# Colossal Trajectory Mining: A Unifying Approach to Mine Behavioral Mobility Patterns

Dear editor,

Please find below the detailed answers to every point raised by the reviewer.

Following the reviewers' concerns, we mainly:

(i) extended the related work,
(ii) introduced how to set up our approach,
(iii) clarified its distinguishing contributions.

Our answers in this letter and major changes in the paper are colored and highlighted in blue.

Changes in the paper sum up to around 9 pages out of 43 (appendix and references excluded).

## Reviewer #1

The paper proposes an approach to mine co-movement patterns from trajectory datasets, called colossal trajectory mining (CTM). The authors evaluated the model with state-of-the-art algorithms and on real and synthetic bases. The provided results indicate that the proposed approach is suitable to mine different types of trajectory patterns such as co-location, flow, swarm, and convoy.

The idea of trajectory data mining using frequent itemset mining techniques has already been proposed, e.g. [1-2]. The manuscript should include a review of existing trajectories data mining models that use FIM. The authors should also detail the distinctions of the proposed method over existing ones. I suggest you to analyze in detail the developed work in [1], this has a lot of correlation with the proposed algorithm and was published in this journal.

> Following the reviewers' concerns, we extended the Related Work section and introduced Table 2 to summarize the main differences between approaches related to CTM. The table highlights that there is no direct competitor including all the features of CTM.

[1] Fonseca-Galindo, J. C., de Castro Surita, G., Neto, J. M., de Castro, C. L., & Lemos, A. P. (2022). A multi-agent system for solving the dynamic capacitated vehicle routing problem with stochastic customers using trajectory data mining. Expert Systems with Applications, 195, 116602.

The authors present a stream-based warehouse system sorting logic that assigns incoming packages to UnitLoad agents. UnitLoad agents bets on unit loads, and the values of the bets consider historical routes with a high likelihood of recurrence. Trajectory data mining techniques are used to extract stochastic information from the distribution of packages (i.e., co-location patterns). Trajectories are abstracted into sets of spatial cells, and PFPgrowth is used to find shared path among trajectories.

With respect to CTM:

- The goal of the paper is different: the paper provides a logic to assign packages in a warehouse system
- Only spatial co-location patterns are considered
- Only the spatial feature is considered
- Trajectory mining is used to find shared cells rather than groups of trajectories moving together

[2] Lv, M., Chen, L., Chen, T., Zeng, D., & Cao, B. (2019). Discovering individual movement patterns from cell-id trajectory data by exploiting handoff features. Information Sciences, 474, 18-32.

The authors extract mining movement patterns from cell-id trajectories (i.e., sequences of cell tower identifiers) by estimating the spatial closeness between cell towers and propose a sequential pattern mining algorithm to mine movement patterns by considering such estimated spatial closeness.

With respect to CTM:

- The extracted patterns can be mapped to flow patterns, spatial co-movement patterns where location adjacency is required.
- Only the spatial feature is considered
- The implementation is not distributed

Section 1, paragraph 6: the author defines the terms of colossal mining and big data, could you include references that help to deepen this terminology?

We rephrased the paragraph clarifying the “Colossal” term as follows.

*We emphasize that the terms ``\textit{colossal} (trajectory) mining'' and ``\textit{big} data'' have different meanings. Colossal \cite{DBLP:conf/kdd/PanCTYZ03} is an application of mining techniques to datasets having the number of dimensions (columns) sensibly higher than the number of instances (rows)\footnote{\mf{This happens, for example, in biological datasets such as gene expression datasets that may contain $10^5$ columns but only $10^3$ rows \cite{DBLP:conf/kdd/PanCTYZ03}}.}; colossal mining can require big data techniques, where big data refers to datasets (that produce results) that are too large to be dealt with by centralized data-processing applications.*

Section 2, paragraph 5: There are other similar measures (as Hausdorff distance [3], LCSS distance [4], Frechet distance [5], distance based on road-map [6], among others) which can be used in trajectories, you should dig into them because this directly impacts the results.

