k-STARs: Sequences of Spatio-Temporal Association Rules

Florian Verhein
School of Information Technologies, University of Sydney, Australia
fverhein@it.usyd.edu.au

Abstract

A Spatio-Temporal Association Rule (STAR) describes how objects move between regions over time. Since they describe only a single *movement between two regions, it is very difficult to see larger patterns in the dataset by considering only the set of STARs. It is especially difficult on complex datasets where the underlying patterns overlap. At best we will miss important patterns - being unable to "see the forest for the trees", and at worst this can lead to false interpretations. We introduce the k-STAR pattern which describes the sequences of STARs that objects obey. Since a k-STAR captures* sequences *of object movements it solves these problems. We also allow space and time gaps between successive STARs, as well as supporting 'replenishable' k-STARs so we are able to capture the rich set of patterns that exist in real world data. We define two important measures;* min-l-support *and* min-l-confidence *that allow us to achieve the above and present various anti-monotonic and weakly anti-monotonic properties for reducing the search space*[1].

1 Introduction

We consider datasets where many uniquely identifiable objects move throughout different sized regions, such as mobile phone users through the cells of a mobile phone network. Such datasets are composed of a sequence of snapshots of which regions the objects are in – that is, we do not know their precise location. A Spatio-Temporal Association Rule (STAR)[7] tells us how objects move between these regions over time:

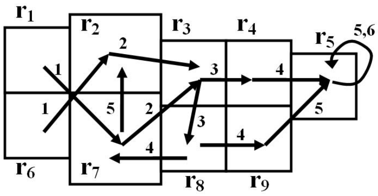
Definition 1 (STAR). $\zeta = (r_a, TI_a) \Rightarrow (r_c, TI_c)$ *Objects appearing in region r_a during time interval TI_a will appear in region r_c during time interval TI_c, where $TI_a < TI_c$ (See Figure 2 for definition of $<$).*

In [7] we were interested in those rules with high support and confidence – that is, where enough objects satisfy the rule (*support*), and where the probability that the rule holds is high enough (*confidence*). Our algorithm, which we call *STARMiner*, scanned the dataset once and efficiently mined all STARs with sufficient support and confidence.

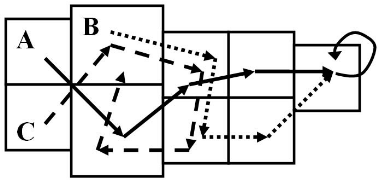
STARs tell us about individual movements that groups of objects make. But this does not tell us anything about their movements beyond the time interval pair $TI' = [TI_a, TI_c]$. Figure 1(a) shows 13 STARs and an indication of the time intervals for which they apply. Each STAR tells us that the movement it describes is interesting, but apart from the time intervals we have no clue where objects go next. We could have many STARs entering and leaving the same region at roughly the same time, so we cannot tell which path to follow – such as in r_4. This is especially confusing when many paths intersect. Furthermore, if there is a significant time delay between when a particular group of objects enter and leave the region, we may be mislead by the time intervals. For example, there is no reason why objects cannot move from r_1 to r_7, spread out into other regions ('go their separate ways'), then converge back into r_7 before moving to r_2 together, rather than immediately moving to r_3 which would be a first assumption. Note that if they merely stayed in r_7 we would have rules telling us this ($r_7 \rightarrow r_7$).

So it should be clear that in a situation such as Figure 1(a), it is very difficult to draw any conclusions about the overall object trajectories and paths – that is, the *sequence* of movements they make – because we cannot *drill up*. Figure 1(b) shows some possible sequences (made by three groups of objects $\{A, B, C\}$) that could have produced Figure 1(a). Unlike the Figure 1(a) there is no ambiguity and it tells us much more useful information that can be

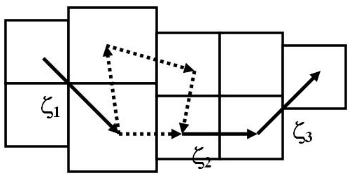
[1] The author's PhD scholarship is partly funded by the Australian Research Council (ARC) Discovery Grant DP0559005.



