**UNIVERSIDADE FEDERAL DO CEARÁ**

**CENTRO DE CIÊNCIAS**

**DEPARTAMENTO DE COMPUTAÇÃO**

**PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**

**MESTRADO ACADÊMICO EM CIÊNCIA DA COMPUTAÇÃO**

**CLEILTON LIMA ROCHA**

# TPRED: UM FRAMEWORK ESPAÇO-TEMPORAL DE PREDIÇÃO DE LOCALIZAÇÃO

**FORTALEZA**

**2016**

CLEILTON LIMA ROCHA

# TPRED: UM FRAMEWORK ESPAÇO-TEMPORAL DE PREDIÇÃO DE LOCALIZAÇÃO

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Banco de Dados.

Orientador: Prof. Dr. José Antônio Fernandes Macêdo.

FORTALEZA

2016

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

R572t Rocha, Cleilton Lima.
TPRED: um framework espaço-temporal de predição de localização / Cleilton Lima Rocha. – 2016.
71 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Ciências, Programa de Pós-Graduação em Ciência da Computação, Fortaleza, 2016.
Orientação: Prof. Dr. José Antônio Fernandes Macêdo.

1. Predição de localização. 2. Espaço-temporal. 3. Trajetória. 4. Árvore de sufixo probabilística. I. Título.
CDD 005

---

CLEILTON LIMA ROCHA

TPRED: UM FRAMEWORK ESPAÇO-TEMPORAL DE PREDIÇÃO DE LOCALIZAÇÃO

Dissertação apresentada ao Curso de Mestrado Acadêmico em Ciência da Computação do Programa de Pós-Graduação em Ciência da Computação do Centro de Ciências da Universidade Federal do Ceará, como requisito parcial à obtenção do título de mestre em Ciência da Computação. Área de Concentração: Banco de Dados.

Aprovada em: 30/08/2016.

BANCA EXAMINADORA

---

Prof. Dr. José Antônio Fernandes Macêdo (Orientador)
Universidade Federal do Ceará (UFC)

---

Prof. Dr. Fábio André Machado Porto
Laboratório Nacional de Computação Científica (LNCC)

---

Prof. Dr. Javam de Castro Machado
Universidade Federal do Ceará (UFC)

---

Prof. Dr. José Maria da Silva Monteiro Filho
Universidade Federal do Ceará (UFC)

Aos meus familiares e amigos.

## AGRADECIMENTOS

Sou eternamente grato ao meu Deus que sempre me deu força, ânimo, descanso, inteligência e sabedoria para que eu conseguisse concluir o mestrado.

Agradeço aos meus familiares que nunca mediram esforços para que eu continuasse os meus estudos. Em especial agradeço ao meus pais que me educaram e me criaram para ser um homem destemido, honesto, sincero e simples. Sei que sempre poderei contar com eles em todas as fases da minha vida.

Quero agradecer a minha esposa e filhas, por terem me encorajado e apoiado. Minha esposa sempre me deu carinho e me consolou com palavras amigáveis quando estava desanimado.

Na UFC fiz grandes amigos. Agradeço a todos que de alguma forma me ajudaram.

Obrigado, João Bosco e Felipe Timbó pelas orientações antes de eu iniciar no mestrado.

Agradeço aos amigos do projeto Eai? e aos meus amigos Igo e Francesco que me ajudaram na elaboração do artigo, muito obrigado!

Sou grato a todos os meus professores que foram tão importantes na minha vida acadêmica.

Agradeço especialmente ao professor José Antônio, além de ter tido o prazer de ser seu aluno ao longo do curso, tê-lo como meu orientador foi realmente uma honra. Muito obrigado pela sua confiança, sinceridade, incentivo, paciência, dedicação e contribuição.

Agradeço também aos meus irmãos em Cristo que sempre oraram por mim quando compartilhava as minhas ansiedades e preocupações.

Agradeço as empresas em que trabalhei durante o curso do mestrado, Great, Nex2me e Instituto Atlântico, por me darem todo o suporte necessário ao longo do mestrado.

Finalmente agradeço a todos que direta ou indiretamente contribuíram para minha formação pessoal, profissional e acadêmica; juntamente comigo, vocês conquistaram esta vitória.

"Assim diz o SENHOR: Não se glorie o sábio na sua sabedoria, nem o forte, na sua força, nem o rico, nas suas riquezas. Mas aquele que se gloria, glorie-se nisto: em me conhecer e saber que eu sou o SENHOR e faço misericórdia, juízo e justiça na terra; porque destas coisas me agrado, diz o SENHOR." (Jeremias, 9:23-24)

## RESUMO

O aumento da difusão de dispositivos equipados com GPS trouxe a possibilidade da coleta de dados dos movimentos dos objetos em uma escala como nunca visto antes. Durante os últimos anos, essa difusão incentivou o desenvolvimento de diferentes técnicas capazes de lidar com a predição de localização. Existem muitos trabalhos que visam principalmente prever o próximo local de um objeto em movimento concentrando-se apenas na informação de domínio espacial. Em nosso trabalho queremos considerar também as informações temporais e para isso introduzimos os conceitos de *ciclo temporal* e *partição temporal* para melhorar a confiabilidade das predições realizadas e também para responder, não apenas *qual* será a próxima localização relevante de um objeto em movimento, mas também prever *quando* esse movimento irá ocorrer, ou seja, quando o objeto deixará o seu local atual. Para atingirmos nosso objetivo propomos o TPRED, um *framework* baseado em um modelo de árvore de sufixo probabilística que aprende quais são os movimentos padrões de um objeto em movimento e computa as predições através da exploração das informações no domínio espacial e temporal. Para validarmos nossa contribuição realizamos um extenso conjunto de avaliações experimentais com diferentes métricas sobre duas bases de dados de aplicações do mundo real, a fim de mostrarmos a eficácia e a eficiência de nossa proposta. Nossa abordagem apresenta os melhores resultados tanto na predição espacial quanto na temporal quando comparado com outros dois *baselines*.



**Palavras-chave**: predição de localização; espaço-temporal; trajetória; árvore de sufixo probabilística.

## ABSTRACT

The vast diffusion of devices equipped with a GPS device has brought the possibility of collecting data related to massive amounts of moving objects on a scale never seen before. During the latest years, such diffusion instigated the development of many different techniques to deal with location prediction problems. Existing works mainly aim at predicting the next location of moving objects by focusing on information in the spatial domain. In this paper we want to take into account information in the temporal domain as well, both to improve the reliability of predictions and to answer not only *where* a moving object is going to move, but also *when* an object is expected to leave its current location. To this end we propose TPRED, a framework based on probabilistic suffix trees which tries to capture typical movement patterns of moving objects, and compute reliable predictions accordingly, by exploiting information both in the spatial and temporal domains. In order to prove the validity of our contribution we conduct an extensive set of experimental evaluations, based on real-world datasets and different performance metrics, where we show the efficiency and effectiveness of our proposal.



**Keywords**: spatio-temporal location prediction; trajectory; mobility; probabilistic suffix trees.

## LISTA DE FIGURAS

## LISTA DE TABELAS

## LISTA DE ALGORITMOS

## SUMÁRIO

# 1 INTRODUÇÃO

Com a crescente difusão de dispositivos móveis equipados com GPS e o fácil acesso as redes móveis (por exemplo, *smartphones*, veículos rodoviários, etc.) é possível coletar dados das trajetórias dos objetos em uma escala como nunca visto antes. Uma vez que os dados tenham sido coletados e armazenados podemos realizar análises sobre tais informações, a fim de identificarmos o comportamento da mobilidade dos objetos, e assim, através desses dados surgem novas possibilidades para desenvolvermos novos *frameworks* teóricos e algoritmos para estudarmos e compreendermos o mundo a nossa volta.

Por exemplo, (HORNE *et al.*, 2007) explora o comportamento dos movimentos dos animais e propõe um novo modelo para estimar os movimentos destes animais. (BARABASI; CRANDALL, 2002); (CHEN *et al.*, 2014) e (NEWMAN *et al.*, 2006) mostram de diferentes formas como as pessoas podem estar relacionadas umas com as outras em uma rede social *online* e como isso afeta suas interações físicas, por exemplo, como as doenças podem ser disseminadas. (BOBADILLA *et al.*, 2013); (SUN *et al.*, 2015) estudaram *como*, *onde* e *quando* os usuários interagem com aplicações (por exemplo, serviços de redes sociais), e como isso pode ajudar na disseminação de novas ideias e na recomendação de itens (livros, filmes, locais, etc.) com alto nível de personalização.

A partir dos dados de geolocalização coletados é possível estudar o comportamento de mobilidade das pessoas em uma cidade e isso pode ajudar as instituições responsáveis a organizarem e melhorarem o trânsito, e a distribuírem melhor o tráfego quando situações caóticas acontecerem (MONREALE *et al.*, 2009).

A atividade de prever a localização é fundamental para criar, melhorar e alavancar o serviço oferecido, dependendo da finalidade e necessidade desse serviço. (DHAR; VARSHNEY, 2011) Essa atividade é aplicada em muitos serviços e aplicações baseados em localização, por exemplo, recomendação de locais, serviços de roteamento, monitoramento, etc. Também pode ser usada em serviços baseados em localização de redes sociais, que exploram os dados das trajetórias dos usuários e suas informações coletadas e/ou compartilhadas na WEB, como por exemplo, o projeto Geolife criado por (ZHENG *et al.*, 2010).

Basicamente podemos descrever que o objetivo da predição da localização é em termos gerais, o desafio de prever a próxima localização, o mais exata possível, de múltiplos objetos em movimento considerando o histórico dos dados espaço-temporais que abrangem seus padrões de mobilidade. O nível da predição pode ser variado. Podemos realizar predições em

nível individual, onde o foco é prever o movimento de um único objeto, e também em nível global, onde o objetivo é resumir os movimentos padrões que caracterizam um conjunto de objetos em movimento a fim de predizer como os movimentos gerais ocorrem.

Nosso objetivo é prever a próxima localização de um objeto em movimento considerando os dados espaço-temporais coletados ao longo do tempo, os dados coletados desse objeto formam o seu histórico de movimentos. Trabalharemos em predições que serão realizadas em nível individual.

Há muitos trabalhos existentes na literatura, por exemplo, (XUE *et al.*, 2013b); (XUE *et al.*, 2013a); (LEI *et al.*, 2013) e (MONREALE *et al.*, 2009) que focam na predição de localização de um único objeto em movimento. Nesses trabalhos observamos que o domínio temporal não desempenha um papel importante, enquanto nós discutimos que esse é um aspecto fundamental para uma melhoria significativa na precisão das predições realizadas, e também no aumento da utilidade dessas predições. Por exemplo, uma aplicação de roteamento pode desejar informar aos seus usuários as condições do tráfego que serão enfrentadas até o destino predito ser alcançado. Já considerando a predição no domínio temporal, o tempo previsto de quando o usuário deixará o seu local, o aplicativo pode fornecer mais informações úteis ao usuário, como por exemplo, recomendar a antecipação de seu deslocamento com base nas condições do tráfego. Em vista disso, pretendemos estender o problema da predição de localização para resolver *qual* a próxima localização de um objeto em movimento - visando uma melhor precisão ao consideramos as informações do domínio temporal - e também prever *quando* um objeto em movimento partirá do seu local atual.

Propomos o TPRED, um *framework* que construirá o modelo preditiivo e também executará o algoritmo de predição sobre este modelo. Nosso *framework* foi inspirado no trabalho de (LEI *et al.*, 2013). No TPRED os pontos de uma trajetória são formados pela latitude e longitude, os quais são transformados em uma sequência de células. Essas células são disjuntas e fazem parte de um grid espacial. A partir dessas células definimos as células de paradas que representam as regiões espaciais relevantes para o objeto em que ele permanece uma quantidade significativa de tempo. As células de parada representam os padrões de mobilidade do objeto e são usadas para construir o seu modelo preditivo. Os nós do modelo representam as células de parada e as arestas representam as transições entre as células de parada e contém as informações espacial e temporal. Também apresentamos um algoritmo preditivo que será responsável por realizar a predição nos domínios espacial e temporal sobre o modelo preditivo do objeto. Para

computar uma predição de localização de um objeto em movimento são necessários seu modelo preditivo, o histórico dos seus movimentos mais recentes e o tempo de consulta para o qual a predição será realizada.

## 1.1 Exposição do problema

Neste trabalho consideramos que durante o deslocamento de um objeto em movimento existem várias possibilidades para seu destino. Diante dessas possibilidades queremos prever, em nível individual, *qual* será a próxima localização para onde o objeto se deslocará no futuro próximo, sendo que nesta localização o objeto sempre passa quantidades substanciais de tempo associadas a esta região espacial. Por exemplo, se um objeto em movimento é uma pessoa, então exemplos de locais relevantes podem ser sua casa, seu local de trabalho, uma academia e assim por diante. Também relacionado a este problema queremos prever *quando* um objeto em movimento partirá de sua localização atual.

Em 4.1, apresentamos um conjunto de definições básicas que são necessárias para descrevermos formalmente os cenários que consideramos, e, na seção 4.2, formulamos o problema que desejamos resolver.

## 1.2 Contribuição

As principais contribuições da dissertação podem ser resumidas da seguinte forma:

- Criamos uma variação do problema tradicional de predição de localização, no qual pretendemos prever, a *próxima* localização relevante de um objeto em movimento e *quando* ocorrerá o próximo deslocamento de um objeto, ou seja, *quando* ele partirá de sua localização atual para o local que está sendo predito.
- Propomos um *framework* chamado TPRED para resolver nossa variante do problema de predição de localização. Inspirado parcialmente no trabalho proposto por (LEI *et al.*, 2013) apresentamos nosso próprio modelo preditivo baseado em árvore de sufixo probabilística da seguinte forma: (i) os nós na árvore são as regiões relevantes para o objeto em questão, considerando o tempo que permanecem nelas; (ii) as arestas entre os nós representam as transições, e os tempos em que as transições ocorrem são armazenados, pois são utilizados na predição espacial e temporal; (iii) para encontrarmos o tempo mais representativo que indica quando um objeto em movimento deixará o local atual em direção ao local predito

utilizamos a técnica de clusterização de histogramas, proposta por (NG *et al.*, 2005), sobre os dados temporais das transições entre as relevantes regiões; (iv) e adicionamos o conceito de *ciclos temporais* e *partições temporais* para identificarmos a periodicidade de quando os padrões de mobilidade se repetem. Através do modelo preditivo fazemos várias contribuições que melhoram a identificação dos padrões de movimento, onde são exploradas as informações dos domínios espacial e temporal. E finalmente, introduzimos o algoritmo de predição de localização responsável por computar a predição com base no modelo citado anteriormente.