We agree with the reviewer: Euclidean, DTW, LCSS (etc.) distance functions affect the results of the clustering algorithm. However, these distance functions are usually adopted with spatio-temporal (only) trajectories, and not in datasets including additional geometric and semantic features. We cited the above-mentioned distance functions while also clarifying their limits in the application of our approach.

*As to the similarity definition, the higher the expressiveness, the higher the computational complexity.*

*The Euclidean distance \cite{van1995some} has linear complexity in trajectory length but requires equally-long trajectories and does not detect time shifts.*

*DTW \cite{DBLP:conf/kdd/RakthanmanonCMBWZZK12} overcomes these limitations and is robust to missing points but at the cost of quadratic complexity.*

*Further distance functions are Hausdorff \cite{DBLP:journals/tip/SimKP99} (the greatest of all the distances from a point in one trajectory to the closest point in the other trajectory), LCSS \cite{DBLP:conf/icde/VlachosGK02}*

*(similarity is expressed in terms of the longest common subsequence of two trajectories), Frechet \cite{DBLP:conf/pods/AgarwalFMNPT18} (the smallest of the maximum pairwise distances), and road based \cite{DBLP:conf/mdm/SilvaLMZC20} (the minimum number of network paths between two trajectories in a certain time window).*

*Different distance functions produce different clusterings (e.g., LCSS, Euclidean, and DTW are compared in \cite{DBLP:conf/icde/VlachosGK02}).*

*However, these \textit{spatio-temporal} distances cannot be directly adopted in our approach for the following reasons:*

*\begin{itemize}*

*\item they do not consider further geometric (e.g., speed or direction) or semantic (e.g., point types or means of transport) features;*

*\item they cannot be applied at different aggregation levels (e.g., neighborhoods and cities);*

*\item they cannot be applied to all co-movement patterns (e.g., LCSS, which returns as similar trajectories sharing a contiguous path, is not suitable for the co-location pattern where path adjacency is not required).*

*\end{itemize}*

Section 2, paragraph 4: in the line "Internal metrics (e.g., the silhouette index) usually measure how much (crisp) partitioning clusters are compact and well-separated" there is a lack of references that can argue/deepen this statement.

> We have rewritten the sentence since it could be ambiguous and added the reference of the silhouette index
>
> *Internal metrics measure how distinguishable clusters are without the need for external knowledge (e.g., the silhouette index measures whether crisp partitioning clusters are compact and well-separated \cite{zhu2010clustering}).*
>
> *However, co-movement patterns can be overlapping since trajectories can contribute to multiple groups of MOs.*

Figure 2 is referenced at the end of section 2, however, Figure 1 is referenced at the beginning of section 3. These images must maintain the sequence within the text.

> Indeed, we wrongly swapped the two Figures. Fixed it.

Section 3.2, Example 1: Path T_G has two points in B2, there may be cases where each point is a point of interest, such as malls or restaurants. In this case, should the information on the number of points per cell be added?

> In our approach the tessellation defines the granularity and the semantics of the analysis. Handling your expressiveness entails two additional features. PlaceType can be easily modeled by adding a specific feature. Modeling how many times MOs visited a place involves a time feature.
>
> We added the above motivation to Example 1
>
> *The tessellation defines the granularity and the semantics of the analysis.*
>
> *If --- for instance --- the user is also interested in distinguishing malls or restaurants or means of transport, space and time features alone are not enough.*
>
> *Additional features (e.g., placeType or transport) must be added to the tessellation.*
>
> *This is the strength of our approach: to consider custom features transparently.*
>
> *Indeed, tiles enable an extensible and transparent management of space, time, and additional features.*

Section 3.3, paragraph 2: in the line "the matching of co-movement patterns [50] since co-movements patterns are incrementally refined", can you define how incremental co-movements patterns are built?

> Incremental refinement is one of core parts of our approach and it is detailed in Sections 5.1 and 5.2. The goal of the sentence in Section 3.3 was just to present the idea. We have rewritten the sentence to provide a clearer understanding and we added a forward reference to the detailed explanation.