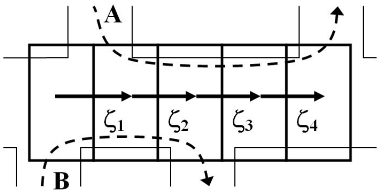
(a) 13 STARs, labeled with a number indicating whether it applies before, after, or at the same time as other STARs.



(b) One possible underlying sequence of movements in (a). k-STARs allow us to mine these sequences, removing the ambiguity. The self-loop tells us that objects remain stationary.



(c) Space and time gaps are allowed, so $\langle\zeta_1, \zeta_2, \zeta_3\rangle$ is mined.



(d) k-STAR 'replenishment' allows us to create $\langle\zeta_1, \zeta_2, \zeta_3, \zeta_4\rangle$ for mining patterns such as roads.

Figure 1. Motivating examples.

used to predict object movements. Finally, we can always *drill down* to subsequences for closer examination. This is the motivation behind *sequences* of STARs – k-STARs: (See Figure 2 for definitions of $TI_*(\cdot)$ and $R_*(\cdot)$).

Definition 2. *A* **k-STAR** *is a* sequence *of STARs* $\Upsilon = \langle\zeta_1, \zeta_2, ..., \zeta_k\rangle : k = 1, 2, ...$ *such that* $TI_C(\zeta_i) \leq TI_A(\zeta_{i+1})$ *with equality only allowed*[2] *when* $R_C(\zeta_i) = R_A(\zeta_{i+1})$. *We say* Υ' *is a* **sub-k-STAR** *of* Υ, *written* $\Upsilon' \sqsubseteq \Upsilon$, *if* $\Upsilon' = \langle\zeta_i, \zeta_{i+1}, ..., \zeta_j\rangle : 1 \leq i \leq j \leq k$. *That is, a* subsequence with no gaps.

Note that this definition is quite broad – we allow quite general patterns that we think are important for studying object mobility datasets:

For $\zeta = (r_a, TI_a) \Rightarrow (r_c, TI_c)$: $TI_A(\zeta) = TI_a$, $TI_C(\zeta) = TI_c$, $R_A(\zeta) = r_a$, $R_C(\zeta) = r_c$ and $TI(\zeta) = [TI_A(\zeta), TI_A(\zeta)]$.
TI_i are fixed width time intervals. Define $TI_i < TI_j$ if TI_i occurs before TI_j and they do not overlap.
$O(r, TI)$ is the set of objects making an appearance in region r during TI. $O_A(\zeta) = O\left(R_A(\zeta), TI_A(\zeta)\right)$ $O_C(\zeta) = O\left(R_C(\zeta), TI_C(\zeta)\right)$ $O(\zeta) = O_A(\zeta) \cap O_C(\zeta)$, the objects that follow ζ.
The *support* of ζ is $\sigma(\zeta) = \lvert O(\zeta)\rvert$. The *support* of region r during TI is $\sigma(r, TI) = \lvert O(r, TI)\rvert$.
The *confidence* of ζ is $c(\zeta) = \sigma(\zeta)/\lvert O_A(\zeta)\rvert$ – the fraction of objects that followed the rule, given that they were in the antecedent of the rule. It is an estimate of $P(o \in O_C(\zeta) \mid o \in O_A(\zeta))$.
For $\Upsilon = \langle\zeta_1, \zeta_2, ..., \zeta_k\rangle$ of length $\lvert\Upsilon\rvert = k$: $\sigma_{i,l} = \left\lvert \cap_{j=i}^{i+l-1} O(\zeta_j)\right\rvert$, the *support* of the sub-k-STAR $\langle\zeta_i, ..., \zeta_{i+l-1}\rangle$ of length l. $c_{i,l} = \sigma_{i,l}/\lvert O_A(\zeta_i)\rvert$, the *confidence* of the sub-k-STAR $\langle\zeta_i, ..., \zeta_{i+l-1}\rangle$ of length l.
$l_\sigma(\Upsilon) = \{l : \sigma(\Upsilon)_l^{min} \geq minSup\}$ $l_c(\Upsilon) = \{l : c(\Upsilon)_l^{min} \geq minConf\}$ (The sets of l for which Υ is min-l-frequent/confident). $l_{\sigma max}(\Upsilon) = \max(l_\sigma(\Upsilon))$