- Avaliamos o framework TPRED sobre dois conjuntos de dados de diferentes aplicações do mundo real contendo dados espaço-temporais de usuários reais. Com esses dados realizamos uma extensa avaliação experimental comparando o TPRED com outros dois *baselines* onde mostramos a eficácia e eficiência da nossa proposta através de diferentes métricas de desempenho.

## 1.3 Publicação

Publicamos o seguinte artigo:

- (ROCHA *et al.*, 2016) Cleilton Lima Rocha, Igo Ramalho Brilhante, Francesco Lettich, Jose Antônio F. De Macedo, Alessandra Raffaetà, Rossana Andrade and Salvatore Orlando. 2016. **TPRED: a Spatio-Temporal Location Predictor Framework**. In 20th International Database Engineering & Applications Symposium (IDEAS '16). Montreal, QC, Canada.

## 1.4 Organização

Os capítulos restantes são organizados da seguinte forma: Apresentamos os conceitos básicos sobre trajetória e predição no capítulo 2, bem como os tipos de predições de localização existentes. O capítulo 3 apresenta os trabalhos relacionados. O capítulo 4 apresenta a metodologia da nossa proposta e no capítulo 5 mostramos os experimentos realizados em nosso *framework*, TPRED, e em outros dois *baselines*. E finalmente, para concluir mostramos no capítulo 6 a conclusão e fazemos as considerações sobre os trabalhos futuros.

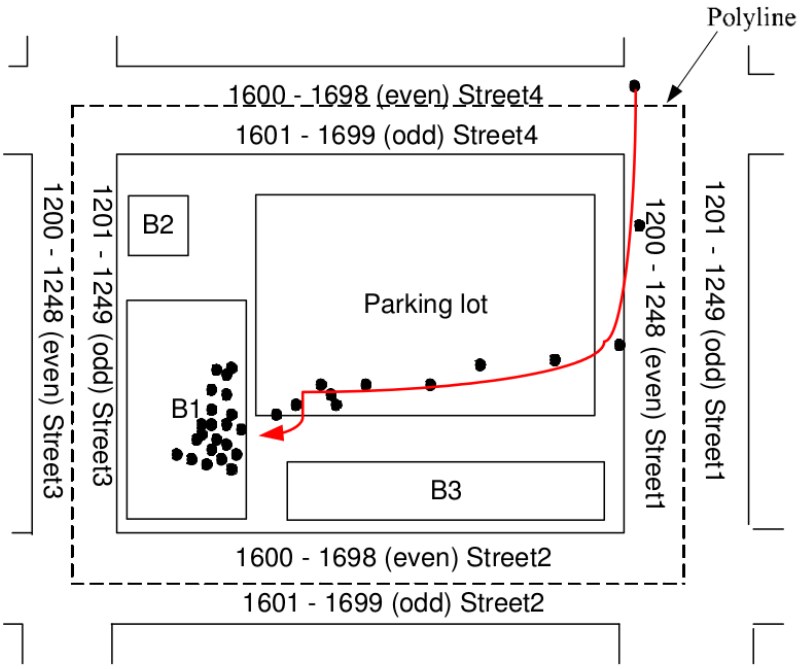
# 2 FUNDAMENTAÇÃO TEÓRICA

## 2.1 Trajetória

Comumente as trajetórias consistem numa sequência de pontos espaço-temporais coletados ao longo do tempo de um objeto em movimento. Durante a coleta são armazenados a informação espacial, que consiste da latitude e longitude do ponto geográfico, e também são armazenados os *timestamps* que indicam quando a coleta de cada ponto geográfico foi realizada, sendo esta a informação temporal. Na Figura 1 mostramos o exemplo de uma trajetória de um objeto em movimento composta de uma sequência de amostras de pontos geográficos coletados ao longo do tempo contendo as informações temporais e espaciais. Observando a Figura 1 percebemos que, após o deslocamento, o objeto chega a um local específico e ali permanece uma quantidade significativa de tempo, pois há uma região densa representada pela concentração de pontos em uma região específica. Por exemplo, se o objeto em movimento for um estudante essa trajetória pode representar o seu trajeto para escola a partir de uma estação de metrô.

Os dados coletados ao longo das trajetórias precisam ser analisados e processados, ou seja, precisam ser minerados para que os padrões do comportamento dos movimentos do objeto possam ser extraídos e, posteriormente, serem representados.

Figura 1 – Visão geral de uma trajetória de um objeto em movimento.




Fonte: (LIU *et al.*, 2006)

## 2.2 Mineração de padrões de trajetória

Os padrões espaço-temporal permitem representar de forma sucinta como os objetos se movem e nos permite compreender como os indivíduos se comportam em relação a mobilidade. Com a difusão das tecnologias ubíquas haverá um aumento significativo tanto na precisão da localização dos indivíduos como no volume de dados a serem analisados relativos as trajetórias individuais (GIANNOTTI *et al.*, 2007), por isso é necessário termos métodos que solucionem o problema de identificação de padrões de trajetórias.

De acordo com (GIANNOTTI *et al.*, 2007), o padrão de trajetória representa um conjunto de trajetórias individuais em que os mesmos locais em um dado espaço, denominados regiões de interesse, são visitados na mesma sequência, e com tempos de viagem de transição dos objetos entre as regiões de interesse semelhantes.

Em um padrão de trajetória não é definida uma rota específica entre duas regiões consecutivas, pois não é relevante o trajeto, mas são relevantes as regiões de interesse. Em nosso trabalho consideramos que nos padrões de trajetórias os mesmos locais são visitados com frequência e a sequência em que estes são visitados é relevante. Os tempos de viagens de deslocamentos podem ser distintos, bem como o início em que as transições, de uma região para outra, ocorrem. Os padrões de trajetórias podem ser definidos considerando apenas os movimentos do objeto em movimento, nível individual, ou podem ser definidos com a colaboração dos movimentos de outros objetos para representar o comportamento de mobilidade de um objeto. Em nossa abordagem apenas os dados do objeto em movimento contribuem para identificação do seu padrão de trajetória e os dados de outros objetos, ainda que semelhantes, não colaboram e nem cooperam para identificação desses padrões, ou seja, somente os dados do objeto em movimento são utilizados para construção do seu modelo preditivo.

É comum considerar o problema para identificar as regiões de interesse de um objeto em movimento, e consequentemente extrair os seus padrões de trajetórias, como um problema em que é necessário discretizar o espaço para que as trajetórias sejam representadas de forma mais concisa. Por exemplo, (KANG; YONG, 2008) propõe um método de discretização do mundo com o objetivo de extrair os padrões de sequências espaço-temporal preservando o máximo possível as informações tanto do domínio espacial quanto temporal. (GIANNOTTI *et al.*, 2007) propõem duas soluções para mineração de trajetórias as quais consideram o problema de identificação das regiões de interesse como um problema de discretização espacial. Na Figura abaixo vemos uma representação do espaço discretizado. Como podemos percerber o mundo foi

dividido em células e entre estas não há intersecção.

Uma vez que os padrões de trajetórias do objeto em movimento foram extraídos podemos representar o seu comportamento de mobilidade e utilizar os padrões de trajetória para realizar as predições de localização.

## 2.3 Predição de localização

O mercado de serviços e aplicações baseados em localização vem se destacando e evoluindo ao longo dos últimos anos. Aplicações e serviços como recomendação de locais, propagandas direcionadas aos usuários e recomendação de trajetórias durante o tráfego, dentre outras, são alguns exemplos que podemos citar que utilizam a localização do usuário, e se beneficiarão ainda mais com técnicas de predição de localização nos âmbitos espacial e temporal.

As pesquisas de predição de localização têm avançado nos últimos anos graças a disseminação dos dispositivos móveis equipados com GPS e às novas técnicas que têm surgido nessa área de pesquisa.

De forma resumida o objetivo da predição de localização é prever qual será o próximo local relevante para onde um objeto em movimento se deslocará considerando os seus movimentos coletados anteriormente. Podemos realizar predições em nível individual, onde o foco é prever a localização de um único objeto, e também em nível global quando a predição ocorre para múltiplos objetos. Além da predição sobre o domínio espacial também são exploradas predições no âmbito temporal que tem por objetivo responder quando o objeto em movimento deixará sua localização atual e se deslocará para o próximo local relevante, e também quando o objeto chegará ao seu destino. Assim as predições podem ser realizadas no escopo espacial e temporal e o uso das informações, obtidas a partir das predições, podem tornar muitos serviços pró-ativos em suas aplicações.

Geralmente a predição de localização em si é um meio para que um objetivo final seja alcançado, por isso, usualmente as aplicações utilizam a predição de localização como parte do processo para que um serviço seja oferecido. Por exemplo, uma aplicação que informa os dados de um local, como temperatura e rotas de acesso, poderá informar automaticamente esses dados de forma proativa ao prever qual o próximo destino do usuário, podendo indicar a rota mais rápida entre sua localização atual e o seu destino e também condições climáticas, considerando o tempo de chegada na localização final.

Para que as predições sejam realizadas é necessário que as informações das tra-

jetórias sejam coletadas, transmitidas a um servidor onde são armazenadas em uma base de dados e analisadas. Também é possível ler esses dados coletados de forma *offline*. Nesse caso, usualmente, os dados são lidos através de algum sistema e são armazenados para serem processados. Para construção e representação das trajetórias, os dados coletados podem passar por um pré-processamento para que sejam extraídos apenas os dados de interesse, os padrões espacial e temporal das trajetórias, e sobre esses padrões as predições serão realizadas.

Na Figura 2 temos o exemplo de uma predição de localização. Observando a imagem e considerando que os movimentos mais recentes do objeto em movimento é o trajeto que pode ser resumido através do caminho **A-B**, e dado que o objeto está na localização **B** e que o algoritmo de predição precisa decidir entre as localizações **C** e **E**, dado essas informações, podemos facilmente computar, baseado na probabilidade informada na Figura 2, que a localização a ser predita será a localização **C**. Considerando que os padrões de trajetórias de um objeto em movimento representados na Figura 2 seja de um estudante, podemos representar a **região A** como sua casa, a **região B** pode representar a universidade em que estuda, e sua rotina pode ser resumida da seguinte forma: alguns dias da semana ele vai para o seu curso de idioma, **região E**, e outros dias, ele vai para o seu local de estágio, **região C**, sendo esta a localização mais frequente ao realizar o percurso **A-B**.

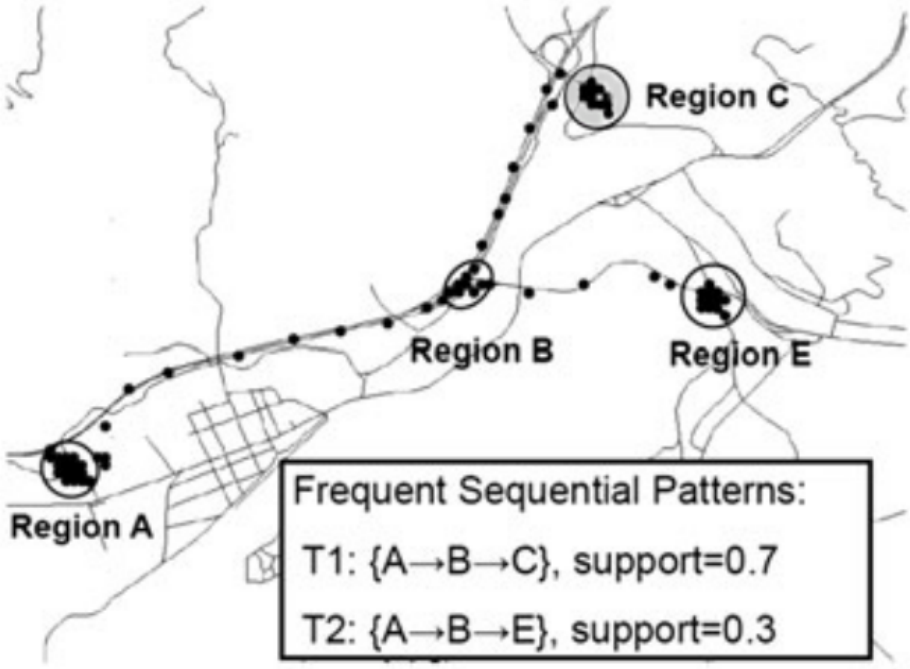
Os métodos de predição de localização mais comum são: predição de localização baseada em padrões de trajetórias, por exemplo (GAMBS *et al.*, 2011); predição de localização baseada na semântica de trajetórias, por exemplo (YING *et al.*, 2013) e predição de localização baseada em redes sociais, por exemplo (ZHU *et al.*, 2014). Estes métodos têm impulsionado ainda mais essa área de pesquisa e tem contribuído para que novas técnicas de predição de localização surjam. A seguir abordaremos esses métodos de predição de localização.

### *2.3.1 Predição de localização baseada em padrões de trajetória*

Muitas são as variáveis que influenciam os padrões de mobilidade das pessoas e de outros objetos. Variando desde meios de transportes para as regiões de interesse, a distância entre essas localizações, prioridades relacionadas à família, condições climáticas, dentre outros (GONZáLEZ *et al.*, 2008).

Uma técnica existente para prever os locais relevantes de um objeto em movimento é a predição baseada em padrões de mobilidade, ou seja, predição de localização baseada em padrões de trajetória, que tem como objetivo analisar os dados dos movimentos anteriormente

Figura 2 – Exemplo de uma predição de localização considerando as probabilidades informadas na imagem.