> *\item exploits monotonicity properties (e.g., as the generation process proceeds, the cardinality of trajectory groups decreases while the length of the path shared by trajectories in the same group increases) to filter out invalid co-movement patterns without generating them (filtering strategies are detailed in \Cref{sec:pruning}).*

Section 4, definition 7: mSup wasn't defined.

> Fixed it.

> *An itemset is \textit{frequent} (FI) if $|sup(I)|\geq mSup$, where $mSup$ is the minimum number of transactions to consider the itemset as frequent.*

Section 4, definition 8: mCrd wasn't defined.

> Fixed it.

> *A \textit{co-movement pattern} $I$ is a FCI such that $|I| \ge mCrd$, where $mCrd$ is the minimum number of trajectories to consider a FCI as a co-movement pattern.*

Section 4, Table 2: sp and tm weren't defined.

> Fixed it.

> *By definition, a co-movement pattern has at least $mCrd$ trajectories and is at least $mSup$ tiles long. This ``basic'' co-movement pattern can be specialized into those reported in \Cref{tab:proximitypattern} depending on the involved features,* ***either space ($f^{sp}$) or space and time ($f^{tm}$)****, and on the additional shape constraints described in \Cref{tbl:constraints2}.*

Section 5.2, example 6: It is not clear how different mSup values can impact the patterns.

We better detailed the example.

*\begin{example}*

*With reference to \Cref{fig:conncomp}, given $I.CT=\{Q_{A1},Q_{B1}\}$ (green) and $I.RT=\{Q_{C3},\ldots,Q_{D4}\}$ (blue), $I.CT \cup I.RT$ cannot produce a convoy with $mSup=7$ (there are no 7 adjacent tiles) but can potentially produce a convoy with $mSup=6$ (there are 6 adjacent --- blue --- tiles).*

*\end{example}*

Section 6.2: How the thresholds level, mCrd, and mSup were selected?

Table 6: Methodology used for the selection of mCrd, mSup S and T values must be included.

We described the guidelines in detail in a new section “6.2. Parameter Tuning”

*$S$, $mCrd$, and $mSup$ are dataset- and problem-specific parameters (see \Cref{sec:related}) and have been chosen to answer the following business question: \textit{``Which computable patterns are meaningful for our analysis?"}.*

*To do this we rely on the following guidelines.*

*\begin{enumerate}*

*\item Choose a tessellation ($S$) according to the goal of the analysis (e.g., looking for co-movement patterns in the Milan neighborhoods).*

*If a geometric grid is adopted, the tile granularity depends on the level of detail of the analysis.*

*Note that the tessellation should depend on the goal rather than on the optimization of the computation time.*

*\item Set the number of shared tiles ($mSup$) that is relevant for the analysis (e.g., to be considered as interesting, a co-movement pattern must traverse at least 3 neighborhoods).*

*\item Set the minimum group cardinality ($mCrd$) that is relevant for the analysis (e.g., if interested in car-sharing applications $mCrd$ could be set between $2$ and $6$).*

*\item Verify if the combination of parameters determines meaningful patterns in a reasonable amount of time.*

*If not, iterate on the parameters $mSup$ and $mCrd$ until (i) results are ``stable" (i.e., varying $mCrd$ and $mSup$ causes limited changes in the number of co-movement patterns) as prescribed by the elbow method \cite{DBLP:conf/icdcsw/SatopaaAIR11}, and (ii) the solution is computable in a reasonable time.*

*While varying the parameters, consider the following.*

*\begin{itemize}*

*\item Higher values of $mCrd$ and $mSup$ entail a lower number of co-movement patterns (e.g., fewer groups of MOs will share a longer path) and, in turn, reduce the computation time.*

*\item The values of $mCrd$ and $mSup$ should also consider the tessellation.*

*For instance, if trajectories are dense, a fine-grained tessellation determines longer patterns.*

*On the other hand, if trajectories are sparse, a fine-grained tessellation amplifies such sparsity reducing the MOs sharing the same tiles; thus, lower values of $mSup$ or $mCrd$ are needed to increase the number of returned patterns.*

