climate change analysis, e.g., finding a group of hurricane trajectories that deviates from the normal hurricane ones. This would allow to early identify other cities that could be affected.

References

1. Belhadi, A., Djenouri, Y., Lin, J.C.W.: Comparative study on trajectory outlier detection algorithms. In: 2019 International Conference on Data Mining Workshops (ICDMW). pp. 415–423. IEEE (2019)
2. Chalapathy, R., Toth, E., Chawla, S.: Group anomaly detection using deep generative models. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 173–189. Springer (2018)
3. Das, K., Schneider, J., Neill, D.B.: Anomaly pattern detection in categorical datasets. In: Proceedings of the 14th ACM SIGKDD. pp. 169–176 (2008)
4. Djenouri, Y., Belhadi, A., Lin, J.C.W., Djenouri, D., Cano, A.: A survey on urban traffic anomalies detection algorithms. IEEE Access **7**, 12192–12205 (2019)
5. Djenouri, Y., Djenouri, D., Lin, J.C.W., Belhadi, A.: Frequent itemset mining in big data with effective single scan algorithms. Ieee Access **6**, 68013–68026 (2018)
6. Djenouri, Y., Lin, J.C.W., Nørvåg, K., Ramampiaro, H.: Highly efficient pattern mining based on transaction decomposition. In: 2019 IEEE 35th International Conference on Data Engineering (ICDE). pp. 1646–1649. IEEE (2019)
7. Li, J., Zhang, J., Pang, N., Qin, X.: Weighted outlier detection of high-dimensional categorical data using feature grouping. IEEE Transactions on Systems, Man, and Cybernetics: Systems (99), 1–14 (2018)
8. Pei, J., Han, J., Mao, R., et al.: Closet: An efficient algorithm for mining frequent closed itemsets. In: ACM SIGMOD workshop on research issues in data mining and knowledge discovery. vol. 4, pp. 21–30 (2000)
9. Soleimani, H., Miller, D.J.: ATD: anomalous topic discovery in high dimensional discrete data. IEEE Transactions on Knowledge and Data Engineering **28**(9), 2267–2280 (2016)
10. Toth, E., Chawla, S.: Group deviation detection methods: A survey. ACM Computing Surveys (CSUR) **51**(4), 77 (2018)
11. Xiong, L., Póczos, B., Schneider, J., Connolly, A., VanderPlas, J.: Hierarchical probabilistic models for group anomaly detection. In: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics. pp. 789–797 (2011)
12. Xiong, L., Póczos, B., Schneider, J.G.: Group anomaly detection using flexible genre models. In: Advances in neural information processing systems. pp. 1071–1079 (2011)
13. Zhang, D., Li, N., Zhou, Z.H., Chen, C., Sun, L., Li, S.: iBAT: detecting anomalous taxi trajectories from GPS traces. In: Proceedings of the 13th international conference on Ubiquitous computing. pp. 99–108 (2011)