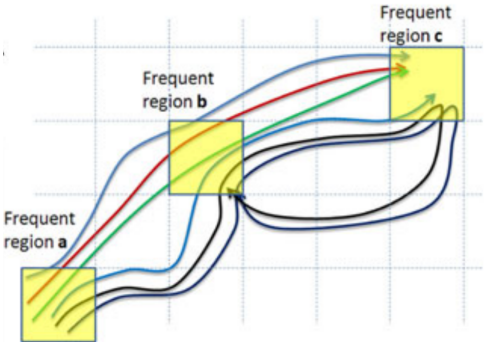



Fonte: (LEI *et al.*, 2013)

capturados, identificar os padrões de mobilidade existentes e representá-los para que seja possível realizar predições sobre tais padrões. Sendo de extrema importância o histórico das localizações, coletados ao longo do tempo, dos objetos.

Na Figura 3 demonstramos como as *trajetórias* (seguimentos), como as *regiões* ( células no grid) e como as *regiões de interesse*, células que são atravessadas por todos os seguimentos, podem ser representadas (LEI *et al.*, 2013). Ao identificarmos as regiões de interesse é possível minerar os padrões de trajetórias de um objeto em movimento e representar esses padrões de mobilidade em um modelo preditivo para que as predições possam ser realizadas sobre o modelo construído.

Figura 3 – Representação de várias trajetórias de um objeto e suas regiões de interesse.




Fonte: (LEI *et al.*, 2013)

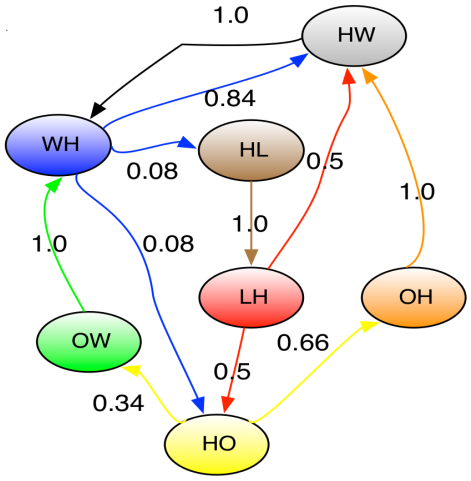
(MONREALE *et al.*, 2009) apresenta um trabalho no qual os movimentos padrões dos objetos são definidos como trajetórias padrões, que são uma representação dos movimentos mais comum dos objetos, como uma sequência de regiões frequentemente visitadas por um objeto. Essa sequência de regiões frequentes constroem uma árvore de decisão chamada árvore de trajetórias padrões, que contém as informações espaço-temporais dos dados coletados. Essa árvore é construída e evoluída a medida que novas regiões frequentes são identificadas após o processamento dos dados coletados dos objetos em movimento. Os objetos que se movimentam por uma determinada região contribuem para definir se uma região é popular ou não, ou seja, não são analisados apenas os dados de um único indivíduo para construção da árvore de decisão probabilística, e naquele trabalho há uma colaboração direta de outros objetos na predição da localização. A predição de localização considera também as informações temporais como parte importante para o cálculo da probabilidade de predição.

A Cadeia de Markvov é bastante utilizada no problema de predição de localização. Ela é utilizada para representar os movimentos de trajetórias padrões de um objeto. (GAMBS *et al.*, 2011) define uma Cadeia de Markov como um automâto probabilístico com a propriedade de que a distribuição de probabilidade das próximas localizações depende apenas da localização atual e não da sequência de locais que a precederam.

(GAMBS *et al.*, 2011) propôs um modelo de mobilidade chamado Cadeia de Markov de Mobilidade (MMC). Este modelo representa de forma compacta os comportamentos de mobilidade de um indivíduo objetivando a privacidade dos dados coletados. (GAMBS *et al.*, 2012) propôs uma extensão da Cadeia de Markov de Mobilidade para prever qual o próximo local em que um indivíduo estará, considerando o comportamento do indivíduo por um período de tempo e as localizações recentes que ele visitou. O modelo estendido, denominado *n*-MMC, diferencia-se do MMC por conter a quantidade *n* de localizações recentemente visitadas. Por exemplo, pode-se desejar prever a próxima localização considerando apenas as 3 (três) últimas localizações anteriores, ou as últimas 4 (quatro), e assim, por diante. Na Figura 4 vemos o exemplo do modelo *n*-MMC construído a partir dos dados de um objeto em movimento onde as predições serão realizadas considerando as duas últimas localizações. Os labels na imagem representam os dois últimos locais relevantes do trajeto de um objeto em movimento, por exemplo, o label **HW**, representa o percurso *casa-trabalho*. Se o objeto em movimento realiza o trajeto **HW**, e dado que está no local **W** é certo que o trajeto **WH**, *trabalho-casa*, será realizado, pois o valor da probabilidade de transição é 1 na Cadeia de Markov.

O próximo tipo de predição de localização explora o impacto do comportamento de mobilidade dos objetos quando associados a uma informação semântica, além de considerar os movimentos padrões dos usuários formados a partir dos seus dados espaço-temporais coletados ao longo do tempo.

Figura 4 – Representação gráfica de uma 2-MMC a partir dos dados coletados de um objeto.




Fonte: (GAMBS *et al.*, 2012)

### *2.3.2 Predição de localização baseada na semântica das trajetórias*

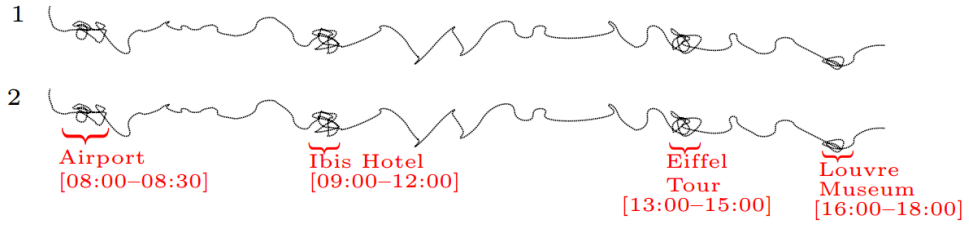
Geralmente uma amostra de um dado espaço-temporal coletado de um objeto em movimento contém informações geográficas e o *timestamp* da coleta; raramente os sistemas coletam ou identificam a informação *semântica* durante a trajetória de um objeto. Uma informação semântica pode ser, por exemplo, a identificação de uma academia associada ao seu endereço principal na via juntamente com a informação espacial e temporal. Através da informação semântica juntamente com a geolocalização os sistemas podem ser mais proativos, adaptáveis, flexíveis e úteis aos usuários. Com as informações semânticas das localizações as aplicações poderiam compreender melhor o usuário em um contexto bem específico. Por exemplo, uma interessante aplicação do uso da informação semântica de uma localização é demonstrada no jogo *Pokémon GO*[1], onde determinados *pokémons* aparecem com maior frequência em locais, cuja dado semântico é ali identificado. Por exemplo, *pokémons* com habilidades aquáticas, aparecerão em locais próximos a rios, lagos, etc.

A noção de trajetórias semânticas foi introduzida por (ALVARES *et al.*, 2007) citado por (YING *et al.*, 2011). Segundo (YING *et al.*, 2011) uma trajetória semântica compreende

[1] https://play.google.com/store/apps/details?id=com.nianticlabs.pokemongo&hl=pt_BR

uma sequência de locais marcados com informação semântica, que além do posicionamento geográfico é possível identificar também as atividades realizadas pelos usuários nessas trajetórias. Na Figura 5 vemos a representação de uma trajetória sem a informação semântica e a mesma trajetória contendo o dado semântico.

Figura 5 – A Figura acima mostra a representação de uma trajetória simples e a Figura abaixo a representação de uma trajetória semântica .




Fonte: (BOGORNY *et al.*, 2009)

Uma vez que a informação semântica não é coletada durante a fase da coleta de dados estas precisam ser mapeadas em uma fase posterior a da coleta. (LIU *et al.*, 2006) propôs um método de extração de informação semântica da localização a partir da trajetória do objeto. A informação é extraída apenas dos locais onde o objeto passa uma quantidade significativa de tempo em uma localização relevante. Atualmente uma forma comum de se obter a informação semântica da localização é através do próprio usuário, quando este explicitamente associa alguma atividade ou marca alguma categoria de um local nas aplicações através dos *check-ins*, pois além das informações geoespaciais os *check-ins* também contém informações relevantes sobre o local e a atividade que o usuário desempenha em determinada localização em um período específico de tempo.

Os métodos de predição de localização em sua maioria consideram apenas informações geoespaciais e temporais para determinar o comportamento padrão dos usuários, restringindo o padrão de comportamento dos usuários somente às propriedades geográficas. Por isso, esses métodos tendem a prever locais populares onde a maioria das pessoas frequenta. Sendo assim, essas técnicas não são adequadas para prever locais que não foram visitados anteriormente pelos indivíduos, visto que não fazem parte do seu padrão de mobilidade e nem do seu histórico de trajetórias (YING *et al.*, 2011).

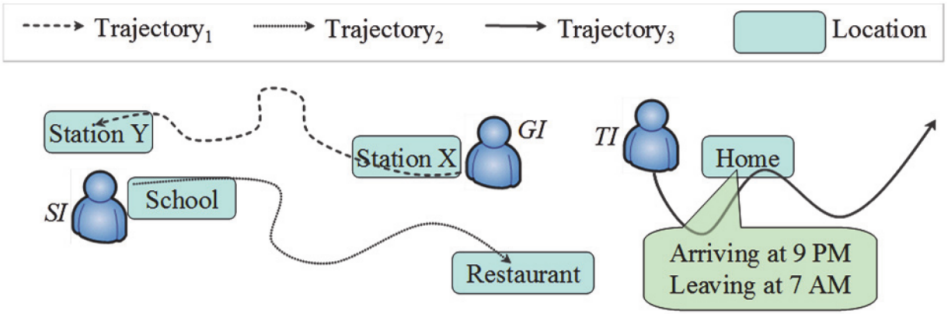
Em (YING *et al.*, 2011) é apresentado um *framework* de predição chamado SEMANPREDICT. Esse *framework* se beneficia das informações de semântica contidas na trajetória para prever o próximo local do usuário. O *framework* consiste em duas atividades principais. A primeira consiste em minerar os dados *offline*, onde são extraídos os comportamentos semânticos

dos movimentos padrões do usuário. Após a fase de mineração dos dados o *framework* realiza o agrupamento de usuários com base no comportamento semântico e explora os padrões semânticos dos usuários pertencentes a um mesmo cluster. A segunda atividade é realizada em tempo real e consiste em realizar a predição baseada em uma técnica de clusterização, explorando tanto as informações semânticas quanto as localizações geográficas.

Segundo (YING *et al.*, 2013) os usuários frequentam os locais por diferentes razões e intenções, e os padrões de trajetórias dos usuários podem ser extraídos a partir dos movimentos frequentes desencadeados por três tipos de intenções: semântica, temporal e geográfica.

Na Figura 6 vemos o cenário de um objeto em movimento baseado nos três tipos de intenções. Na itenção geográfica pode-se mapear o local específico e as razões pelas quais há o deslocamento de um local para o outro; pode-se prever que os usuários que estão na estação **X** irão para a estação **Y**. Já a itenção temporal informa as razões pelas quais um usuário visita e deixa um local em determinado momento; como vemos na Figura 6 um usuário tende a deixar sua casa pela manhã e só retornar a noite. E a informação semântica se refere ao contexto geral e as razões pelas quais os usuários se deslocam de algumas localizações para outras; considerando que os usuários almoçam após o horário de suas atividades, pode-se prever, para os usuários que deixam a escola no horário do almoço, locais que contenha restaurantes como sendo a próxima localização relevante.

Figura 6 – Ilustração da trajetória de um usuário com os três tipos de intenções as itenções: geográficas, temporais e semântica .




Fonte: (YING *et al.*, 2013)

Para identificar esses padrões (YING *et al.*, 2013) propõe uma abordagem, GTS-LP (*Geographic-Temporal-Semantic Location Prediction*), de predição de localização baseada em mineração de localização considerando as intenções citadas anteriormente. Para isso, ele define um novo padrão de trajetória para capturar as propriedades essenciais dos comportamentos de mobilidade dos usuários que são motivadas pelos três tipos de intenções. A predição tem

como base o cálculo da similaridade das informações semânticas, temporais e geográficas entre o movimento atual do usuário e os padrões anteriormente extraídos, e a partir do cálculo da similaridade a predição da próxima localização é, finalmente, realizada.

Finalmente exploraremos, sucintamente, o último tipo de predição de localização, predição de localização baseada nas redes sociais.

### *2.3.3 Predição de localização baseada em redes sociais*

As redes sociais baseadas em localização estão se popularizando cada vez mais nos últimos anos. Através de vários serviços e aplicativos (Forquare[2], Facebook[3], etc.) os usuários podem compartilhar suas localizações, atividades e informações específicas do local em que estão através dos *check-ins*, comentários e outras formas de interação fornecida pelas aplicações. A importância do compartilhamento dos dados de localização e informações sobre o local é tão importante que tem afetado o estilo de vida das pessoas. Por exemplo, o aplicativo *waze*[4], cujo serviço oferecido tem como dados de entrada o compartilhamento de informações do trânsito em tempo real pelos seus usuários, influencia, de forma instantânea, a rota de tráfego de outros usuários. Com o objetivo de melhorar os serviços oferecidos pelas redes sociais baseadas em localização é importante que hajam soluções capazes de prever a próxima localização que será visitada pelo usuário considerando, além do seus movimentos, os movimentos de trajetórias semelhantes aos seus e as informações semânticas compartilhadas, por exemplo através dos *check-ins*.

A predição de localização baseada em redes sociais, usualmente, parte do princípio de que as comunidades influenciam diretamente no próximo local de um usuário em movimento. E para identificar as comunidades na rede, considerando os movimentos dos objetos, é necessário construir as trajetórias dos usuários, identificar a similaridade entre essas trajetórias e descobrir as comunidades baseada nesses movimentos, sendo que tanto dados espaciais quanto temporais devem ser considerados na identificação da similaridade existente (ZHU *et al.*, 2014). Juntamente com as comunidades outra informação bastante explorada são os *check-ins*.