*\end{itemize}*

*\end{enumerate}*

*Although automatizing the approach is beyond the scope of the paper, the points above encode the principles that should drive such an automatic solution.*

As stated above, defining a general approach to set the grid for the Colossal Trajectory mining problem is beyond the scope of this paper and probably entails a brand-new paper. It is well-known that the grid can affect the clustering process when using grid-based approaches and that setting the grid parameters (i.e., the grid resolution or the cell density) can be difficult, especially in multi-dimensional dataset [1] with heterogeneous densities across dimensions. In these cases, even adaptive grids that allow the number of bins in a dimension to change based on the characteristics of the data [2, 3], are not a solution since they require the user to make some assumptions about the clusters to be discovered.

[1] Parsons, Lance, Ehtesham Haque, and Huan Liu. "Evaluating subspace clustering algorithms." Workshop on Clustering High Dimensional Data and its Applications, SIAM Int. Conf. on Data Mining. 2004.

[2] Lu, Yansheng, et al. "A grid-based clustering algorithm for high-dimensional data streams." International Conference on Advanced Data Mining and Applications. Springer, Berlin, Heidelberg, 2005.

[3] Goil, Sanjay, Harsha Nagesh, and Alok Choudhary. "Mafia: Efficient and scalable subspace clustering for very large data sets." Proc. 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Citeseer. 1999.

Additionally to Section 6, the issues related to the tuning of the parameters are discussed and emphasized in many points of the paper:

**Introduction (to emphasize the contribution)**

*\item characterizes MOs through a tessellation including spatial and temporal features as well as additional features that enable the comprehension of semantic mobility behaviors \cite{DBLP:journals/tist/Yan0PSA13} (e.g., characterizing mobility behaviors by means of transport or activity)\footnote{Note that the tessellation is given as input and the tuning of its granularity is out of the scope of the paper, whose goal is introducing an algorithm to mine co-movement patterns out of any (possibly semantic) tessellation. Conversely, tuning the granularity is a dataset specific task (see \Cref{sec:related})};*

**Related work (to improve the positioning of the paper)**

*While CTM could be associated with grid-based clustering, automatically finding the best grid for a specific dataset/problem/co-movement pattern is out of the scope of the paper.*

*To ease the computation of similarity, grid-based clustering uses a grid to build the clusters out of adjacent densely-populated cells.*

*Setting the resolution grid is difficult \cite{parsons2004evaluating}, especially in multi-dimensional datasets with heterogeneous densities across dimensions.*

*In this case, even adaptive grids \cite{DBLP:conf/adma/LuSXL05}*

*are not a solution since they raise the following dataset- and problem-specific issues.*

*\begin{itemize}*

*\item \textit{Dataset-specific}: computing the ``best'' tessellation requires to define what are the ``best'' co-movement patterns.*

*Internal metrics measure how distinguishable are the clusters without the need for external knowledge (e.g., the silhouette index measures whether crisp partitioning clusters are compact and well-separated \cite{zhu2010clustering}).*

*However, co-movement patterns can be overlapping since trajectories can contribute to multiple groups of MOs.*

*External metrics require labels to find the ``best'' co-movement patterns out of the given dataset/tessellation (e.g., the ``purity'' of clusters through entropy); this type of supervised evaluation is usually leveraged to compare different grid-based algorithms \cite{parsons2004evaluating}.*

*Besides the metrics, defining the ``best'' co-movement patterns also depends on the dataset heterogeneity (e.g., taxis in \cite{DBLP:conf/gis/YuanZZXXSH10} produce homogeneous and precise trajectories, while Movebank \cite{DBLP:journals/envsoft/KranstauberCWFTWK11} collects data from thousands of studies).*

*\item \textit{Problem-specific}: co-movement patterns depend on (i) the goal of the analysis (e.g., monitoring co-movement patterns at single user levels at the scale of meters, or aggregated network data \cite{zhang2020exploring} at the scale of kilometers);*