Figure 2. Definitions not in the text

First, we allow space and time gaps between individual STARs. Figure 1(c) shows three STARs that are considered interesting (ζ_1, ζ_2 and ζ_3) and some paths that objects follow, but with insufficient support and confidence to be of interest. Specifically, objects move along ζ_1, then some follow the top path while others follow the bottom path, before they merge again to follow ζ_2 and ζ_3. The sequence $\langle\zeta_1, \zeta_2, \zeta_3\rangle$ accurately describes this pattern, but if we did not allow space gaps (ie: $R_C(\zeta_1) \neq R_A(\zeta_2)$) or time gaps (ie: $TI_C(\zeta_1) < TI_A(\zeta_2)$) we would miss it[3]. The sequence $\langle\zeta_1, \zeta_2, \zeta_3\rangle$ is also different to mining $\langle\zeta_1, \zeta_2\rangle$ and ζ_3 separately, which would tell us that not enough (or indeed any) objects travel the entire sequence.

Secondly, we allow limited 'replenishment' of patterns. This is useful because it allows us to mine patterns that are supported by many objects, but they might not all travel down the *entire* length of the sequence. Consider Figure 1(d) which shows two groups of objects (A and B). Enough objects travel the paths to ensure that each of $\zeta_1, ..., \zeta_4$ are interesting, but none travel the entire sequence. The se-

[2] $R_C(\zeta_i) \neq R_A(\zeta_{i+1})$ but $TI_C(\zeta_i) = TI_A(\zeta_{i+1})$ makes no sense as objects cannot be in two places at once

[3] Note that we *cannot* consider the k-STAR as a sequence of regions and time pairs as this cannot express space or time gaps.

quences $\langle \zeta_1, \zeta_2 \rangle$ and $\langle \zeta_2, \zeta_3, \zeta_4 \rangle$ are therefore interesting. But what about $\Upsilon = \langle \zeta_1, \zeta_2, \zeta_3, \zeta_4 \rangle$? Since all *consecutive and overlapping* subsequences of Υ of length 2 are interesting ($\langle \zeta_1, \zeta_2 \rangle$, $\langle \zeta_2, \zeta_3 \rangle$, $\langle \zeta_3, \zeta_4 \rangle$) we say (for now[4]) that Υ is "interesting at level $l = 2$". We do this because there are many situations where such a pattern is useful, such as finding roads and highways. Not all objects travel their entire length, so we would not get a k-STAR that represents them if we do not allow this 'replenishment'. Secondly, the pattern gives a high level view of the objects motion, and the levels (there may be more than one) at which the sequence is interesting tells us a lot about the behaviour of the objects. If we want more detail we can always *drill down* the lattice defined by these k-STARs to explore them in more detail. The overlap of consecutive sequences ensures we find only 'real' sequences, not just an arbitrary concatenation of sequences. The above discussion leads to two measures of 'interestingness' that apply to k-STARS: *min-l-support* and *min-l-confidence.*

Finally, note that ζ may have $R_A(\zeta) = R_C(\zeta)$ which allows us to express sequences that includ the scenario when the objects remain still. There is an example of this in Figure 1(a) and 1(b). Note that this is different to the time gap scenario, which tells us that objects did *not* remain stationary, but did not generate any interesting movements either.

Our **contributions** are as follows:

1. We define k-STARs – a new type of pattern for mining object mobility databases that is flexible enough to mine a wide variety of sequences that are interesting in real world applications.
2. We outline the properties of two novel measures supporting the above for evaluating the interestingness of k-STARs: *min-l-support* and *min-l-confidence*. The Lemmas that we present enable efficient mining of k-STARs. We also describe a *lattice* on all k-STARs that allows us to *drill up* and *drill down* for exploratory data mining. This greatly improves the ability to interpret the many patterns that are found.