(YE *et al.*, 2013) adicionam em seu trabalho a informação dos *check-ins* durante o processo de identificação dos movimentos padrões dos usuários. Assim, além das informações espaciais e temporais dos usuários, as *categorias* dos *check-ins* também são utilizadas na predição

[2] https://foursquare.com
[3] https://www.facebook.com
[4] https://play.google.com/store/apps/details?id=com.waze&hl=pt_BR

da localização do usuário. Na abordagem proposta é utilizado uma Cadeia de Markov Oculta para modelar os movimentos padrões dos usuários e a dependência entre as categorias dos *check-ins*. Os estados da Cadeia de Markov Oculta são representados pelos locais que o usuário fez o *check-in* e também pelos locais considerados como pontos de interesse do usuário, informação identificada com base nos padrões comportamentais dos usuários. A predição do próximo local é realizada da seguinte forma: primeiro é identificada a categoria do *check-in* mais provável e, em seguida, o local mais provável da categoria do *check-in* é estimado, considerando o histórico dos *check-ins* e a trajetória atual com as informações espacial e temporal.

Em (NOULAS *et al.*, 2012) os *check-ins* também são fundamentais para prever o exato local para o qual o usuário irá. Para que a predição ocorra é necessário fornecer o histórico dos *check-ins* do usuário e os movimentos recentes da trajetória atual. O próximo passo é classificar os possíveis candidatos, sendo que os locais já visitados pelo usuário são os candidatos mais prováveis. Posteriormente, as informações temporais sobre os movimentos dos usuários são exploradas através de um conjunto de funcionalidades e a predição é realizada pelo *framework* de aprendizagem supervisionada.

## 2.4 Árvore de sufixo probabilística

(RON *et al.*, 1996) introduziram as noções de uma árvore de sufixo probabilística, PST (*Probabilistic Suffix Tree*). (BEJERANO; YONA, 2001) definem uma PST sobre a qual cada aresta é mapeada por um único símbolo do alfabeto, de modo que a partir do nó não há arestas com símbolos iguais, ou seja, o grau de cada nó é limitado pelo tamanho do alfabeto e a cada nó é atribuído um vetor de distribuição de probabilidade sobre o alfabeto.

Estas probabilidades correspondem aos símbolos contidos na consulta que foram observados antes, e foram utilizados na construção da PST. Na Figura 7 podemos observar o vetor de probabilidades associado às transições que podem ocorrer de um nó para outro sobre o alfabeto definido $\Sigma = \{a,b,c,d,r\}$. Por exemplo, considerando a subsequência **bra**, cujo vetor de probabilidades é [0,4; 0,2; 0,1; 0,1; 0,2] para os símbolos (a,b,c,d e r), respectivamente, há uma maior probabilidade de se observar o nó **a** depois desta subsequência. Quando a PST é utilizada para prever padrões significativos a partir da consulta de uma *string*, as probabilidades são, então, utilizadas. Cada nó em uma PST é marcado por um label, uma sequência que corresponde o caminho partindo do nó até o nó *root*. Por exemplo, o nó com rótulo **bra**, indica que o percurso **arb** foi realizado no sentido contrário, do nó *root* até o nó, conforme vemos na na Figura 7. Os

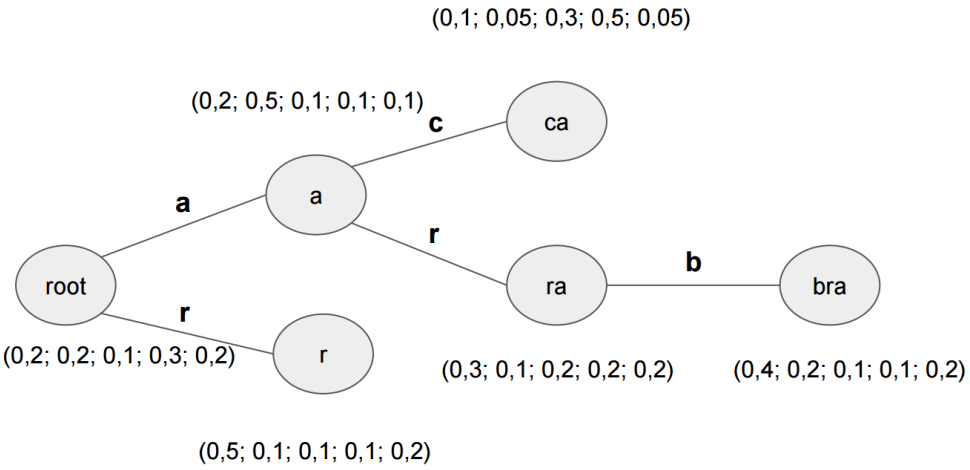
labels são necessários para identificar, por exemplo, a maior subsequência existente entre a *string* usada na consulta e os nós da árvore. Uma vez encontrado o nó mais similar as probabilidades podem ser utilizadas nos cálculos que seguem.

Observando a Figura 7 percebemos que o pai do nó **bra**, em uma PST, é o nó **ra**, ou seja, o pai de um nó em uma PST é representado pelo label do nó sem o primeiro símbolo. Se representássemos esse mesmo nó em uma de árvore de sufixo o pai dele seria o nó **br**, isto é, os labels dos nós de uma PST são o inverso dos labels mapeados em uma árvore de sufixo (BEJERANO; YONA, 2001). Uma árvore de sufixo é uma estrutura de dados semelhante a uma árvore onde todos os sufixos de uma *string* são mapeados (UKKONEN, 1995).

Nosso modelo preditivo, que representa os padrões de trajetórias de um indivíduo, é baseado na árvore de sufixo probabilística. Embora seja uma modelagem simples, nessa estrutura temos todas as informações necessárias para representarmos os padrões das trajetórias juntamente com os dados dos domínios espacial e temporal. Em nosso modelo os nós da árvore representam as regiões de interesse, células de parada, em que o objeto permaneceu uma quantidade suficiente de tempo e as arestas representam as transições entre as regiões e armazenam os dados referente as transições, como o número de transições de uma célula para outra, os horários em que as transições ocorreram, os horários de chegada naquela nó e o tempo de permanência naquela célula.

(LEI *et al.*, 2013) criaram uma variante da estrutura de dados PST, denominada STT (*Spatial-Temporal Trajectory Model*) para representar as regiões frequentes de uma trajetória. (ZHU *et al.*, 2014) proporam uma estrutura de dados de árvore de decisão, semelhante a PST, a estrutura de dados definida é denominada Árvore de Probabilidade Sequencial. A árvore é usada para identificar os padrões de trajetórias dos usuários e identificar grupos de usuários que pertencem a um mesmo cluster com base na semelhança entre movimentos dos usuários. Árvores de sufixo probabilística podem ser aplicadas em outros domínios de aplicação para identificação de padrões que não sejam apenas a identificação de padrões de trajetórias. Em (BEJERANO; YONA, 2001) as PSTs foram utilizadas para representar sequências de padrões de proteínas. A estrutura criada é utilizada durante o processo de classificação da família dessas proteínas, considerando as proteínas já conhecidas e classificadas.

Figura 7 – Ilustração de uma PST sobre o alfabeto Σ = {a,b,c,d,r} juntamente com o vetor de probabilidades associado as transições para os nós na ordem a, b, c, d e r.




Fonte: (BEJERANO; YONA, 2001)

## 2.5 Algoritmo do máximo sufixo comum

O algoritmo do máximo sufixo comum é também conhecido como algoritmo da máxima substring comum. O objetivo deste algoritmo é identificar a máxima substring comum entre duas ou mais *strings*. (BERGROTH *et al.*, 2000) resolveu o problema para identificar a máxima *substring* comum utilizando programação dinâmica e usaremos a mesma abordagem para solucionar esse problema. Programação dinâmica é um método para resolver problemas complexos subdividindo o problema maior em problemas menores. É comum utilizar alguma estrutura para armazenar as soluções dos problemas menores para evitar que estes subproblemas sejam novamente computados. Exemplificando o algoritmo, de forma bem sucinta, podemos por exemplo, computar que a máxima substring comum dentre as *strings* **DABCD** e **CBABCBCE** é a *string* **ABC**.

Em predição de trajetórias consideramos que os movimentos mais recentes possuem maior prioridade quando comparados com os movimentos que ocorreram anteriormente, por isso, são vários os exemplos de trabalhos que privilegiam os movimentos mais recentes, exemplo disso é que há algumas abordagens que exploram as Cadeias de Markov e suas variantes na predição de localização, por exemplo (GAMBS *et al.*, 2012).

Nós utilizamos o algoritmo do máximo sufixo comum em nossa abordagem para identificarmos o nó mais similar à trajetória atual priorizando os movimentos mais recentes. Para identificar o nó mais similar toda a árvore é percorrida e durante a consulta de cada nó, extraímos o maior sufixo comum entre a consulta e o label nó consultado, e calculamos a similaridade do movimento recente (consulta) e o nó que está sendo consultado. O nó com a maior similaridade é

escolhido para realizarmos a predição a partir deste. O cálculo da similaridade é feito aplicando a fórmula definida na Seção 4.4.1.

## 2.6 Histogramas

Os histogramas são uma representação gráfica da distribuição de dados numéricos e seu uso é bastante utilizado quando se deseja reduzir o volume de dados estudados (HAN; KAMBER, 2000). Para construir um histograma é necessário fazer a distribuição de frequência de um ou mais atributos e nessa distribuição de frequência não há intersecção dos intervalos. Um dos objetivos do uso de histograma é extrair a frequência em que os atributos representados no histograma ocorrem. Por exemplo, na Figura 8 utilizamos um histograma para representar e visualizar o número de professores em uma universidade, cujo intervalo da faixa etária é de 10 anos.

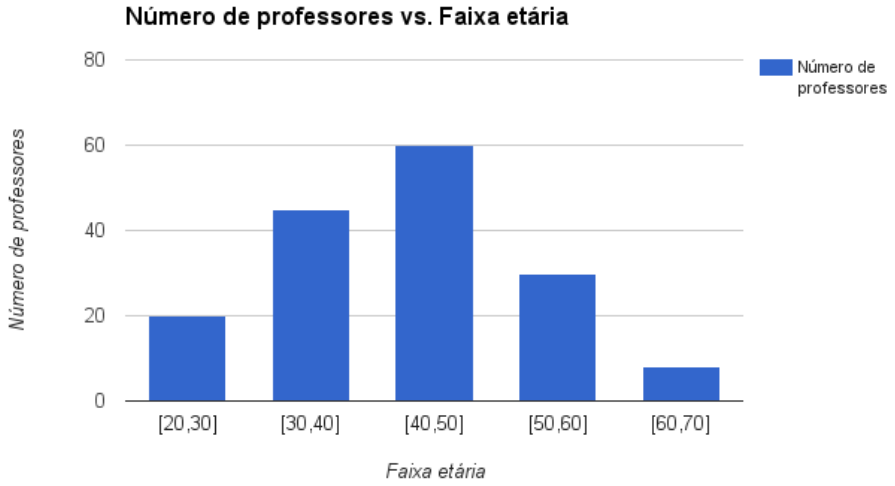
Os tipos mais comuns de histogramas são *Equal-width*, sobre o qual os atributos são divididos em *N* intervalos de tamanhos iguais, assim, os histogramas podem ter uma quantidade de dados totalmente diferente entre si; e os histogramas *Equal-depth* no qual o histograma é dividido em *N* intervalos de tal forma que os elementos dispostos em cada histograma sejam aproximadamente o mesmo (APPEL, 2010). Como podemos perceber a Figura 8 é um exemplo de um histograma *Equal-width*.

Nosso *framework* utiliza um algoritmo de clusterização temporal sobre os tempos de saída definidos nas transições entre uma célula de parada e outra. Por isso construímos os histogramas considerando os intervalos de tempo, definidos em minutos, e o número de ocorrências em que os tempos de saídas encontrados nas transições de uma região de interesse para outra ocorrem dentro do intervalo delimitado. A fase do algoritmo de clusterização que é responsável por construir os histogramas e a forma como estes são construídos é detalhada na Seção 4.3.2.5.

## 2.7 Resumo

Nesse capítulo procuramos introduzir alguns conceitos necessários para uma melhor compreensão do trabalho que propomos. As trajetórias são essenciais para realizarmos a predição de localização e dentre os tipos de predição de localização existente nosso trabalho se enquadra naquele cuja predição é realizada sobre os padrões de trajetória de único indivíduo, e para isso

Figura 8 – Exemplo de um histograma representando a quantidade de professores em uma universidade.




Fonte: elaborada pelo autor

propomos um modelo preditivo para representar esses padrões através das árvores de sufixo probabilística, que como vimos serve como uma ótima estrutura para identificação de padrões de trajetória e também nos fornecer as informações necessárias para predição de localização.

# 3 TRABALHOS RELACIONADOS

Vários trabalhos de pesquisa com foco em mineração de padrões de trajetória utilizam técnicas de mineração de padrão sequencial para descobrir, a partir de um histórico de trajetórias, um conjunto de regiões frequentes. Essas regiões são analisadas a fim de descobrir as relações existentes entre elas e o comportamento dos movimentos dos objetos. (GIANNOTTI *et al.*, 2006) propôs um algoritmo para minerar os padrões sobre um conjunto de sequências temporariamente anotadas, onde cada transição da sequência é anotada com algum tempo característico procedente da base de dados que está sendo analisada, por exemplo, as trajetórias de um objeto em movimento que se desloca entre localizações relevantes. (MONREALE *et al.*, 2009) utiliza o algoritmo proposto por (GIANNOTTI *et al.*, 2006) para extrair os padrões dos movimentos das trajetórias de objetos em movimento e com base nesses padrões capturados realizar a predição da localização. Nesse trabalho a consulta utilizada para representar uma requisição de predição de localização é constituída, apenas, dos movimentos recentes do usuários e a informação temporal associada não é explorada como deveria na computação da predição, porém o dado temporal é utilizado na fase de construção da árvore dos padrões da trajetórias, como foi citado. A semelhança do nosso trabalho (MONREALE *et al.*, 2009) propõe que os padrões de mobilidade de um objeto, ou seja seu modelo preditivo, seja representado utilizando uma árvore como estrutura de dados.