*and (ii) the type of moving objects (e.g., to retrieve a convoy of 5 people moving in the same car a cell of 6 square meters could be enough, but this is not true to detect convoys of cars or trucks spanning hundreds of meters ---for instance in a highway).*

*\end{itemize}*

**Evaluation (to emphasize that setting the tessellation is problem- and dataset-specific)**

*“To mine co-movement patterns with CTM, it is first necessary to identify the business question and then to properly set the tessellation.*

*Tessellations can be built out of diverse types of features, features can be continuous (e.g., speed) or discrete (e.g., means of transport), absolute (e.g., timestamp) or aggregated (e.g., hour bins), geometric (e.g., latitude/longitude) or semantic (e.g., administrative neighborhoods and municipalities, as in \Cref{fig:multilevtess}).”*

*“Since \sf{Oldenburg} and \sf{Hermoupolis} are synthetic datasets --- and no business value can be extracted from them --- we limit the spatial feature to a uniform grid and we shape the size of tiles to get a number of tiles that is comparable to \sf{Milan} (around 90), then we consider the time granularity of minutes for both of them.*

*Additionally, \sf{Hermoupolis} contains the following features: means of transport, activity, and speed (i.e., move or stop).”*

Is there a methodology to get to the optimal value? How do these values impact the results?

Table 8, and Table 9: you show the impact of mCrd and mSup as a function of time, but you do not demonstrate how these values can help in the configuration of these parameters.

The guidelines proposed in Section 6.2 also specify the criteria adopted to define optimality.

*Verify if the combination of parameters determines meaningful patterns in a reasonable amount of time.*

*If not, iterate on the parameters $mSup$ and $mCrd$ until (i) results are ``stable" (i.e., varying $mCrd$ and $mSup$ causes limited changes in the number of co-movement patterns) as prescribed by the elbow method \cite{DBLP:conf/icdcsw/SatopaaAIR11}, and (ii) the solution is computable in a reasonable time.*

*While varying the parameters, consider the following. [...]*

Once again, defining a general approach to set the optimal parameters for the Colossal Trajectory mining problem is beyond the scope of this paper.

Section 6.2.1, paragraph 2: The Milan dataset was selected to answer "Which neighborhoods are crossed by significant groups of people showing high mobility?", and "How does mobility change during the day?". Specify how the patterns obtained can solve these questions, in addition, is there any methodology to search the entire set of patterns for those that can be used? Since, for example, in Flow there are almost 1x10^5 patterns.

> The misunderstanding comes from the definition of “significant.” The paper was missing the results of the analysis and we added them now.
>
> *In the context of urban planning, \textit{``La città intorno''} focused on ranking neighborhoods by their attractiveness in order to understand how to allocate economic resources for requalification.*
>
> *The attractiveness of a neighborhood is defined as the percentage of co-movement patterns passing through that neighborhood.*
>
> *To fulfill the analysis, we initially define a tessellation where the spatial feature represents the 88 neighborhoods in Milan (\Cref{fig:milnil}) and the temporal feature represents a relative dimension that partitions absolute timestamps into six bins, such as night (from 0 to 3) and morning (from 8 to 11); overall $|S| = 88 \cdot 6 = 528$ tiles.*
>
> *Then, together with domain experts, we set relevant values for $mCrd$ and $mSup$. \Cref{fig:qualitativeswarm} depicts an example of a swarm pattern for $mCrd=100$ and $mSup=7$ in which at least 100 people follow the same path around the city center in the morning and from the city center to the central station in the afternoon.*
>
> *\Cref{tbl:att} shows the results of our attractiveness analysis, highlighting the need for higher requalification in ``Lodi - Corvetto'', ``Padova'', and ``Adriano''; the most attractive neighborhoods are the ones closest to the city center\footnote{By filtering tiles on the time bin, it is possible to characterize how attractiveness changes during the day.}.*
>
> Note that attractiveness is independent of the number of returned patterns, and it further summarizes the information carried by the patterns.

In Figure 7 and Figure 8, you draw patterns obtained, however, I missed an interpretability section, in which different patterns are explored, and different use cases are analyzed.

