Generalização de Mineração de Sequências Restritas no Espaço e no Tempo*

Antonio Castro[1], Heraldo Borges[1], Ricardo Campisano[1]
Esther Pacitti[2], Fabio Porto[3], Rafaelli Coutinho[1], Eduardo Ogasawara[1]

[1]CEFET/RJ - Centro Federal de Educação Tecnológica Celso Suckow da Fonseca

[2]INRIA, University of Montpellier

[3]LNCC, Laboratório Nacional de Computação Científica

{antonio.castro, hborges, rcampisano}@eic.cefet-rj.br

esther.pacitti@lirmm.fr, fporto@lncc.br

rafaelli.coutinho@cefet-rj.br, eogasawara@ieee.org

Abstract. *Spatiotemporal patterns bring knowledge of sequences of events, place and time when they occur. Finding such patterns is a complex task and one of great value for different domains. However, not all patterns are frequent across an entire dataset, often occurring in restricted space and time. This work formalizes the Mining of Restricted Sequences in Space and Time, without the use of previous restrictions of time and space, allowing different sequence sizes, time intervals and space (in three dimensions) to present such patterns. It also brings validation with a tested implementation on a real seismic dataset. Resulting in a sensitivity analysis and evaluation of the use of resources that indicate the validity and feasibility of the solution.*

Resumo. *Padrões espaço-temporais trazem conhecimento de sequências de eventos, local e momento em que ocorrem. Encontrar tais padrões é uma tarefa complexa e de grande valor para diferentes domínios. No entanto, nem todos os padrões são frequentes em todo um conjunto de dados, ocorrendo com frequência em espaço e tempo restritos. Este trabalho formaliza a Mineração de Sequências Restritas no Espaço e no Tempo, sem o uso de limiares de restrição para tempo e espaço. Isso permite que diferentes tamanhos de sequências, intervalos de tempo e espaço tridimensional apresentem tais padrões. Traz também validação com uma implementação testada sobre um conjunto de dados sísmico real. Tendo como resultado uma análise de sensibilidade e avaliação do uso de recursos que indicam a validade e viabilidade da solução.*

1. Introdução

A evolução tecnológica torna cada vez mais comum o acesso a dispositivos digitais providos de sensores e GPS. A partir deles surgem extensos conjuntos de dados espaço-temporais. A descoberta e análise de padrões em meio a estes conjuntos de dados passa a ser um diferencial importante [Huang et al., 2008].

*Os autores agradecem à FAPERJ, à CAPES (código 001) e ao CNPq pelo financiamento do projeto.

Algoritmos de mineração de dados têm sido aplicados na descoberta de padrões em uma grande diversidade de problemas, em especial a mineração de sequências no espaço e no tempo tem se tornado importante para diversos domínios [Alatrista-Salas et al., 2016; Li and Fu, 2014; Huang et al., 2008]. No entanto, nem sempre a frequência de ocorrência desses padrões é alta por todo o conjunto de dados. Surge, então, a demanda por descobrir padrões que sejam frequentes não por todo um conjunto de dados, mas em espaço e tempo restrito.

Há diferentes métodos para mineração de dados espaço-temporais. Alguns utilizam apenas mineração de dados na busca por padrões frequentes, levando em conta somente o tempo [Li and Fu, 2014]. Outros combinam técnicas buscando padrões frequentes no tempo e posteriormente os agrupam no espaço [Flamand et al., 2014]. Além disso, há uma diversidade na forma com que se lida com as restrições. Alguns usam um suporte global, um valor de suporte que é válido para todo o conjunto de dados [Alatrista-Salas et al., 2016]. Outros consideram um suporte local, fazendo uso de janelas pré-definidas de tempo e de espaço [Koseoglu et al., 2020].

O presente trabalho difere ao buscar sequências frequentes no tempo que ocorrem em grupos espaciais. Ao invés de usar limiares de restrição para tempo e espaço, estabelece-se três parâmetros de densidade: uma frequência mínima a ser alcançada no período, uma distância máxima que uma posição pode estar de alguma outra no grupo e um limite inferior de posições distintas no grupo. Assim, a formalização apresentada neste trabalho é capaz de encontrar diferentes tamanhos de sequências, intervalos de tempo e regiões do espaço onde uma sequência é frequente.

Até onde alcançaram as pesquisas realizadas, o único trabalho com abordagem semelhante encontrado na literatura é o proposto por Campisano et al. [2018]. Tal trabalho busca por sequências frequentes, mas considera o espaço de forma linear. O presente trabalho é uma generalização que apresenta uma formalização que considera o espaço na sua forma tridimensional. Em contrapartida, a busca pelas sequências é feita em tempo e espaço integrado, mas em ordem diferente da apresentada por Campisano et al. [2018].