2 Related Work

We considered STARs and other interesting patterns in [7]. Interesting work that deals with spatio-temporal patterns in the spatio-temporal domain (but are not association type patterns) include [5, 3, 4, 2]. Except for [2], none of these consider sequences. Traditional temporal sequence mining [1] does not address the issues of spatio-temporal data and it is not possible to map k-STARs into these sequence mining algorithms.

Cao et al. [2] is the most relevant work. They consider the mining of frequent sequences of line segments that approximate an object's movements over time. Since they consider strings of (x, y, t) coordinates, they cannot mine patterns where there is a space or time gap because their pattern cannot express this type of behaviour. Their patterns are also fundamentally different to ours; we consider a set of region while [2] use the object coordinates, we mine patterns supported by *many* objects, while [2] mine recurring patterns of the *same* object. The ability to mine space and time gaps, as well as replenishable patterns, k-STAR lattice and min-l-support and min-l-confidence measures also distinguishes our work.

3 k-STARs

Based on the discussion in Section 1, our **problem definition** is to mine all[5] possible k-STARs Υ with *min-l-support* and *min-l-confidence* (which we introduce shortly) above $minSup$ and $minConf$ thresholds and the possible l for which these hold. The desired l are sets $l_\sigma(\Upsilon)$ and $l_c(\Upsilon)$ (which we discuss later), and we have the additional constraint that $\exists l, l' : l \in l_\sigma(\Upsilon) \wedge l' \in l_c(\Upsilon) \wedge l, l' \geq minL$ (another threshold). That is, Υ must be min-l-confident and min-l-frequent for at least one $l \geq minL$.

Motivated by the discussion in Section 1, we are interested only in Υ where *each* sub-k-STAR of length $l \geq minL$ is frequent or confident. By frequent (confident) we mean 'has support (confidence) above threshold $minSup$ ($minConf$)'. The reader may find Figure 3(a) useful, and should refer to Figure 2 for the relevant notation. In the following let $\Upsilon_k = \langle \zeta_1, \zeta_2, ..., \zeta_k \rangle$ so $|\Upsilon_k| = k$.

Definition 3. *The* **min-l-support** *of* Υ_k *is* $\sigma(\Upsilon_k)_l^{min} = \min_{i \in \{1,...,k-l+1\}} \sigma_{i,l}$, $1 \leq l \leq k$.

Definition 4. *The* **min-l-confidence** *of* Υ_k *is* $c(\Upsilon_k)_l^{min} = \min_{i \in \{1,...,k-l+1\}} c_{i,l}$, $1 \leq l \leq k$, *an estimate of* $\min_{i \in \{1,...,k-l+1\}} P\left(\cap_{j=i}^{i+l-1} O(\zeta_j)\,|O_A(\zeta_i)\right)$.

That is, it is the minimum support (confidence) of all the $k - l + 1$ *sub-k-STARs* of length l. This means

[4] We define our interestingness measures in Section 3.

[5] However, for practical purposes it makes sense to impose a user defined upper limit, $maxDelay$, and a user defined *Neighbourhood function* $N(r, t)$ which limit the time and space gaps. Finally, we have two more optional parameters, $maxD$ and $minK$. These are all described in [6].

that if we are given a k-STAR with $\sigma_{min}(\Upsilon_k)_l = \beta$ and $c_{min}(\Upsilon_k)_l = \alpha$, we can say that a) for any region and time specified by the antecedents of the first $k - l + 1$ ζ_i, at least β objects follow Υ_k for at least l ζ's and b) any object appearing in any of the regions at times specified by the antecedents of the first $k - l + 1$ ζ_i will follow Υ_k for at least l ζ's with probability[6] at least α. This is very useful (and indeed required) for making use of such rules. It should be clear that $l < k$ allows the 'replenishment' and associated patterns discussed in Section 1, and $minL$ restricts the possible replenishment.