> We detailed the use case in the point above.
>
> Figures 7 and 8 provide a qualitative example of patterns and have been better discussed.

Table 5 is referenced at page 32, Table 6 at page 35, however, Table 7 is referenced at page 33. These tables must maintain the sequence within the text.

Fixed it.

Table 6.2: in the comparison of Figure 9, it is not clear how the amount of movement patterns improves or gets worse in the solution, you should detail the impact of these results in the formulation of the proposed approach.

We better detail the explanation of Figure 9 (now Figure 10) but note that there is no “improvement” or “getting worse” in the number of patterns. As in frequent (closed) itemset mining in the domain of market basket analysis, if the user asks for products that have been bought together at least 5 times, the FIM algorithm returns all groups of products. The group of products that have been bought together at least 4 times will be higher (but not worse). The goal of Figure 9 (now Figure 10) is to show that the number of FIs (Frequent Itemsets) returned by PFPGrowth is always orders of magnitude higher than the number of co-movement patterns returned by CTM and this makes PFPGrowth strongly inefficient.

*Finally, Frequent Itemset Mining approaches, such as PFPGrowth \cite{DBLP:conf/recsys/LiWZZC08}\footnote{Note that any algorithm for FIM can be picked}, could be leveraged to compute \textit{all} FIs and then filtering out only the actual co-movement patterns through a post-processing phase.*

*We leverage PFPGrowth \cite{DBLP:conf/recsys/LiWZZC08}, a well-known implementation of distributed FIM, to compute \textit{all} the (potential) swarm patterns on the entire \sf{Oldenburg} dataset; with respect to SPARE, the computation of FIs is feasible since PFPGrowth does not rely on Apriori enumeration.*

*\Cref{fig:pfpgrowth} shows the comparison results.*

*Although both FIs and co-movement patterns decrease by increasing the $mSup$ threshold (as expected, the higher $mSup$ the lower the valid patterns),*

*the number of FIs to post-process remains 5 orders of magnitude higher than the number of co-movement patterns and its extraction requires orders of magnitude longer execution time (as confirmed in \cite{DBLP:journals/tkde/LuccheseOP06}).*

*This makes FIM approaches strongly inefficient.*

Section 6.3: I have got a few questions:

(1) How could you evaluate the efficiency related to state-of-the-art algorithms?

In this section we evaluate the performance of CTM alone, while in Section 6.4 we evaluate it against its antagonist approach (SPARE) that, to the best of our knowledge, is the (only) antagonist *distributed* approach for the *uniform extraction* of co-movement different patterns.

(2) How does time behave as a function of the number of trajectories and the number of features?

Adding features directly impacts the tessellation by increasing the number of tiles. In Tables 10 and 11 (now 12 and 13) we show how the computational time is affected by increasing the size of the tessellation (by fine-graining the tiles — but this would have the same effect as adding a new feature) and increasing the number of trajectories.

Table 12: for a fairer analysis, you should compare the impact of distributed machine configurations with the distributed implementation of FP growth, for example, PFP: parallel FP-growth [7] implemented in spark

(https://spark.apache.org/docs/latest/ml-frequent-pattern-mining.html).

Could you include an explanation of how these settings impact the patterns obtained?

We discuss the comparison between CTM and PFPGrowth in Section 6.5. Starting from all the itemsets extracted by PFP it would be possible, in principle, to postprocess all of them to get the co-movement patterns directly returned by CTM. However, PFP returns 5 (five) orders of magnitude more frequent itemsets than CTM and does not encode the spatio-temporal adjacency constraints necessary to extract co-movement patterns.