2. Sequências espaço-temporais

Uma **sequência com marcação de tempo (TS)** é uma sequência ordenada de observações obtidas por meio de medições repetidas ao longo do tempo. Seja $t = <v_1, v_2, \ldots, v_n>$ uma TS, onde v_i é um item, $|t| = n$ é o número de itens em t, e v_n é o item mais recente em t. Uma **subsequência** é uma amostra contínua de uma TS t com um comprimento definido m que começa em uma marcação de tempo p é uma sequência ordenada de itens representada por: $sub_{m,p}(t) = <v_p, v_{p+1}, \ldots, v_{p+m-1}>$, onde $|sub_{m,p}(t)| = m$ e $1 \leq p \leq |t| - m$. Uma **sequência** $s = <w_1, w_2, \ldots, w_k>$ está incluída em uma TS $t = <v_1, v_2, \ldots, v_n>$, se existir uma posição inicial q tal que $w_1 = v_q, w_2 = v_{q+1}, \ldots, w_k = v_{q+k-1}$. Assim, uma sequência s é definida por: $s = <w_1, w_2, \ldots, w_k>$, $\exists q \mid s = sub_{k,q}(t)$, onde $|s| = k$ [Campisano et al., 2018].

Uma **posição** p é definida como um trio ordenado (x, y, z), onde x, y e z indicam valores das coordenadas no sistema Cartesiano. Sejam f e h duas posições, tais que $f = (x_f, y_f, z_f)$ e $h = (x_h, y_h, z_h)$. A distância entre f e h, denotada por $dist(f, h)$, é calculada usando a distância euclidiana: $dist(f, h) = \sqrt{(x_h - x_f)^2 + (y_h - y_f)^2 + (z_h - z_f)^2}$.

Seja $P = \{p_1, p_2, ..., p_m\}$ um conjunto de posições, uma **sequência com marcação de tempo e espaço (STS)** st é uma dupla (p, t), onde $p \in P$ é uma posição e t é a TS associada. Desta forma, um conjunto de dados de STS D é um conjunto de STS. Diz-se que uma STS $st = (p, t)$ suporta uma sequência s, se s é uma subsequência em t: $sup(s, st) = |Q|, \forall q \in Q \mid s = sub_{|s|,q}(st.t)$. O **suporte** de uma sequência s em D é o número de marcações de tempo em D em que s está incluído, denotado por: $sup(s, D) = |Q|, \forall q \in Q, \exists st_i \in D \mid s = sub_{|s|,q}(st_i.t)$, onde Q é o conjunto de marcações de tempo da sequência s em D.

A frequência de uma sequência s em uma STS st é a fração de $st.t$ que apresenta suporte s: $freq(s, st) = \frac{sup(s,st)}{|st.t|}$. Da mesma forma, a **frequência** de uma sequência s em D é a fração de tempo em D que suporta s, representada por: $freq(s, D) = \frac{sup(s,D)}{|st.t|}$, $st \in D$, assumindo que $|st.t|$ é o mesmo em todas as STS. Dado um valor mínimo definido pelo usuário $\gamma \in]0, 1]$, uma sequência é dita frequente, se $freq(s, D) \geq \gamma$ [Saleh and Masseglia, 2008].

Um **período** $r = (r_s, r_e)$ é definido por uma marcação de tempo inicial r_s e uma marcação de tempo final r_e. O tamanho do período r é dado por: $|r| = r_e - r_s + 1$. Tem-se que PR é o conjunto de todos os possíveis períodos sobre o conjunto de dados D.

3. Formalização

Considerando um conjunto de dados STS D, o problema abordado neste trabalho é encontrar pares de sequências e de janelas espaço-temporais nas quais tais sequências sejam frequentes em D. Esta seção apresenta a formalização para resolução deste problema.

Um **grupo de posições** (por simplicidade **grupo**) g é definido por um conjunto de posições onde seus elementos devem estar a uma distância máxima σ de ao menos um outro elemento do mesmo grupo, ou seja: $g \mid \forall p \in g, \exists q \in g \mid dist(p, q) \leq \sigma$. P é o conjunto de todas as posições. PG é o conjunto de todos os possíveis grupos de posições sobre o conjunto de dados D. O conjunto de STS de um grupo g é definido por: $sts(g) = SG \mid \forall st \in SG, st.p \in g$.