Em (XUE *et al.*, 2013a) e (XUE *et al.*, 2013b) os autores propõem um método de predição de localização que lida com o problema de dispersão de dados. Esse problema ocorre quando a consulta para predição de localização de um objeto em movimento contém regiões de interesses que não estão mapeadas em seus padrões de mobilidade. A fim de lidar com esta problemática, os autores decomponhem todas as trajetórias em sub-trajetórias compostas de dois vizinhos locais; depois, as sub-trajetórias são adicionadas às trajetórias "sintetizadas". Enquanto que a trajetória de consulta corresponde a qualquer parte da trajetória sintetizada, e o destino da trajetória sintetizado é utilizado como a localização predita. Uma demonstração deste trabalho é apresentado em (XUE *et al.*, 2013a).

Outra linha interessante de trabalho que utiliza árvore de sufixo probabilística para representar os padrões de locomoção dos objetos é o trabalho proposto por (LEI *et al.*, 2013). Nesse trabalho, os autores propõe um framework chamado *QS-STT* (*QuadSection clustering and Spatial-Temporal Trajectory model*) que captura os comportamentos padrões de mobilidade dos objetos e realiza as predições de localização com granularidade de um único objeto. O

framework *QS-STT* possui duas macro atividades que são: i) construir o modelo de predição, que representa o comportamento de mobilidade do objeto em movimento, e ii) executar o algoritmo de predição de localização a partir da consulta informada. Para construir o modelo preditivo é realizado um processo de clusterização para extrair o melhor conjunto possível de regiões frequentes. Essas regiões espaciais são identificadas quando o número de segmentos que passam por elas é igual ou superior a um limite mínimo de segmentos. Esse é um parâmetro necessário para identificar as regiões relevantes e consequentemente a construção do modelo preditivo. Por exemplo, serão consideradas como regiões relevantes apenas as regiões cujo o número mínimo de segmentos de trajetórias que passam por essas regiões seja 5 (cinco). Assim é necessário que no mínimo 5 (cinco) segmentos de trajetórias tenham atravessado uma dada região, para que ela considerada como uma região de interesse para o objeto em movimento.

Posteriormente, os autores propõem um modelo de trajetória espaço temporal, denominado *STT*. Esse é modelo é construído com base nas regiões frequentes identificadas na fase de clusterização, o objetivo é explorar os padrões dos movimentos do objeto e representá-los em uma árvore de sufixo probabilística contendo as informações espacial e temporal. E finalmente, para realizar a predição de localização eles propõem um algoritmo que percorre o modelo preditivo, encontra o nó mais similar, quando comparado a consulta que representa os movimentos mais recentes do objeto, e computa a predição a partir do nó similar identificado. Os resultados experimentais mostram que o framework *QS-STT* é capaz de capturar ambos os padrões espaciais e temporais de comportamentos de mobilidade, e mostra uma performance melhor quando comparado aos *baselines* utilizados no artigo.

A solução proposta por (LEI *et al.*, 2013) demonstra ser eficaz e eficiente para a previsão de localização, por esta razão, nós estendemos o modelo *STT* para incorporar novas funcionalidades a fim de resolver o problema apresentado em nosso trabalho; prever não apenas a próxima localização relevante do objeto, mas prever também o tempo estimado em que o objeto deixará sua localização atual.

Uma vez que nosso trabalho é parcialmente inspirado em (LEI *et al.*, 2013), se faz necessário elencar e destacar as principais diferenças entre ambos. As diferenças que mais se destacam são:

- A resolução das células é um parâmetro de entrada em nosso framework para mapearmos as células e definirmos as regiões de interesse. O QS-STT identifica a resolução das células, regiões, considerando o número máximo de regiões que são atravessadas pelo número

mínimo de segmentos que definem as regiões de interesse.Essa abordagem aumenta o tempo de processamento da construção do modelo preditivo.

- Nossas regiões frequentes, células de parada, são definidas como aquelas que atendem as especificações detalhadas na Definição 4.1.5, onde os principais componentes utilizados são os limites temporais e a resolução. No *QS-STT* as regiões frequentes são definidas na etapa de clusterização e não são utilizados dados temporais para identificá-las.
- A probabilidade espacial calculada por nosso algoritmo de predição não é a probabilidade condicional, consideramos apenas o número de transições da célula pai para célula filha e o número total de transições partindo do nó pai, com estas informações temos o que é necessário para computarmos as probabilidades espaciais de transição entre o nó pai e seus filhos. No *QS-STT* o cálculo da probabilidade espacial é realizado utilizando a probabilidade condicional sendo necessário mais processamento para computar a predição, uma vez que a probabilidade condicional efetua mais operações do que a probabilidade simples.
- Exploramos as características temporais tanto na construção do modelo preditivo quanto na fase na predição. Primeiramente utilizamos os dados temporais para identificarmos as localizações relevantes, células de parada. Na predição, o tempo de consulta da predição é fundamental, pois a partir dele identificamos por meio do algoritmo de clusterização temporal o tempo mais representativo e utilizamos esse tempo encontrado no cálculo da probabilidade temporal.
- Ao estendermos o problema clássico de predição de localização precisamos de uma maneira eficaz que explore os dados temporais armazenados na coleta dos dados das trajetórias e para isso introduzimos os conceitos de *ciclo temporal* e *partições temporais*, além dos limites temporais definidos na identificação das regiões frequentes, para alavancar as predições realizadas pelo *TPRED*.

## 3.1 Resumo

Um dos principais diferenciais de nosso trabalho foi explorar os dados temporais tanto na construção do modelo preditivo quando na predição da localização. No modelo preditivo definimos os conceitos de *ciclo temporal* e *partições temporais*, e na predição identificamos o tempo mais representativo próximo ao tempo de consulta da predição. Embora nossa proposta tenha como estado da arte o trabalho proposto por (LEI *et al.*, 2013) criamos um modelo

próprio para identificar e representar os padrões dos movimentos dos objetos e vemos diferenças significativas quando comparamos os dois *frameworks*. Outro fator relevante em nosso trabalho é a devida relevância dada aos dados temporais que foram tão bem explorados que, juntamente com os dados espaciais, contribuem para uma melhor performance quando comparamos com outros dois *baselines*.

# 4 METODOLOGIA

Com o propósito de resolvermos o problema descrito na seção 4.2 precisamos apresentar um conjunto de definições básicas necessárias para descrevermos formalmente os cenários que consideramos necessários para prevermos a próxima localização revelante de um objeto em movimento e quando esse objeto partirá de sua localização atual.

## 4.1 Definições básicas

Para responder aos dois questionamentos expostos no problema, seção 1.1, nós precisamos de um modelo preditivo para representar os movimentos padrões, em ambos os domínios espacial e temporal, a fim de permitir que as predições realizadas sejam confiáveis e significativas com base nos padrões de mobilidade capturados a partir de uma análise do histórico dos movimentos passados de um objeto. O primeiro desafio encontrado é o de analisar os movimentos passados de um objeto, (i) encontrar suas localizações relevantes e (ii) quanto tempo o objeto passa em cada localização relevante. Em seguida fornecemos um conjunto de definições básicas sobre as quais podemos criar ferramentas úteis que são capazes de explorar as informações capturadas a partir dos dados das trajetórias. Notamos que a parte de noções introdutórias são semelhantes às introduzidas por (SPACCAPIETRA *et al.*, 2008).

**Definição 4.1.1 (Trajetória de um objeto em movimento)** *Definimos que uma trajetória de um objeto u em movimento é uma sequência de amostras organizada temporalmente,* $T = \langle s_1, \cdots, s_n \rangle$, $s_i = (l_i, t_i)$, $t_1 \leq \cdots \leq t_n$, *onde* $l_i = (x_i, y_i)$ *representa a localização espacial do objeto no tempo* $t_i$.

Ao lidar com dados do mundo real, há a possibilidade de os movimentos passados de um objeto serem representados por múltiplas trajetórias não sobrepostas temporalmente. Em tais casos, é conveniente supor que estas trajetórias possam ser transformadas em uma única trajetória.

Considerando que nós precisamos descobrir os locais mais relevantes para um objeto, e admitimos que as regiões mais significativas são aquelas em que os objetos passam uma maior quantidade de tempo, precisamos planejar uma maneira de limitar as regiões do espaço onde isso realmente acontece. Para atingirmos nosso objetivo a ideia é sobrepor algum tipo de grid sobre o mundo $\mathscr{W}$, assim o mundo será discretizado em um conjunto de células disjuntas, denotada por $\mathscr{C}$. O problema, então, torna-se em encontrar as células $c \in \mathscr{C}$ que são relevantes para um objeto.

Ressaltamos que para identificar os locais relevantes para cada objeto, não podemos recorrer aos pontos de interesse conhecidos (POIs), como em outros cenários; pois em nosso cenário os locais relevantes podem variar de um objeto para outro.

Primeiro, vamos definir uma função que mapeia os locais dos objetos para células na grid, tirando proveito de algumas formas de discretização do mundo.

**Definição 4.1.2 (Função de mapeamento da célula)** *Dado o mundo $\mathscr{W}$ e uma decomposição de $\mathscr{W}$ em um conjunto de células disjuntas $\mathscr{C}$, uma função de mapeamento celular $f : \mathbb{R}^2 \to \mathscr{C}$ associa um par de coordenadas $l = (x, y)$, sobre o mundo $\mathscr{W}$, a uma célula $c \in \mathscr{C}$.*

O tamanho das células deve ser escolhido de acordo com a natureza dos movimentos do objeto considerado. Por exemplo, se o objetivo é prever uma localização cuja região deve ser geograficamente mais precisa, então o tamanho da célula deve ser menor. Mas se é aceitável prever uma localização com uma menor precisão geográfica então a resolução da célula pode ser maior. Também precisamos atribuir a cada célula em $\mathscr{C}$ um identificador único que a represente, isto é, precisamos enumerar as células de $\mathscr{C}$. Para isto o próximo passo é mapear qualquer par de coordenadas geográficas com o identificador da célula relacionado a esta posição.

**Definição 4.1.3 (Função de enumeração da célula)** *Dado o mundo $\mathscr{W}$ e uma decomposição de $\mathscr{W}$ em um conjunto de células disjuntas $\mathscr{C}$, uma função de enumeração celular $g : \mathscr{C} \to \mathbb{N}_0$ é uma função que associa qualquer $c \in \mathscr{C}$ com um número inteiro representando um identificador único.*

Notamos que diferentes funções podem ser usadas para implementar $g$, por exemplo, pode-se utilizar uma função linear simples, algum tipo de função espacial para esse fim, dependendo das necessidades específicas. Graças as definições de $f$ e $g$, podemos introduzir a noção de trajetória transformada.

**Definição 4.1.4 (Trajetória transformada de um objeto em movimento)** *Dado um objeto $u$ e uma trajetória descrevendo os seus movimentos, $T_u$, definimos que uma trajetória transformada de $u$ é uma sequência de amostras transformadas temporalmente ordenadas $T' = \langle s'_1, \cdots, s'_n \rangle$, onde $s'_i = (g(f(l)), t)$ e $g(f(l))$ representa o identificador da célula sobre a qual $u$ está na posição $l$ no tempo $t$.*

De agora em diante podemos simplificar e denotar $c = g(f(l))$, e introduzirmos a noção de *célula de parada*, que nesse contexto é equivalente a noção de *localização relevante*.

**Definição 4.1.5 (Célula de parada)** *Dada uma trajetória transformada de um objeto em movimento u, $T'_u$, um limite temporal de permanência $\sigma$, e um limite temporal inter-amostra, $\delta$, definimos que uma célula $c \in \mathscr{C}$ é uma célula de parada, quando o tempo de permanência de u em c é igual ou superior a $\sigma$, e a diferença máxima entre o par de duas amostras consecutivas é menor ou igual a $\delta$. Em outras palavras, é possível descobrir uma sequência consecutiva de amostras $S'_u \subseteq T'_u$, onde $S'_u = \langle s'_1, \cdots, s'_j \rangle$ e $s'_i = (c_i, t_i)$, mantendo as seguintes condições:*

$$\begin{aligned} &c_1 = c_2 = \cdots = c_j,\ t_j - t_1 \geq \sigma \text{ (i) condição de permanência} \\ &\quad \forall i,\ 1 \leq i < j,\ t_{i+1} - t_i \leq \delta \text{ (ii) condição de consistência} \end{aligned} \tag{4.1}$$

A condição de permanência garante que um objeto em movimento está permanecendo em uma determinada célula gastando uma quantidade significativa de tempo de acordo com o limite temporal $\sigma$. O valor delimitador de permanência, $\sigma$, é escolhido em função das características dos objetos. Por exemplo, objetos que possuem pouco deslocamento tendem a ter um valor temporal $\sigma$ maior do que objetos que se locomovem muito. Por exemplo, se o objeto em questão for uma pessoa e se a atribuição desse valor se dá em função de sua profissão, professores terão um limite temporal de permanência maior do que representantes comerciais, visto que estes se locomovem mais do que aqueles no exercício de suas atribuições. A condição de consistência garante que há informação suficiente nos dados para inferir que *u* está efetivamente permanecendo nessa célula. Esta condição é fundamental, pois em sua ausência haveria a possibilidade de termos *pseudo* células de parada. Por exemplo, se os dados de uma trajetória fossem coletados muito distantes entre si quando fosse aplicado a condição de permanência, essa seria satisfeita, mas pelo fato de as amostras terem sido coletadas em intervalos de tempo muito distantes entre si, poderia haver um deslocamento da "célula de parada" para outra célula qualquer, e isto não seria identificado se aplicássemos apenas a condição de permanência. Nós denotamos o conjunto de *células de parada* relacionadas a um objeto em movimento *u* por $C^s_u \subseteq C$.

Espera-se que um objeto em movimento, tipicamente repita seus padrões de trajetória ao longo do tempo (por exemplo, ao longo de um dia, uma semana, um mês), consequentemente, é possível que um objeto chegue e deixe uma única célula de parada várias vezes durante um intervalo de tempo fixo. Por exemplo, considerando objetos em movimento representados por seres humanos e períodos relacionados aos dias úteis (de segunda a sexta-feira), nós provavelmente vamos observar que a maioria dos objetos saem de suas casas para seus locais de trabalho

durante o início da manhã e retornem ao mesmo local, suas casas, no final da tarde. Para termos predições mais precisas é fundamental capturar esses detalhes no domínio temporal e a próxima definição trata a respeito disso.