In the remainder of this paper we present a number of useful properties. The proofs are available in [6]. Let $\Upsilon_k = \langle\zeta_1, \zeta_2, ..., \zeta_k\rangle$ and $\Upsilon_{k+1} = \langle\zeta_1, \zeta_2, ..., \zeta_k, \zeta_{k+1}\rangle$ or $\Upsilon_{k+1} = \langle\zeta_{k+1}, \zeta_1, \zeta_2, ..., \zeta_k\rangle$. That is, we add the extra ζ to either the front or the back of Υ_k.

Fact 1. $\sigma(\Upsilon_k)_l^{min} \geq \sigma(\Upsilon_{k+1})_l^{min}$. *That is, min-l-support is anti-monotonic in k.*

Lemma 1. $\sigma(\Upsilon_k)_l^{min} \geq \sigma(\Upsilon_k)_{l+1}^{min}$. *That is, min-l-support is anti-monotonic in l.*

These follow from the fact that $\sigma_{j,l} \geq \sigma_{i,l+1} \forall j : \langle\zeta_j, ..., \zeta_{j+l-1}\rangle \subset \langle\zeta_i, ..., \zeta_{i+l}\rangle$. In summary we have $\sigma(\Upsilon_k)_l^{min} \geq \sigma(\Upsilon_k)_{l+1}^{min} \geq \sigma(\Upsilon_{k+1})_{l+1}^{min}$. This means that if $\Upsilon_k = \langle\zeta_1, ..., \zeta_k\rangle$ is l_1-frequent, then it is l-frequent for all $l \leq l_1$. Furthermore, any sub-k-STAR $\Upsilon' = \langle\zeta_i, ..., \zeta_j\rangle : 1 \leq i \leq j \leq k$ is also *at least* min-l_1-frequent.

Fact 2. $c(\Upsilon_k)_l^{min} \geq c(\Upsilon_{k+1})_l^{min}$. *That is, min-l-confidence is anti-monotonic in k.*

Lemma 2. $c(\Upsilon_k)_l^{min} \geq c(\Upsilon_k)_{l+1}^{min}$, $l < k$ *if and only if* $\exists i \in \{1, ..., k-l\} : c_{i,l+1} \leq c_{k-l+1,l}$. *That is,* $c(\Upsilon_k)_l^{min} < c(\Upsilon_k)_{l+1}^{min}$ *if and only if* $c_{i,l+1} > c_{k-l+1,l} \forall i \in \{1, ..., k-l\}$. *We say that min-l-confidence is weakly anti-monotonic in l.*

The term *weakly anti-monotonic* means it is anti-monotonic in all except perhaps *one* sub-k-STAR. To understand why this is the case first note that the denominator of each $c_{i,l}$ is the same as that of $c_{i,l+1}$ for all $i \in \{1, .., k-l\}$. Therefore, $c_{i,l} \geq c_{i,l+1}$ since $\sigma_{i,l} \geq \sigma_{i,l+1}$. However is is *not* always true that $c_{i,l} \geq c_{i-1,l+1}$, since the denominators are no longer the same. Secondly, if we increment l we 'loose' one sub-k-STAR, as illustrated in Figure 3(a) - namely, $c_{k-l+1,l+1}$ does not exist. That is, there is one less *sub-k-STAR* of length $l + 1$ than there is of length l. We say $c_{k-l+1,l}$ is 'lost' if we increment l. Now if $c_{k-l+1,l} < c_{i,l} \forall i \in \{1, .., k-l\}$ then it is *possible* (the Lemma tells us when) that min-l-confidence increases when we increment l, as we loose the minimum term, so to speak.

[6] All probabilities mentioned are estimated relative to the total number of objects, N. eg: $P(O_A(\zeta_i)) = |O_A(\zeta_i)| / N$.

To **mine k-STARs**, we leverage these anti-monotonic and weak-anti-monotonic properties. We grow the k-STARs from shorter k-STARs by joining them together, exploiting the lemmas in the remainder of this section.