Um ***Ranged Group* (RG)** rg é um trio (s, r, g), onde s é uma sequência, r é um período e g é um grupo. As ocorrências de uma sequência s em um RG rg, definido por $occur(s, r, g)$, referem-se ao número de todas as ocorrências de s no intervalo r em $sts(g)$. O suporte de uma sequência s em um RG rg, denotado por $sup(s, r, g)$, é o número de marcações de tempo em que s começa tendo intervalo r em $sts(g)$, ou seja: $sup(s, r, g) = |Q|, \forall q \in Q, \exists st \in sts(g) \mid s = sub_{|s|,q}(st.t), r_s \leq q \leq r_e, |s| \leq r_e$. A frequência de uma sequência s em um RG rg, $freq(s, r, g)$, é a divisão do suporte do RG $sup(s, r, g)$ pelo tamanho de r: $freq(s, r, g) = \frac{sup(s,r,g)}{|r|}$.

Dados os limites mínimos, definidos pelo usuário, para frequência γ e para tamanho do grupo β, as características de um ***Kernel Range-Group* (KRG)** e de um ***Solid Range-Group* (SRG)** são apresentadas nas Definições 1 e 2, respectivamente.

Definição 1 *Um RG $rg = (s, r, g)$ é chamado de KRG se e somente se atender a:*

1. $freq(s, r, g) \geq \gamma$, a frequência é maior ou igual a frequência mínima γ definida pelo usuário.
2. $|g| \geq \beta$, o grupo g deve respeitar o tamanho mínimo β definido pelo usuário.
3. $\forall r' \in PR \mid r' \subset r$ e $r'.r_s = r.r_s$, ambas as condições se aplicam:
   (a) $sup(s, r', g) < sup(s, r, g)$

(b) $freq(s, r', g) \geq \gamma$

Diminuir o período mantém uma frequência maior que o mínimo, mas diminui o suporte. Tem-se que o tamanho de r é mínimo entre os intervalos que começam na mesma marcação de tempo.

4. $\forall g' \in PG \mid g \subseteq g', occur(s, r, g') = occur(s, r, g)$, aumentar o grupo mantém o mesmo número de ocorrências. Tal condição garante que o tamanho de g é máximo.
5. $\forall g' \in PG \mid g' \subset g, occur(s, r, g') < occur(s, r, g)$, diminuir o grupo reduz o número de ocorrências. Tal condição garante que o tamanho de g é mínimo.

Definição 2 *Um RG $rg = (s, r, g)$ é chamado de SRG se e somente se atender a:*

1. $freq(s, r, g) \geq \gamma$, a frequência é maior ou igual a frequência mínima γ definida pelo usuário.
2. $|g| \geq \beta$, o grupo deve respeitar o tamanho mínimo β definido pelo usuário.
3. $\forall r' \in PR | r \subseteq r'$, é possível ter a) ou b) ou ambas:
   (a) $sup(s, r', g) = sup(s, r, g)$
   (b) $freq(s, r', g) < \gamma$

   Não adianta aumentar o período, pois o suporte é mantido, mas pode reduzir a frequência a um valor menor que a mínima definida pelo usuário. Tal condição garante que o período r é máximo.
4. $\forall r' \in PR \mid r' \subset r, sup(s, r', g) < sup(s, r, g)$, diminuir o período diminui o suporte. Tal condição garante que o tamanho do período r é mínimo.
5. $\forall g' \in PG \mid g \subseteq g', occur(s, r, g') = occur(s, r, g)$, aumentar o grupo mantém o número de ocorrências da sequência. Tal condição garante que o tamanho do grupo g é máximo.
6. $\forall g' \in PG \mid g' \subset g, occur(s, r, g') < occur(s, r, g)$, diminuir o grupo diminui o número de ocorrências da sequência. Tal condição garante que o tamanho do grupo g é mínimo.

Fazendo-se uso da formalização apresentada, o objetivo deste trabalho é encontrar todos os SRGs que respeitem a Definição 2.

4. Avaliação Experimental

Com o intuito de avaliar a formalização proposta, foi desenvolvido um algoritmo capaz de produzir os SRGs. A avaliação foi feita por meio de análise de sensibilidade, observando-se os resultados e o uso de recursos a partir da variação dos parâmetros de entrada e do tamanho do conjunto de dados.

O algoritmo recebe como entrada um conjunto de dados STS D, um conjunto de todos os itens distintos I apresentados em D, um conjunto de posições P referentes às STS e os limites definidos pelo usuário: γ no intervalo $]0, 1]$, β com valores inteiros a partir de 2, e σ com valores inteiros começando de 1.