**Definição 4.1.6 (Tempo de partida e de chegada de uma célula de parada)** *Dado que um objeto em movimento que permanece em uma célula de parada c é representado por uma sequência de amostras $S'_u = \langle s'_1, \cdots, s'_j \rangle$, representando sua chegada, $s'_1$, e sua partida, $s'_j$, dentro do intervalo $j$ e $s'_i = (c, t_i)$, definimos que $t_1$ é o tempo de chegada de u na célula de parada c e $t_j$ representa o tempo de partida em que u deixa a célula de parada c.*

Uma vez que somos capazes de detectar as células de parada, e num passo futuro determinar quando temos as transições entre as células, ou seja, descobrimos as subsequências contíguas de amostras dentro de $T'u$, cujos identificadores das células são os mesmos das células de parada e relacionar essas subsequências umas com as outras. Para este fim, que introduziremos o conceito de *limite temporal de transição inter-celular*, que é essencialmente a condição de consistência expressada na 4.1.5, mesmo sendo aplicada as transições (movimentos) ao invés da permanência em si.

**Definição 4.1.7 (Limite temporal de transição inter-celular)** *Suponhamos que temos a trajetória transformada $T'_u$. Vamos supor também que $T'_u$ foi criada a partir da seguinte subsequências de amostras:*

$$\begin{aligned} &\langle (id_X, t_1), \cdots, (id_X, t_n), \\ &(id_k, t_{n+1}), \cdots, (id_j, t_{n+m}), \\ &(id_Y, t_{n+m+1}), \cdots, (id_Y, t_{n+m+q}) \rangle, \end{aligned} \tag{4.2}$$

*onde assumimos que $id_X$ e $id_Y$ são as únicas células de parada das subsequências acima. Em seguida, é dado um limite temporal $\tau$, dizemos que há uma transição de $id_X$ para $id_Y$, ou que u se moveu de $id_X$ para $id_Y$, somente se a condição abaixo é satisfeita:*

$$\forall j \in [n, n+m],\ t_{j+1} - t_j \leq \tau.$$

*Nós chamamos $\tau$ de limite temporal de transição inter-células.*

Esta definição é bem similar a definição da condição de consistência expressa na definição 4.1.5. Porém na *condição de consistência* estamos interessados no intervalo entre as amostras que definem um célula $c$ em uma célula de parada, e nesta definição estamos

interessados em identificar se as transições entre as células de parada ocorreram dentro do limite determinado. Isso possibilita tratar fluxos de exceções quando o deslocamento entre uma célula de parada e outra ocorreram muito distantes entre si. Por exemplo, se o objeto em movimento for uma pessoa e se esta realizasse uma viajem de avião entre SP (São Paulo) e CE (Ceará), sem esta definição seria possível mapear no modelo preditivo uma transição entre células de parada em estados muito distantes. Sabemos que reconhecer esse padrão pode ser útil, dependendo da aplicação ou serviço, e para isto faz-se necessário configurar corretamente $\tau$.

## 4.2 Definição do problema

Considerando os cenários que expomos, a disseminação de meios habilitados com GPS *tracker* e a facilidade de acesso a internet através desses meios, é razoável supor que um objeto em movimento $u$ possui uma trajetória que representa os seus padrões de mobilidade, denotada por $T_u$, e também uma trajetória, denotada por $T_u^H$, que representa seus deslocamentos mais recentes, sobre a qual devemos extrair as informações necessárias para realizar uma predição. Assim podemos formalizar o problema da predição de localização como segue.

**Definição 4.2.1 (Problema de Predição de Localização)** *Dado um objeto em movimento u, uma trajetória, denotada por $T_u$, que representa os movimentos passados de u, uma trajetória, denotada por $T_u^H$, que representa os movimentos mais recentes de u, sendo que esta trajetória não foi utilizada para compor o padrão de mobilidade composto pelos movimentos passados de u, e o tempo de consulta $t_{now}$, nós desejamos computar a consulta, $q(T_u^H, t_{now})$, que prevê quando u deixará sua localização atual e qual será sua próxima localização relevante considerando o histórico dos movimentos passados de u.*

Nas próximas seções vamos introduzir o modelo preditivo e o algoritmo preditivo proposto para solucionar os dois problemas apresentados na Definição 4.2.1. O modelo preditivo é responsável por capturar os padrões dos movimentos relevantes de um único objeto em movimento, aproveitando-se do histórico dos movimentos passados e das informações relacionadas a estes movimentos, tanto no domínio espacial quanto temporal. E o algoritmo preditivo deve explorar o modelo preditivo para computar as consultas de predições requisitadas.

A seguir, apresentamos o projeto e a construção do modelo preditivo, e posteriormente esboçamos o algoritmo de previsão.

## 4.3 Modelo preditivo

Com o objetivo de resolvermos os problemas apresentados na definição 4.2.1, nós precisamos extrair os padrões dos movimentos de um objeto considerando o histórico dos seus movimentos passados. Mais especificamente, precisamos (i) descobrir os locais relevantes (células de parada), onde um objeto em movimento permanece quantidades relevantes de tempo, e (ii) capturar as relações existentes entre suas células de parada, ou seja, descobrir como essas células estão interligadas, por meio das transições e quando tais transições ocorrem ao longo do tempo. Para representar os padrões de mobilidade citado, nós exploramos árvores de sufixo probabilística, apresentada na seção 2.4, e que demostrou ser uma ferramenta muito valiosa ao ser utilizada em uma problemática semelhante (LEI *et al.*, 2013).

Para identificarmos as células de parada, principalmente em âmbito espacial, nós exploramos as noções introduzidas nas definições 4.1.5 e 4.1.6. Para identificação daquelas no âmbito temporal, observamos as definições 4.1.6 e 4.1.7, noções temporais adicionais que ajudam a identificar a ciclicidade através dos comportamentos de mobilidade que se repetem e se revelam ao longo do tempo.

### *4.3.1 Ciclo temporal e partições temporais*

Uma vez que os movimentos padrões dependem fortemente dos hábitos dos objetos em movimento, primeiro precisamos considerar a ciclicidade de como esses padrões se repetem ao longo do tempo. De acordo com os tipos de movimentos dos objetos considerados, pode-se desejar estudar os padrões dos movimentos sobre os ciclos temporais que abrangem um dia, uma semana, um mês, um ano e assim por diante. E dentro de um ciclo pode-se querer particionar o intervalo de tempo relacionado em intervalos temporais disjuntos. Por exemplo, sempre que queremos estudar os padrões de movimentos que ocorrem em uma semana, é conveniente separar os padrões que ocorrem nos dias úteis daqueles que ocorrem nos finais de semana, uma vez que estes podem ser muito distintos; por exemplo, se uma pessoa geralmente faz o percurso casa-trabalho durante os dias da semana é provável que ela faça outros percursos durante os finais de semana. A seguir apresentaremos a noção de ciclo *ciclo temporal* e *partições temporais* dentro de um ciclo.

**Definição 4.3.1 (Ciclo temporal e partições temporais)** *Consideremos o conjunto de todos os timestamps possíveis, Time. Vamos considerar também um conjunto finito de símbolos,*

$\{Time_1^+, \cdots, Time_n^+\}$*, cada um representando uma entidade temporal (e.g., uma janela de tempo, um dia genérico, uma semana genérica). Denominamos esse conjunto união*

$$Time^+ = \cup_{i=1}^{n} Time_i^+ \tag{4.3}$$

*ciclo temporal, enquanto cada partição* $Time_i^+$ *é denominada de partição temporal dentro do ciclo.*

Assim definimos a função $temp : Time \rightarrow Time^+$ para ser a função de custo de associação dos *timestamps Time*s para os $Time^+$s símbolos.

É claro que os símbolos que compõem $Time^+$ devem ser escolhidos de modo que a semântica associada ao ciclo temporal seja consistente. Por exemplo, vamos considerar o ciclo temporal de uma semana. Podemos, por exemplo, particionar o ciclo em dias individuais, da seguinte forma, $Time^+ = \{Seg, Ter, Qua, Qui, Sex, Sab, Dom\}$, ou particionarmos o ciclo em dias úteis e finais de semana, i.e., $Time^+ = \{DiasDaSemana, FinaisDeSemana\}$.

### *4.3.2 Modelo da árvore de sufixo probabilística*

Nesta seção, vamos esboçar a nossa variante da árvore de sufixo probabilística (PST) usada para representar o modelo preditivo, as estruturas de dados fundamentais e o algoritmo usado para construí-las. Neste contexto, dado um objeto em movimento $u$, o histórico de movimentos passados $T_u$ e um ciclo temporal $Time^+$, o objetivo de uma PST é representar os padrões dos movimentos mais comuns de $u$ de forma que $T_u$ se relacione com $Time^+$. Em seguida, vamos descrever como projetamos nossa variante de árvore de sufixo probabilística e consideraremos a trajetória transformada, $T_u'$, obtida a partir de $T_u$.

#### *4.3.2.1 Estrutura da PST*

Em nosso contexto, definimos uma PST $\mathscr{T}$ como uma árvore que representa um conjunto de *movimentos padrões* através das células de parada. Como tal, cada aresta na árvore é rótulada por uma célula de parada e representa a transição de uma célula de parada para outra, possivelmente repetida, durante uma partição temporal específica. Cada nó $\mathscr{T}$, com exceção do nó root, é rotulado com o caminho do percurso realizado do nó até o nó root. Por exemplo, se o rótulo de um nó é $c_3, c_2, c_1$ representa que o percurso $root \rightarrow c_1 \rightarrow c_2 \rightarrow c_3$ foi realizado. O nó é assim rotulado para que seja otimizada a atividade para identificar o nó mais similar

durante uma consulta de predição. Cada nó possui uma tabela preditiva que possuem os dados necessários para calcular a probabilidade de transição entre as células de parada. A relação entre o nó e a célula de parada representada na tabela preditiva descreve a observação de pelo menos uma transição a partir do nó, para a próxima célula de parada, abrangendo a partição temporal especificada. Na sequência detalhamos a semântica associada aos nós e as arestas, bem como as informações associadas a eles para que previsões sejam realizadas numa fase posterior.

#### *4.3.2.2 Arestas da PST*

Baseado nos dados em $T'_u$, uma aresta que conecta um par de nós representa a observação, de possivelmente, várias transições do nó pai para o nó filho, ambos os nós associados aos seus respectivos percursos realizados. Cada aresta possui um rótulo que representa a transição de uma célula de parada para outra.

**Definição 4.3.2 (Rotulagem das transições da PST)** *Seja $T'_u$ a trajetória transformada associada ao objeto em movimento u. Dado também $\langle (id,t_1),\cdots,(id,t_n)\rangle$ como a sequência de amostras que descrevem a permanência de u dentro de uma célula de parada, de acordo com a Definição 4.1.5. Então, supondo que a sequência de amostras ocorre dentro da partição temporal $Time_i^+$, o rótulo, label, de uma aresta que representa que u realizou uma transição da célula de parada id para outra durante $Time_i^+$, é $(id, Time_i^+)$.*

Por exemplo, se um objeto em movimento *u* está permanecendo em uma célula de parada, cujo identificador é 7, na segunda-feira, e assumindo que estamos usando um ciclo temporal semanal $Time^+ = \{Seg, Ter, Qua, Qui, Sex, Sab, Dom\}$, então $\mathscr{T}$ terá uma transição com label $(7, Seg)$. Esta transição pode ser representada nas arestas e/ou na tabela preditiva, como podemos ver na Figura 9.

#### *4.3.2.3 Tipos de nós da PST*

Cada nó $\mathscr{T}$, com exceção do nó root, representa o percurso realizado por um objeto em movimento, que retrata uma sequência de transições entre as células de parada dentro de uma partição temporal. Nós definimos três tipos diferentes de nós:

- *Nó root*: É um nó especial que representa o *ponto de entrada* da árvore, ele não se refere a nenhum percurso, mas se relaciona com os nós, através das arestas, que representam o início da sequência, transições, por meio das células de parada.

- *Nós internos*: Cada nó interno representa o percurso de um objeto em movimento em uma célula de parada específica dentro de uma partição temporal; é sempre o filho de um nó interno ou do nó root, e pai de múltiplos nós internos ou folhas.
- *Folhas*: As folhas possuem as mesmas propriedades dos nós internos, mas com uma diferença, elas não possuem filhos. Conseqüentemente uma folha representa o penúltimo ponto final de uma sequência de transições, uma vez que o último ponto, última célula de parada de um percurso, é representada na tabela preditiva.

A posição de um nó na árvore depende das sequências de transições das células de parada observadas a partir de $T'_u$, sobre como as transições ocorrem ao longo do tempo e como se relacionam umas com as outras. Por exemplo, a mesma célula de parada, associada a mesma partição temporal, pode aparecer em vários nós, dependendo das sequências de transições observadas a partir dos dados dos movimentos (i.e, uma transição que representa o trajeto casa-trabalho-casa e outra que representa o trajeto casa-universidade-casa). Uma vez que cada nó expressa o sufixo do percurso, em uma partição temporal, das transições observadas entre as células de parada, nós precisamos de uma convenção de rotulagem para expressarmos a combinação destas informações.

**Definição 4.3.3 (Rotulagem dos nós da PST)** *Seja $T'_u$ a trajetória transformada associada ao objeto em movimento u. Dado também a seguinte subsequência de amostras de $T'_u$ que descrevem os movimentos de u:*

$$\begin{aligned} &\langle (id_X, t_1), \cdots, (id_X, t_n), \\ &(id_k, t_{n+1}), \cdots, (id_j, t_{n+m}), \\ &(id_Y, t_{n+m+1}), \cdots, (id_Y, t_{n+m+q}) \rangle, \end{aligned} \tag{4.4}$$

*e assumindo que $id_X$ e $id_Y$, são as únicas células de parada de acordo com a Definição 4.1.5. Então, supondo que a subsequência de amostras ocorre dentro da partição temporal $Time_i^+$; o rótulo do nó que representa o percurso root $\rightarrow id_X \rightarrow id_Y$ dentro de $Time_i^+$ é, $(id_Y, Time_i^+)$,$(id_X, Time_i^+)$.*

Como foi mencionado anteriormente, cada nó em $\mathscr{T}$ é rotulado com o caminho do percurso realizado do nó até o nó root, ou seja, caminho inverso ao realizado do root até o nó. Mapear o caminho inverso é uma característica intrínseca às arvores de sufixo probabilística permitindo que a pesquisa do caminho mais similar, mapeado em $\mathscr{T}$, quando comparado aos movimentos recentes de um objeto, seja identificado mais rapidamente.