Refer to Figure 2 for the definitions of $l_\sigma(\Upsilon)$, $l_c(\Upsilon)$ and $l_{\sigma max}(\Upsilon)$. Since *min-l-support* is anti-monotonic in l (Lemma 1), $l_\sigma(\Upsilon)$ will always have the form $\{1, 2, 3, ...l_{\sigma max}(\Upsilon)\}$. $l_c(\Upsilon)$ on the other hand may have gaps as it is only weakly anti-monotonic (Lemma 2).

In the following lemmas, Let $\Upsilon_\alpha = \langle\zeta_1, ..., \zeta_m\rangle$ and $\Upsilon_\beta = \langle\zeta_{m+1}, ..., \zeta_{m+n}\rangle$ be non-overlapping and $TI_C(\zeta_m) \leq TI_A(\zeta_{m+1})$ so that $\Upsilon = \Upsilon_\alpha \cup \Upsilon_\beta = \langle\zeta_1, ..., \zeta_m, \zeta_{m+1}, ..., \zeta_{m+n}\rangle$ is a valid k-STAR.

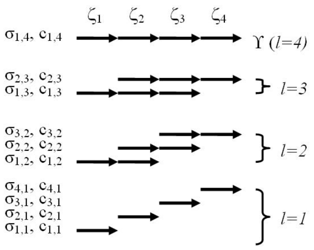
Lemma 3. Joining k-STARs for min-l-support: $l_{\sigma max}(\Upsilon) \leq l^u_{\sigma max}(\Upsilon_\alpha, \Upsilon_\beta)$ *where* $l^u_{\sigma max}(\Upsilon_\alpha, \Upsilon_\beta)$ *is given in Figure 3(b).*

Since *min-l-confidence* is only weakly anti-monotonic in l, it is more complicated.

Lemma 4. Joining k-STARs for min-l-confidence: $l_c(\Upsilon) \subseteq l^u_c(\Upsilon_\alpha, \Upsilon_\beta)$ *where* $l^u_c(\Upsilon_\alpha, \Upsilon_\beta)$ *is given in Figure 3(b).*

The above lemmas reduce the search spaces for $l_{\sigma max}(\Upsilon)$ and $l_c(\Upsilon)$. In the case of Lemma 3, $l^u_{\sigma max}(\Upsilon_\alpha, \Upsilon_\beta)$ is an upper-bound on $l_{\sigma max}(\Upsilon)$ while in Lemma 4, $l^u_c(\Upsilon_\alpha, \Upsilon_\beta)$ is a superset of $l_c(\Upsilon)$. The following lemmas give us a quick way to perform the search in the remaining spaces $l_{\sigma max}(\Upsilon) \leq l^u_{\sigma max}(\Upsilon_\alpha, \Upsilon_\beta)$ and $l_c(\Upsilon) \subseteq l^u_c(\Upsilon_\alpha, \Upsilon_\beta)$ to find the desired $l_{\sigma max}(\Upsilon)$ and $l_c(\Upsilon)$. Clearly, if $l^u_\sigma(\Upsilon_\alpha, \Upsilon_\beta) = 1$ then $l_{\sigma max}(\Upsilon) = 1$ and if $l^u_c(\Upsilon_\alpha, \Upsilon_\beta) = \{1\}$ then $l_c(\Upsilon) = \{1\}$. To test whether $\Upsilon = \Upsilon_\alpha \cup \Upsilon_\beta$ is min-l-frequent (min-l-confident) when $l > 1$ and for all $l \leq l^u_{\sigma max}(\Upsilon_\alpha, \Upsilon_\beta)$ ($l \in l^u_c(\Upsilon_\alpha, \Upsilon_\beta)$) we need to evaluate $\sigma(\delta_{l,j})_l^{min}$ ($c(\delta_{l,j})_l^{min}$) for all j and l where $\delta_{l,j}$ is the jth sub-k-STAR of Υ of length l that *overlaps* both Υ_α and Υ_β. For example, if $\Upsilon_\alpha = \langle\zeta_1, \zeta_2, \zeta_3\rangle$ and $\Upsilon_\beta = \langle\zeta_4, \zeta_5, \zeta_6\rangle$ then $\delta_{3,1} = \langle\zeta_2, \zeta_3, \zeta_4\rangle$ and $\delta_{3,2} = \langle\zeta_3, \zeta_4, \zeta_5\rangle$ are the $\delta_{3,j}$ we need to check.