O processo de busca por SRG foi dividido em três passos: (*i*) encontrar os KRGs, (*ii*) unir KRGs para identificar SRGs, e (*iii*) gerar candidatos para a próxima rodada. O algoritmo começa pelo passo (*i*) gerando sequências candidatas de tamanho um construídas a partir dos itens em I, considerando todo o seu período e todas as posições P. Dado que $\mathbf{SRG_k}$ é o conjunto de todos SRG de tamanho k. Em seguida busca todos os SRG_k que respeitem os parâmetros definidos pelo usuário γ, β, e σ. No passo (*ii*) para cada candidato, quando possível, os KRGs encontrados são mesclados e, a partir dos KRGs mesclados, o conjunto de SRG_k é gerado. Finalmente, no passo (*iii*) as sequências candidatas de tamanho $k + 1$ são geradas a partir da combinação de SRG_k. O processo é interrompido quando não se obtém sequências candidatas de tamanho $k + 1$. O algoritmo fornece como saída todos os SRGs encontrados.

Os resultados apresentados neste trabalho foram focados em um conjunto de dados sísmico público, o *inline* T401 (disponível em [DAL, 2020]), que faz parte do *F3 Block*,

produzido pelo método de reflexão sísmica em uma região localizada no setor holandês do Mar do Norte. O *inline* é composto por 951 sequências com marcação de tempo e espaço com 462 observações discretizado com um alfabeto de tamanho 25.

O conjunto de dados utilizado foi dividido em 16 quadrantes organizados de maneira retangular (4 x 4), enumerados em sequência, com o intuito de permitir que fossem realizados testes com variações de diferentes tamanhos de conjuntos de dados de entrada. Dessa forma foi possível fornecer como entrada diferentes quantidades de quadrantes. Cada quadrante tem aproximadamente 237 sequências com marcação de tempo e de espaço com 115 observações. Os experimentos foram implementados em R e executados em um computador com processador com 16 *cores*, 128GB de RAM e Ubuntu 20.04 LTS.

A Figura 1.a apresenta a correlação entre os parâmetros de entrada (γ, β e σ) e as informações de saída do algoritmo: o número de SRG, o número de ocorrências, o uso de memória e o tempo de execução, variando-se as seguintes configurações: $\gamma = \{0,6; 0,8; 1,0\}$, $\beta = [5, 10]$ e $\sigma = [5, 10]$. É possível observar que um valor menor de γ permite ao algoritmo localizar mais ocorrências, desta forma ele usa mais memória devido à quantidade maior de dados e gasta mais tempo para lidar com tais dados. No entanto, quanto mais baixo for γ, tem-se menos SRGs, dado que com uma frequência menor aumenta-se a possibilidade de unir SRG próximos. Tem-se também que com um β menor, geram-se mais grupos e ocorrências, e gasta-se mais memória e tempo. Para valores maiores de σ o número de ocorrências e SRG também são maiores, o uso de memória é menor e a execução é mais demorada.

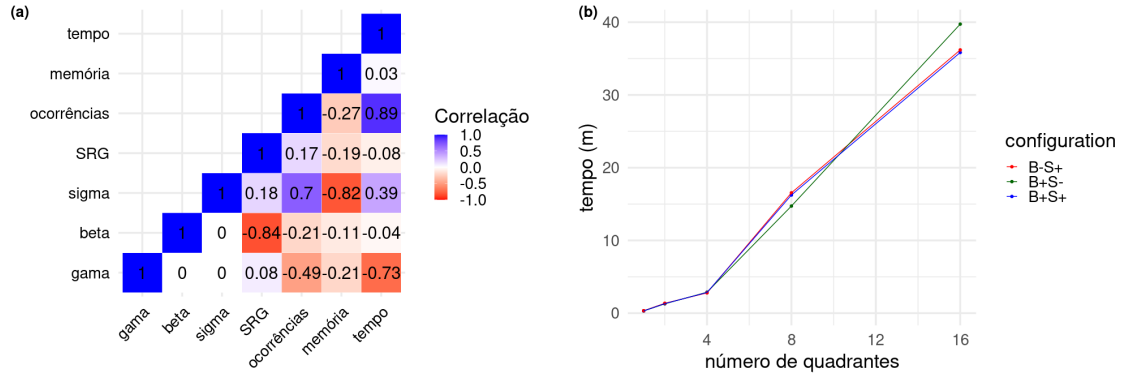

Figura 1. Correlação entre os parâmetros de entrada (γ, β e σ), o uso de recursos, e os resultados do algoritmo (a); Tempo de execução do algoritmo usando diferentes configurações e tamanhos de conjunto de dados (b).