Notamos que a permanência de um objeto em movimento, dentro de uma única célula de parada, pode abranger múltiplas partições temporais sucessivas. Por exemplo, as pessoas normalmente dormem em casa durante as noites e se usarmos $Time^{+} = \{Seg, Ter, Qua, Qui, Sex, Sab, Dom\}$, provavelmente observaremos duas partições, dentro da mesma célula de parada, abrangendo duas partições temporais consecutivas e distintas. Para esses casos, criamos os trajetos na PST com todas as células de parada na mesma partição temporal, ou seja, as células no trajeto concordam com as partições temporais abrangidas, assim, em um mesmo trajeto só há células de parada de uma mesma partição temporal. Por exemplo, supondo que uma pessoa realizou o trajeto *(casa, Seg) (trabalho, Seg) (casa)*, e uma vez que a pessoa dormiu em sua *casa*, os dados de sua última localização relevante contém dados de duas partições temporais, *Seg* e *Ter*, porém, os dados do trajeto *(casa, Seg) (trabalho, Seg) (casa)* devem está na mesma partição temporal, por isso, o trajeto realizado foi concluído ao fim da partição temporal *Seg*, logo teremos o trajeto *(casa, Seg) (trabalho, Seg) (casa, Seg)*. E um novo trajeto, que ocorrerá na próxima partição temporal *Ter*, será iniciado na célula de parada *(casa, Ter)*.

#### *4.3.2.4 Tabela preditiva dos nós da PST*

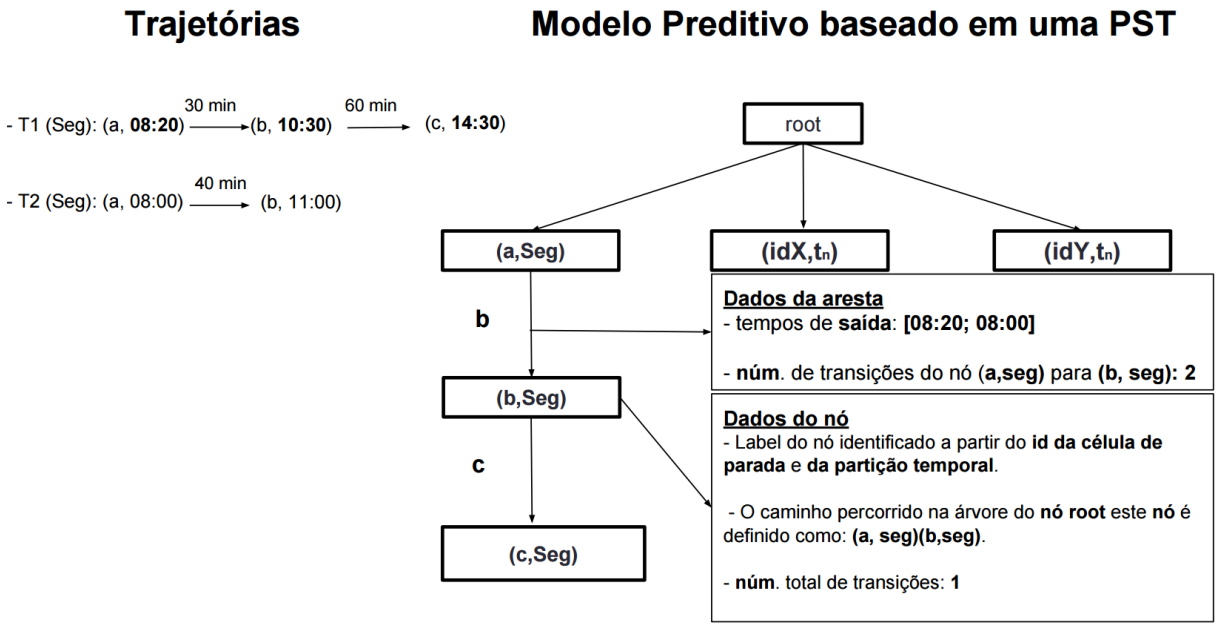
A semelhança de uma PST clássica onde cada nó em $\mathscr{T}$ possui um vetor de distribuição de probabilidade sobre o alfabeto $\Sigma$, como foi definido na Seção 2.4. Os nós reprentados em nosso modelo preditivo possui uma tabela preditiva com as informações espaciais e temporais necessárias para o cálculo da probabilidade das transições entre as células de parada. Como podemos ver na Figura 9 cada nó possui, além do número total de transições que ocorreram a partir de si, uma tabela preditiva contendo, a próxima célula de parada, o número de transições do nó para a respectiva célula e os tempos de inicío da transição, ou seja, os tempo de saída do nó para a célula de parada seguinte.

Na Figura 9 observamos um modelo preditivo com as informações identificadas a partir das trajetórias mencionadas na mesma Figura. Como podemos ver as arestas e os nós foram rotulados, e as tabelas preditivas foram construídas contendo os dados necessários para calcularmos a probabilidade de uma transição.

#### *4.3.2.5 Predição na PST*

Uma vez que introduzimos o conceito de *consulta*, na Definição 4.2.1, necessária para prever a próxima célula de parada e quando o objeto em movimento deixará sua localização

Figura 9 – Representação simples de um modelo preditivo construído a partir das trajetórias mencionadas.




Fonte: elaborada pelo autor.

atual, é fundamental determinar, para qualquer par de células de parada, a probabilidade de transição entre as células. Esta probabilidade, por sua vez, depende das informações espaciais e temporais, observadas nos dados utilizados para construir o modelo preditivo, e do tempo de consulta, $t_{now}$. A probabilidade de transição entre duas células de parada, com a finalidade de identificar a próxima localização relevante, é definida da seguinte forma:

**Definição 4.3.4 (Probabilidade da transição entre duas células de parada)** *Vamos considerar a árvore de sufixo probabilística $\mathscr{T} = (V,E)$, onde $V$ e $E$ são, respectivamente, o conjunto de nós e arestas associadas a $\mathscr{T}$. Considere também um nó interno $n \in V$, o conjunto, $C$, que representa as células de parada cujas transições ocorrem partindo de $n$ e o tempo de consulta, $t_{now}$. A probabilidade associada a uma transição, partindo de $n$ para qualquer uma das células de parada $c \in C$ é:*

$$Pro(n,c,t_{now}) = Pro_{espacial} \times Pro_{temporal}, \tag{4.5}$$

*onde $Pro_{espacial}$ representa o componente espacial da probabilidade, e $Pro_{temporal}$ representa o componente temporal. Ambos os componentes devem assegurar que $\sum_{c \in C} Pro(n,c,t_{now}) = 1$.*

Em seguida definimos a probabilidade espacial e temporal associada a uma transição:

Probabilidade espacial

Ao considerar a informação no domínio espacial, a principal questão é como gerenciar as observações de $T'_u$ das transições físicas que ocorrem entre a célula de parada associada ao nó e as próximas células de parada descritas na tabela preditiva, e traduzir estas observações em uma medida significativa que expresse a probabilidade de um transição. Para este fim, apresentamos a definição de *probabilidade espacial de uma transição*.

**Definição 4.3.5 (Probabilidade espacial de uma transição)** *Considere um nó n e o conjunto, C, de células de parada que representam as possíveis transições partindo de n. Considere também que a quantidade de transições de n para cada célula de parada $c \in C_n$, denotada por $tr_c^n$. Definimos a probabilidade espacial de uma transição a partir de n para uma célula $c \in C_n$ como:*

$$Pro_{espacial}(n,c) = \frac{tr_c^n}{\sum_{z \in C_n} tr_z^n} = \frac{\textit{núm. de transições para c a partir de n}}{\textit{núm. de transições total do nó n}} \tag{4.6}$$

Em outras palavras, a probabilidade espacial de uma transição partindo de $n$ para uma célula de parada $c \in C_n$ é maior quando a quantidade de transições entre eles torna-se dominante, no que diz respeito ao número total de transições de $n$ em relação às celulas. Finalmente, notamos que a equação 4.6 garante que $\sum_{c \in C_n} Pro_{espacial}(n,c) = 1$. Adotamos o uso da probabilidade simples ao invés da probabilidade condicional pelo benefício do custo operacional e também porque o nó mais similar é identificado, conforme descrito na Seção 4.4.1, beneficiando as predições de localização.

Probabilidade temporal

A seguir, descrevemos como consideramos e utilizamos as informações no *domínio temporal*. Quando avaliamos a probabilidade de uma transição considerando o componente temporal é primordial que, a probabilidade seja influenciada (i) pelo tempo de consulta, $t_{now}$, informado na consulta da predição e (ii) pelos hábitos descritos pelos movimentos passados do objeto em movimento.

Como exemplo ilustrativo, considere uma pessoa que, durante os dias úteis, sai de *casa* para o *trabalho* durante o início da manhã e volta para *casa* no início da noite. Porém, às terças e quintas-feiras, essa pessoa vai do *trabalho* para *academia* no final da tarde e finalmente

vai para *casa*. Duas observações podem ser feitas: primeiro, uma pessoa pode ficar várias vezes dentro da mesma célula de parada (por exemplo, *em casa*) na mesma partição temporal (um dia). Como consequência, para prever a próxima célula de parada torna-se fundamental considerar o tempo em que a predição será realizado, $t_{now}$, e ao mesmo tempo compreender, a partir do histórico dos dados, se os tempos de saídas relacionados às transições que partem do *trabalho* para outra célula de parada está concentrada em volta de momentos específicos: que denominamos de *tempos representativos de saída*. Segundo, uma pessoa pode planejar suas atividades de acordo com programações específicas, neste exemplo, a pessoa vai para a academia apenas às terça e quintas-feiras, e como tal, o modelo preditivo é capaz de inferir as relações entre os movimentos padrões e os intervalos específicos de tempo, desde que as partições temporais sejam utilizadas, enquanto que o algoritmo preditivo aproveita estas informações para realizar previsões mais assertivas. A seguir, ilustraremos o conceito de *tempos representativos de saída*, como podemos integrar essa informação na PST e como podemos usá-la para calcularmos a probabilidade do componente temporal.

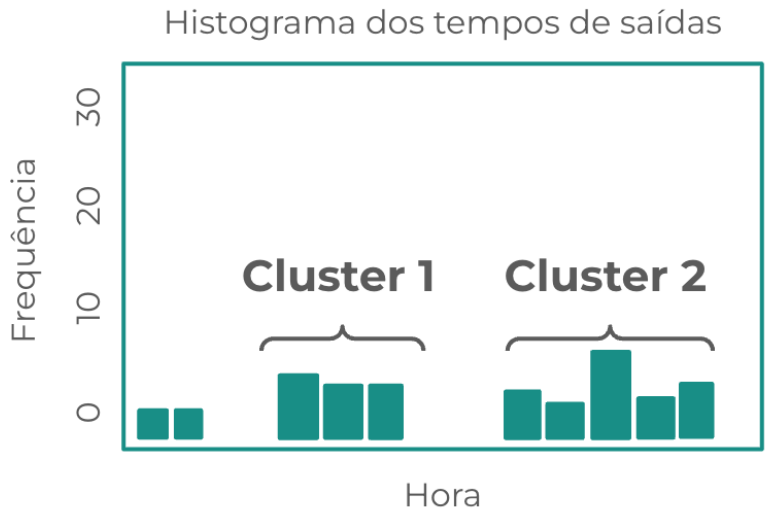
Consideremos que cada transição $((id_X, W), (id_Y, Z))$ está associada a um conjunto de tempos de saídas observados em cada transição indo de $(id_X, W)$ para $(id_Y, Z)$. Os tempos de saída representam o início do movimento de um objeto que se desloca da célula de parada $id_X$ para a célula de parada $id_Y$. Assumamos também que os tempos de saída geralmente dependem dos hábitos dos objetos em movimento, logo, tendem a formar diferentes grupos temporais, *clusters temporais*, cada um seguindo uma distribuição de frequência para dados agrupados por intervalos. Nós definimos os *tempos representativos de saída* durante uma transição de $(id_X, W)$ para $(id_Y, Z)$ como os tempos mais prováveis de ocorrerem dentro de uma partição temporal $W$. O problema, então, se traduz em descobrir os clusters temporais e determinar, para cada um deles, suas propriedades estatísticas. Assim, uma vez identificado os cluster temporais, a média dos tempos de saída de cada cluster temporal representa um tempo representativo de transição da célula $id_X$ para $id_Y$. O objetivo dos tempos representativos é determinar qual cluster temporal possui a média mais próxima do tempo de consulta da predição, $t_{now}$. Vejamos um exemplo para consolidarmos o conceito de *tempos representativos de saída*.

Suponhamos que para um dado objeto em movimento $u$, observamos em $T'_u$ várias transições da célula de parada $id_1$ (local do trabalho) para célula de parada $id_2$ (casa) que ocorre durante os dias úteis da semana. Vamos supor que para tais transições observamos o seguinte conjunto de tempos de saída: {17:56, 11:57, 12:00, 18:03, 12:03, 11:55, 17:57, 18:02, 18:23}.

A partir desses dados, podemos facilmente observar que tais transições formam dois clusters temporais distintos, um cluster com os tempos de saída {11:57, 12:00, 12:03, 11:55} e com tempo representativo por volta das *12:00*, e outro cluster com tempos de saída {17:56, 18:03, 17:57, 18:02, 18:23} e com o tempo representativo por volta das *18:00*.

Precisamos de uma estratégia para descobrir os clusters temporais para associar conjuntos de tempos de saídas a cada transição na PST e determinar um tempo representativo de saída para cada um desses conjuntos. Para este fim, propomos um algoritmo baseado em dois componentes propostos na abordagem de (NG *et al.*, 2005); mais precisamente, exploramos a fase de *construção do histograma*, que no nosso caso consiste em associar os tempos de saída das transições às frequências em que estes ocorrerem. E a fase de *identificação da região densa* que tem por objetivo identificar os clusters temporais, com base na frequência dos dados e a dispersão entre estes, para que as informações estatísticas destes sejam utilizadas na probabilidade temporal, Definição 4.3.6. Na Figura 10 ilustramos os clusters temporais identificados. O esboço do algoritmo é descrito em seguida e o pseudo código é demostrado no algoritmo 1.