In the paper, we have the following discussion (Section 6.5)

*Finally, Frequent Itemset Mining approaches, such as PFPGrowth \cite{DBLP:conf/recsys/LiWZZC08}\footnote{Note that any algorithm for FIM can be picked}, could be leveraged to compute \textit{all} FIs and then filtering out only the actual co-movement patterns through a post-processing phase.*

*We leverage PFPGrowth \cite{DBLP:conf/recsys/LiWZZC08}, a well-known implementation of distributed FIM, to compute \textit{all} the (potential) swarm patterns on the entire \sf{Oldenburg} dataset; with respect to SPARE, the computation of FIs is feasible since PFPGrowth does not rely on Apriori enumeration. %\sf{Hermoupolis}.*

*\Cref{fig:pfpgrowth} shows the comparison results.*

*Although both FIs and co-movement patterns decrease by increasing the $mSup$ threshold (as expected, the higher $mSup$ the lower the valid patterns),*

*the number of FIs to post-process remains 5 orders of magnitude higher than the number of co-movement patterns and its extraction requires orders of magnitude longer execution time (as confirmed in \cite{DBLP:journals/tkde/LuccheseOP06}).*

*This makes FIM approaches strongly inefficient.*

As to the settings of RAM and Executors, RAM and Executors do not affect the result (in terms of the generated co-movement patterns) but only the performance. We added the following sentence at the end of the discussion of Table 14 (the former Table 12).

*Note that changing RAM and Executors does not affect the result in terms of the generated co-movement patterns but only the time necessary for their extraction.*

Section 6.4: You performed an effectiveness comparison with an algorithm and changed the benchmark to compare the algorithm for big data performance analysis. This comparison is unfair since the algorithm does not obtain the same results or the same implementation. Could you configure experiments under the same characteristics?

We agree with the reviewer, the comparison was missing the explanation of its settings.

The result of SPARE (now explained in sec 6.5) is not directly comparable with CMT. SPARE treats the temporal and spatial features sequentially. At first, it groups trajectory points by absolute time bins (e.g., 3 Mar 2020, 10:00:00 and 3 Mar 2020, 11:00:00), then it clusters trajectories in each time bin, and, finally, it uses an Apriori-like approach to create the co-movement patterns out of the clusters from different time bins.

This results in the lack of possibility to find co-movement patterns within the same time bin. This makes it impossible, by construction, to have the same result unless the time bin is fine-grained enough to guarantee that no trajectory has more than one location in the same time bin.

With this in mind, we tried to make the comparison as fair as possible by:

- reducing both mCrd and sampling trajectories down to 2000. This is necessary since SPARE follows an Apriori enumeration that is exponential in the number of trajectories in the co-movement patterns. We tested SPARE with the entire Oldenburg dataset, but SPARE failed to compute with 1.000.000 trajectories.
- Reducing the time bin granularity in the order of seconds, so that no trajectory had more than one location in the same time bin. This increased the size of the tessellation to |S|=3422

We added the following paragraph to 6.5

*The points above explain why, by construction, it is unfeasible to have exactly the same co-movement patterns unless the time bin is fine-grained enough to guarantee that no trajectory has more than one location in the same time bin. With this in mind, we tried to make the comparison as fair as possible by:*

*\begin{itemize}*

*\item Narrowing the comparison down to \sf{Oldenburg} by only varying the minimum support since SPARE is limited to an absolute time dimension and cannot handle additional semantic features (\sf{Milan} was too sparse to produce meaningful results).*

*\item Reducing the time bin granularity in the order of seconds so that no trajectory had more than one location in the same time bin. This increased the size of the tessellation to $|S|=3422$.*

*\item Reducing $mCrd$ to 10 and sampling trajectories down to $|T|=2000$. This is necessary since SPARE follows an Apriori enumeration that is exponential in the number of trajectories in the co-movement patterns. We tested SPARE with the entire Oldenburg dataset, but SPARE failed to compute with $10^6$ trajectories.*

*\end{itemize}*

References


[1] Fonseca-Galindo, J. C., de Castro Surita, G., Neto, J. M., de Castro, C. L., & Lemos, A. P. (2022). A multi-agent system for solving the dynamic capacitated vehicle routing problem with stochastic customers using trajectory data mining. Expert Systems with Applications, 195, 116602.

[2] Lv, M., Chen, L., Chen, T., Zeng, D., & Cao, B. (2019). Discovering individual movement patterns from cell-id trajectory data by exploiting handoff features. Information Sciences, 474, 18-32.