Lemma 5. Υ *is min-l-frequent* ($l > 1$) *if* $\min_j \left(\sigma(\Upsilon_\alpha)_l^{min}, \sigma(\delta_{l,j})_l^{min}, \sigma(\Upsilon_\beta)_l^{min}\right) \geq minSup$.



(a) All 10 *sub-k-STARs* of $\Upsilon = \langle \zeta_1, \zeta_2, \zeta_3, \zeta_4 \rangle$. $\sigma_{i,l}$ and $c_{i,l}$ are defined in Figure 2.

$$l^u_{\sigma max}(\Upsilon_\alpha, \Upsilon_\beta) \doteq \begin{cases} l_{\sigma max}(\Upsilon_\alpha) + l_{\sigma max}(\Upsilon_\beta) & l_{\sigma max}(\Upsilon_\alpha) = m \, and \, l_{\sigma max}(\Upsilon_\beta) = n \quad (1) \\ l_{\sigma max}(\Upsilon_\alpha) & if \;\; l_{\sigma max}(\Upsilon_\alpha) < m \, and \, l_{\sigma max}(\Upsilon_\beta) = n \quad (2) \\ l_{\sigma max}(\Upsilon_\beta) & l_{\sigma max}(\Upsilon_\alpha) = m \, and \, l_{\sigma max}(\Upsilon_\beta) < n \quad (3) \\ \min(l_{\sigma max}(\Upsilon_\alpha), l_{\sigma max}(\Upsilon_\alpha)) & l_{\sigma max}(\Upsilon_\alpha) < m \, and \, l_{\sigma max}(\Upsilon_\beta) < n \quad (4) \end{cases}$$

$$l^u_c(\Upsilon_\alpha, \Upsilon_\beta) \doteq \begin{cases} l_c(\Upsilon_\alpha) \cap (l_c(\Upsilon_\beta) \cup \{n+1, ..., m\}) \cup \{m+1, ..., m+n\} & l_{cmax}(\Upsilon_\alpha) = m > n \quad (1) \\ l_c(\Upsilon_\alpha) \cap l_c(\Upsilon_\beta) \cup \{m+1, ..., m+n\} & l_{cmax}(\Upsilon_\alpha) = m = n \quad (2) \\ l_c(\Upsilon_\beta) \cap (l_c(\Upsilon_\alpha) \cup \{m+1, ..., n\}) \cup \{n+1, ..., n+m\} & if \;\; l_{cmax}(\Upsilon_\alpha) = m < n \quad (3) \\ l_c(\Upsilon_\alpha) \cap (l_c(\Upsilon_\beta) \cup \{n+1, ..., m\}) & l_{cmax}(\Upsilon_\alpha) < m > n \quad (4) \\ l_c(\Upsilon_\alpha) \cap l_c(\Upsilon_\beta) & l_{cmax}(\Upsilon_\alpha) < m \leq n \quad (5) \end{cases}$$

(b) Equations for Lemmas 3 and 4

Figure 3. Example and Equations.

Lemma 6. *Υ is min-l-confident ($l > 1$) if* $\min_j \left(c(\Upsilon_\alpha)^{min}_l, \; c(\delta_{l,j})^{min}_l, \; c(\Upsilon_\beta)^{min}_l \right) \geq minConf$ *and* $l \in l^u_c(\Upsilon_\alpha, \Upsilon_\beta)$.

In both Lemmas, if any $x(\Upsilon_Y)^{min}_l$ ($x \in \{\sigma, c\}, Y \in \{\alpha, \beta\}$) don't exist (this happens when $l > |\Upsilon_Y|$), they are removed from the calculation. Since $l \leq |\Upsilon_\alpha| + |\Upsilon_\beta|$, this will happen.