A Figura 1.b mostra o tempo de execução com o aumento do tamanho do conjunto de dados e diferentes configurações. Para este gráfico variam os parâmetros β e σ, mantendo $\gamma = 0,8$ conforme detalhado a seguir: A configuração $B + S-$ é a mais restritiva, encontra apenas SRG mais "densos", com um maior número de elementos por grupo e menor distância mínima dos elementos do grupo ($\beta = 10$ e $\sigma = 5$). Por outro lado, a configuração $B - S+$ é a menos restritiva, permitindo poucos elementos em um grupo com grande distância uns dos outros ($\beta = 5$ e $\sigma = 10$). Finalmente, a configuração $B + S+$ tem muitos elementos por grupo, mas permite grande distância entre seus elementos ($\beta = 10$ e $\sigma = 10$).

Conforme esperado, aumentando-se o tamanho do conjunto de dados, o uso de recursos aumenta. A maior diferença de tempo ocorre com todo o conjunto de dados (16 quadrantes), uma diferença de 3,9 minutos, que corresponde a 9,82% de aumento de tempo das configurações $B + S+$ para $B + S-$.

A mesma variação de parâmetros foi feita para averiguação do uso de memória. A maior diferença também ocorre ao usar todo o conjunto de dados, 340,6 MB, que corresponde a 10,55% de aumento no uso de memória. Esse comportamento indica que as configurações dos parâmetros de entrada não fazem grande diferença no desempenho ou no uso de recursos, o que mais afeta o funcionamento do algoritmo é o tamanho do conjunto de dados dado como entrada.

5. Conclusão

Este trabalho abordou o problema de mineração de sequências restritas no espaço e no tempo. Fundamentos importantes para o processo e as noções de grupo, RG, KRG e SRG foram introduzidos. Através destas informações é possível encontrar diferentes tamanhos de sequências frequentes, intervalos de tempo e regiões do espaço onde uma sequência é frequente. Até onde se sabe, este é o primeiro trabalho a abordar o problema com uma dimensão de tempo e três dimensões de espaço.

Foram realizados experimentos com um algoritmo sobre um conjunto de dados sísmicos real. O comportamento do algoritmo foi detalhado com diferentes configurações de parâmetros e tamanhos de conjuntos de dados em uma análise de sensibilidade. De maneira geral, a alteração de parâmetros de entrada resulta em poucas mudanças no uso de recursos. Esse comportamento indica que o que mais afeta seu funcionamento é o tamanho do conjunto de dados fornecido como entrada.

Referências

H. Alatrista-Salas, S. Bringay, F. Flouvat, N. Selmaoui-Folcher, and M. Teisseire. Spatio-sequential patterns mining: Beyond the boundaries. *Intelligent Data Analysis*, 20(2):293–316, 2016.

R. Campisano, H. Borges, F. Porto, F. Perosi, E. Pacitti, F. Masseglia, and E. Ogasawara. Discovering tight space-time sequences. In *Lecture Notes in Computer Science*, volume 11031 LNCS, pages 247–257, 2018.

Data Analitics Lab DAL. Generalized Discovery of Tight Space-Time Sequences. Technical report, https://eic.cefet-rj.br/ dal/generalized-discovery-of-tight-space-time-sequences/, 2020.

C. Flamand, M. Fabregue, S. Bringay, V. Ardillon, P. Quénel, J.C. Desenclos, and M. Teisseire. Mining local climate data to assess spatiotemporal dengue fever epidemic patterns in French Guiana. *Journal of the American Medical Informatics Association*, 21(e2):e232–240, 2014.

Y. Huang, L. Zhang, and P. Zhang. A framework for mining sequential patterns from spatio-temporal event data sets. *IEEE Transactions on Knowledge and Data Engineering*, 20(4): 433–448, 2008.

B. Koseoglu, E. Kaya, S. Balcisoy, and B. Bozkaya. ST Sequence Miner: visualization and mining of spatio-temporal event sequences. *Visual Computer*, 36(10-12):2369–2381, 2020.

K. Li and Y. Fu. Prediction of human activity by discovering temporal sequence patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(8):1644–1657, 2014.

B. Saleh and F. Masseglia. Time aware mining of itemsets. In *Proceedings of the International Workshop on Temporal Representation and Reasoning*, pages 93–97, 2008.