[3] Sim, D. G., Kwon, O. K., & Park, R. H. (1999). Object matching algorithms using robust Hausdorff distance measures. IEEE Transactions on image processing, 8(3), 425-429.

[4] Robinson, M. T. (1990). The temporal development of collision cascades in the binary-collision approximation. Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, 48(1-4), 408-413.

[5] Agarwal, P. K., Fox, K., Munagala, K., Nath, A., Pan, J., & Taylor, E. (2018, May). Subtrajectory clustering: Models and algorithms. In Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems (pp. 75-87).

[6] da Silva, T. L. C., Lettich, F., de Macêdo, J. A. F., Zeitouni, K., & Casanova, M. A. (2020, June). Online clustering of trajectories in road networks. In 2020 21st IEEE International Conference on Mobile Data Management (MDM) (pp. 99-108). IEEE.

[7] Li, H., Wang, Y., Zhang, D., Zhang, M., & Chang, E. Y. (2008, October). PFP: parallel fp-growth for query recommendation. In Proceedings of the 2008 ACM conference on Recommender systems (pp. 107-114).

# Reviewer #2

This paper studied mobility pattern mining problem in trajectory data mining. In this article, a unifying approach named Colossal Trajectory Mining (CTM) is proposed to efficiently extract heterogeneous mobility patterns out of a multidimensional space. In addition, coarse-grained solutions and distributed design enables the method to achieve better computational efficiency in big data. The introduction of the method is clear and detailed, and the validity of the algorithm is verified in real datasets. Overall, the article is well-written and easy to follow.

Minor:

1) The pictures are a little rough. For example, in Fig.1, the text tries not to overflow the box; in Fig.2, one of the arrows has different thicknesses

> Fixed it. We enlarged the boxes in Fig 1 to avoid overflows and fixed the arrow in Fig 3.

2) None of the formulas are numbered.

> Fixed it. We number the distance function (it is the only equation/formula in the paper)

3) Most of the experimental results in this paper are presented in the form of tables, and it is suggested to use more intuitive graphics.

> The reason for using tables instead of charts is that we always report two measures with different scales (e.g., Time and Enum) and this would require a double-scale chart that is also complex to be read.

Questions:

1) It is mentioned in the abstract that the algorithm in this paper is better than SOTA, and the specific improvement degree of the method in the relevant evaluation indicators should be given, rather than a qualitative generalization.

> We smoothened the abstract and introduction. Indeed, CTM is novel in its expressiveness. Hence it is not directly comparable with the SOTA algorithm. For instance, when comparing SPARE with a limited version of the original dataset, CTM wins when extracting “colossal” co-movement patterns, but it is complemental to SPARE when small patterns should be found.

2) There is only one comparison algorithm in this paper, which was published in 2016. It is suggested that the author further consult the literature of recent three years to find more comparison algorithms to support your conclusion.

Following the reviewers' concerns, we updated the Related Work section up to 2023 and we also introduced Table 2 to summarize the main differences between approaches related to CTM. The table highlights that there are no direct competitors including all the features of CTM. Note that we do not find as useful comparisons against approaches that are less expressive than CTM (i.e., that do not produce the same results). This is why we only compare against PFPGrowth and SPARE, two big data approaches for the extraction of FIs and co-movement patterns, respectively.

Finally, note that SPARE is still considered a "novel" contribution in the **generic/holistic** extraction of co-movement patterns

- Tritsarolis et al. "Predicting Co-movement patterns in mobility data," GeoInformatica 2022. An approach that defines a new generalized mobility pattern is presented in [9] where the general co-movement pattern is proposed.
- Orkzai et al. "Distributed mining of convoys in large scale datasets," GeoInformatica 2021. The authors propose a generic framework GCMP for mining co-movement patterns and its implementation called the Star Partitioning and ApRiori Enumerator (SPARE) framework. Orkzai et al. provide an end-to-end optimized solution for convoy mining from data partitioning to convoy discovery but do not provide a generalized framework such as SPARE nor deal with any type of feature such as CTM.