Note that the δ are all that we need to calculate, as we know the $\sigma(\Upsilon_\alpha)^{min}_l$, $\sigma(\Upsilon_\beta)^{min}_l$, $c(\Upsilon_\alpha)^{min}_l$ and $c(\Upsilon_\beta)^{min}_l$ we need for Lemmas 5 and 6 already. This is because we know the $l_\sigma(\Upsilon_\alpha)$, $l_\sigma(\Upsilon_\beta)$, $l_c(\Upsilon_\alpha)$, $l_c(\Upsilon_\beta)$ and their respective *min-l-confidences* and *min-l-supports* since we have already mined Υ_α and Υ_β (recall we mine k-STARs by joining smaller k-STARs together). Similarly, we have also generated *all* the new sub-k-STARs of Υ that are now possible. Specifically, all the δs that are confident or frequent are valid k-STARs. So it should be clear that this procedure not only creates $\Upsilon = \Upsilon_\alpha \cup \Upsilon_\beta$, but also all new sub-k-STARs of Υ. Specifically, we join all *suffixes* of Υ_α to all *prefixes* of Υ_β. Other sub-k-STARs of Υ_α and Υ_β already exist as Υ_α and Υ_β exist.

In our *k-STARMiner* **algorithm**, we join new ζ to existing k-STARs as they are mined by *STARMiner* [7]. That is, $|\Upsilon_\beta| = 1$. We use a number of novel datastructures to find potential Υ_α to join onto, and to perform the join efficiently. Details may be found in [6].

Finally, we define the **k-STAR lattice** as linking all k-STARs Υ and Υ' if and only if $\Upsilon' \sqsubset \Upsilon$ and either $|\Upsilon'| \in l_\sigma(\Upsilon)$ or $|\Upsilon'| \in l_c(\Upsilon)$ (or both). Each node in this lattice is a k-STAR and holds the min-l-confidence and min-l-support value. Using this lattice we can drill down or up through the sequences for exploratory data mining.

4 Conclusion

We described *sequences* of Spatio-Temporal Association Rules (k-STARs) that greatly enhance the interpretability and power of STAR mining. k-STARs can capture interesting 'replenishable' patterns such as 'roads' and space and time gaps. A lattice defined over the sub-k-STARs enables us to *drill down* or *drill up* to explore the results. The introduction of the *min-l-support* and *min-l-confidence* measures allow us to do these things and we presented properties required for mining k-STARs efficiently.

References

[1] R. Agrawal and R. Srikant. Mining sequential patterns. In *Eleventh International Conference on Data Engineering*, pages 3–14, Taipei, Taiwan, 1995. IEEE Computer Society Press.

[2] H. Cao, N. Mamoulis, and D. Cheung. Mining frequent spatio-temporal sequential patterns. In *Fifth IEEE Conference on Data Mining (ICDM'05)*, 2005.

[3] Y. Ishikawa, Y. Tsukamoto, and H. Kitagawa. Extracting mobility statistics from indexed spatio-temporal datasets. In *STDBM*, pages 9–16, 2004.

[4] N. Mamoulis, H. Cao, G. Kollios, M. Hadjieleftheriou, Y. Tao, and D. W. Cheung. Mining, indexing, and querying historical spatiotemporal data. In *KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 236–245. ACM Press, 2004.

[5] I. Tsoukatos and D. Gunopulos. Efficient mining of spatiotemporal patterns. In *SSTD '01: Proceedings of the 7th International Symposium on Advances in Spatial and Temporal Databases*, pages 425–442, London, UK, 2001. Springer-Verlag.

[6] F. Verhein. k-stars: Sequences of spatio-temporal association rules (tr 589). Technical report, School of IT, University of Sydney, NSW, Australia, 2006.

[7] F. Verhein and S. Chawla. Mining spatio-temporal association rules, sources, sinks, stationary regions and thoroughfares in object mobility databases. In *The 11th International Conference on Database Systems for Advanced Applications (DASFAA'06)*, 2006.