Figura 10 – Ilustração de clusters identificados a partir do histograma criado construído sobre os tempos de saídas




Fonte: elaborada pelo autor.

O objetivo do algoritmo é: para cada nó $n \in V$, vamos considerar os tempos de saída associados às transições, registrados na tabela preditiva $T$, e em seguida construir um histograma a partir destes tempos. Para isso criaremos um histograma em intervalos disjuntos e de tamanhos iguais, chamados de *bins*, por exemplo, se dividirmos um dia, 24 horas, em intervalos de 30 minutos teremos 48 *bins*, e consequentemente teremos classes que obedecerão

**Algoritmo 1:** Identificação do cluster temporal

**Input** : PST $\mathscr{T} = (V, E)$
**Output** : PST $\mathscr{T}$, com a informação dos clusters temporais adicionadas.

```
1  begin
2      foreach n ∈ V do
3          T = retorneTabelaPreditiva(n)
4          foreach t = (n, (id_{k+1}, Z)) ∈ T do
5              Construir um histograma H a partir dos tempos de saídas da tabela preditiva associada ao nó, n, célula de
                parada (id_k, W). Para isto, use a fase de construção do histograma.
6              Descobrir os clusters temporais, C = {c_1 ··· c_k}, no histograma H, aplicando a fase de identificação de
                regiões densas.
7              foreach c ∈ C do
8                  Calcule a média, variância e o desvio padrão de c.
9              end
10             Associar C à transição, t, representada na tabela preditiva, T, de n.
11         end
12     end
13 end
```

os seguintes intervalos, em minutos, $(0 \vdash 30, 31 \vdash 60, ..., 1381 \vdash 1410, 1411 \vdash 1440)$. Assim cada tempo de saída é associado a uma classe criando um histograma (linha 5). Notamos que essa parte corresponde à fase de *construção do histograma* apresentada em (NG *et al.*, 2005). Posteriormente, detectamos as sequências de classes consecutivas onde os tempos de saída tendem a se concentrar. Cada conjunto de classes consecutivas representa um cluster temporal (linha 6), isto é, uma distribuição de frequência para dados agrupados por intervalos, dos tempos de saídas distribuídos em volta de um tempo representativo de saída. Esta fase corresponde a fase *detecção da região densa* apresentada em (NG *et al.*, 2005). Finalmente, para cada cluster temporal calculamos a média dos tempos de saídas do cluster, que representa o tempo representativo de saída, e o desvio padrão (linha 8), e associamos estas informações a transição (linha 10). O desvio padrão, $\sigma$, e a variância, $\sigma^2$, são medidas de dispersão que descrevem como os dados estão distribuídos em torno da média e descrevem o quanto os dados de uma variável aleatória são semelhantes. O desvio padrão determina o quão oscilável são os valores de um conjunto de dados em relação a média da distribuição que representa os dados; se o desvio padrão é alto indica que os dados oscilam muito em relação a média, se o desvio padrão é baixo, significa que os dados oscilam pouco, e se o desvio padrão é zero significa que os dados do conjunto são todos iguais, pois a dispersão é nula. O desvio padrão é utilizado para comparar cada amostra com o valor central, média, enquanto que a variância analisa todas as amostras. Na prática o desvio padrão representa o quão disperso está cada amostra do dado em relação a média, e a variância indica o qual disperso está todo o conjunto de amostras em relação a média. Como podemos observar o desvio padrão é a raiz quadrada da variância, $\sigma = \sqrt{\sigma^2}$.

Uma vez que cada transição tem as informações estatísticas associadas aos seus clusters temporais, podemos definir a probabilidade do componente temporal, $Pro_{temporal}$.

**Definição 4.3.6 (Probabilidade temporal de uma transição)** *Dado o tempo atual de consulta, $t_{now}$, um nó $n = (id, X)$, o conjunto de transições, $T_n$, associadas a n e o conjunto de clusters temporais associados a cada transição partindo de n para seus possíveis destinos, onde $C_{(n,ts)}$ representa o conjunto de clusters temporais associados à transição $(n,t)$, $t \in T_n$. Também, $\mu_c$ e $\sigma_c^2$ representam, respectivamente, a média e a variância dos tempos de saídas associados ao cluster temporal c. Então, a probabilidade temporal associada a uma transição $(n,t)$, $t \in T_n$, é dada por:*

$$Pro_{temporal}(n,t,t_{now}) = \frac{CI_{temp}(n,t,t_{now})}{\sum_{t \in T_n} CI_{temp}(n,t,t_{now})} \tag{4.7}$$

$$CI_{temp}(n,t,t_{now}) = \frac{\sigma_c^2}{|\mu_c - t_{now}|^2} \tag{4.8}$$

*onde o cluster temporal c é selecionado de acordo com $\min_{c \in C_{(n,t)}} |\mu_c - t_{now}|^2$.*

*A equação 4.8 é baseada na inequação de Chebysev (TCHEBICHEF, 1874) e poderia ser usada para representar a probabilidade temporal, mas a condição $\sum_{t \in T_n} CI_{temp}(n,t,t_{now}) = 1$ não seria satisfeita; para atender a condição acima, é necessário normalizar os valores obtidos em todas as transições entre n e $t \in T_n$ para representar as probabilidades temporais obtidos nas transições, conforme observamos na equação 4.7.*

A ideia por trás da escolha do cluster temporal $c \in C_{(n,r)}$, onde o denominador na equação 4.8 é minimizado é a seguinte: uma vez que uma transição pode ser associada a diferentes clusters temporais, e a função que determina a probabilidade temporal depende do tempo de consulta $t_{now}$, faz sentido selecionar o cluster temporal que tem o representante mais próximo de $t_{now}$. Voltando ao Exemplo 4.3.2.5, e supondo que $t_{now} = 11{:}30$, o primeiro cluster a ser selecionado seria o que possui tempo médio de saída igual a 12:00, uma vez que é o mais próximo de 11:30.

A sequir mostramos como a árvore de sufixo probabilística proposta, modelo preditivo, é construída com base nas definições apresentadas até o momento.

### *4.3.3 Construção de uma árvore de sufixo probabilística*

O algoritmo 2 é responsável pela construção de uma árvore de sufixo probabilística, a partir de uma trajetória transformada de um usuário $u$, $T'_u$. Dado que $Q$ representa uma sequência genérica de amostras, denotaremos, $Q(x)$ e $Q(ltima)$ para indicar a $x$-ésima amostra e *última* amostra, respectivamente, dentro de $Q$.

**Algoritmo 2:** Construção da árvore de sufixo probabilística

**Input :**
- A trajetória transformada referente aos movimentos de um objeto em movimento $u$, $T'_u$.
- Um ciclo temporal, $Time^+$.
- Um valor mínimo, limite temporal, de permanência, $\sigma$, e um limite temporal inter-amostra, $\delta$.
- Um limite temporal de transição inter-celular, $\tau$.

**Output :** Uma árvore de sufixo probabilística $\mathscr{T}$.

```
1  begin
2      V ← {root}, E ← ∅
3      if T'_u ≠ ∅ then
4          celulas ← converterAmostrasParaCelulas(T'_u)
           celulasDeParada ← identificarCelulasDeParada(celulas, σ, δ) if celulasDeParada ≠ ∅ then
5              < P_1, P_2, ··· , P_m > ← descobrirParticoesTemporais(celulasDeParada, Time^+)
6              for P ∈ < P_1, ··· , P_n > do
7                  < C_1, C_2, ··· , C_k > ← minerarPadroesDeMobilidade(celulasDeParada, P, τ) for C ∈ < C_1, ··· , C_j >
                   do
8                      noAnterior ← root
9                      movPadroes ← ⊥
10                     tempoDeSaida ← ⊥
11                     while C ≠ ∅ do
12                         cpAtual ← proximaCelulaDeParada(C)
13                         atualizarTabelaPreditivaNo(noAnterior, (cpAtual, P), tempoDeSaida)
14                         movPadroes ← movPadroes ∪ {cpAtual}
15                         noAtual ← pesquisarNoAtual((cpAtual, P), noAnterior, V)
16                         if noAtual = ⊥ then
17                             noAtual ← criarNo((cpAtual, P), noAnterior)
18                             V ← V ∪ {noAtual}
19                             criarAresta(E, (noAnterior, noAtual))
20                             criarTabelaPreditivaNo(noAtual)
21                         end
22                         tempoDeSaida ← recuperarTempo(amostras(cpAtual)(ultima))
23                         noAnterior ← noAtual
24                         C ← C \ movimentosPadroes
25                     end
26                 end
27             end
28         end
29     end
30     return 𝒯 ← (V, E)
31 end
```

A ideia básica é descobrir em $T'_u$ as células de parada e as transições entre elas, que por sua vez, terão as informações temporais e espaciais armazenadas na árvore, construindo assim, o modelo preditivo. O algoritmo funciona da seguinte forma: em primeiro lugar, ele inicializa os conjuntos de nós e arestas, respectivamente $V$ e $E$ (linha 2). Em seguida, inicia

a operação que converte as amostras de $T'_u$ (linha 4) em células, conforme a Definição 4.1.2. Após isso ele tenta descobrir uma subsequência de células que representam a permanência em uma célula de parada (função *identificarCelulasDeParada*, linha 4), de acordo com a Definição 4.1.5. Então o algoritmo prossegue para verificar se células de parada foram encontrada ou não (linha 4): no primeiro caso, o algoritmo analisa *celulasDeParada* a fim de atualizar a PST, caso contrário ele termina imediatamente retornando a PST $\mathscr{T}$. De fato, se $celulasDeParada = \emptyset$ significa que não descobrirmos permanências, células de parada, em $T'_u$.

A análise das células de parada identificadas prossegue da seguinte maneira: primeiro, o algoritmo determina as partições temporais, estendidas pelas *celulasDeParada*, em $m$ partições disjuntas, (função *descobrirParticoesTemporais*) linha 5. Em seguida o algoritmo prossegue para descobrir os padrões de mobilidade em dada partição temporal, $P$, como podemos ver na linha 7 (função *minerarPadroesDeMobilidade*) de acordo com a Definição 4.1.7. Uma vez que os caminhos realizados foram definidos, iniciamos a construção da árvore de sufixo probabilística e, para isso, analisamos as transições entre as células de parada em cada padrão, $C$, considerando a partição $P$. Para determinar os nós e as arestas inicializamos o *noAnterior*, *movPadroes* e *tempoDeSaida* e para cada $C \in < C_1, \cdots, C_j >$ essas variavéis são reinicializadas, visto que a análise sempre começará a partir do nó *root*, linhas 7 a 10 . Na linha 12 é pesquisado a próxima célula de parada no $C$ corrente e na linha 13 é atualizado a tabela preditiva do nó anterior com as informações da transição entre o nó *noAnterior* e a célula atual, *cpAtual*. Notamos que, nesta fase, também incorporarmos (ou atualizamos) em cada transição todas as informações necessárias para determinarmos a probabilidade espacial e os clusters temporais associados, ou seja, a quantidade de transições e os tempos de saída (variável *tempoDeSaida*) observados até o momento. Posteriormente a *cpAtual* é adicionada aos movimentos percorridos (linha 14), e posteriormente é identificado o nó atual, linha 15. Se o nó atual não existir em $V$ ele será construído e adicionado a $V$ e, a transição, que parte do *noAnterior* para o *noAtual*, será mapeada em $E$, linhas 16 a 21.

Posteriormente a variável *tempoDeSaida* é atualizada considerado as informações da última amostra de *cpAtual*, que representa o tempo de saída de *cpAtual* para a próxima célula de parada existente em $C$, linha 22, em seguida o *noAnterior* é atualizado com o valor do *noAtual*, linha 23, e o algoritmo prossegue removendo de $C$ os movimentos já percorridos até o momento, linha 24, para que o critério de parada, definido na linha 11, seja satisfeito. O algoritmo finaliza quando todos os padrões de mobilidade para todas as partições temporais

foram analisados. E finalmente, na linha 30, $\mathscr{T}$ é construída a partir de $V$ e $E$.

## 4.4 Algoritmo preditivo

Nesta seção ilustraremos o algoritmo usado para computar as consultas definidas na definição do problema (Definição 4.2.1).

### *4.4.1 Explorando o histórico dos movimentos recente de um objeto em movimento*

A consulta fornece duas entradas, o tempo de consulta atual , $t_{now}$, e uma trajetória, $T_u^H$, que representa os movimentos mais recentes de um objeto em movimento $u$. Assim, a primeira tarefa é descobrir um nó na PST $\mathscr{T}$ de $u$ a partir do qual possamos realizar a predição.

Consequentemente, o algoritmo preditivo primeiro determina a sequência de células de parada (se houver) dentro de $T_u^H$; posteriormente, ele descobre um nó adequado $\mathscr{T}_u$ que representa o último ponto em um caminho da árvore, cujo sufixo, é o mais similar dentre todos os que caracterizam a sequência de células de parada dentro de $T_u^H$, onde as células de parada que representam os movimentos mais recentes possuem um peso maior que as anteriores (esta última operação é realizada usando a mesma abordagem usada em (LEI *et al.*, 2013)). Por exemplo, supondo que na consulta as células de parada identificadas na trajetória foram **abc**, e no modelo preditivo existem os nós {**a, b, c, bc e ab** }, aplicando a função que calcula a similaridade do movimento, Equação 4.9, temos que o valor do movimento similar para cada nó candidato, respectivamente, é { **0,07; 0,28: 0,64; 0,93, e 0,36** }. Logo, o nó cujo percurso é **bc** será o nó mais similar considerando a trajetória **abc** identificada em $T_u^H$.

$$MS(n_k, s_q) = \sum_{i=x}^{n} \frac{i^2}{\sum_{j=1}^{q} j^2} \tag{4.9}$$

onde $q = tamanho(s_q)$, $x = pesoFirstIndex(MSC(n_k, s_q), s_q))$, $MSC(n_k,s_q)$ é o máximo sufixo comum entre o nó candidato, $n_k$, e a trajetória da consulta, $s_q$, $n = pesoUltIndex(s_q, MSC(n_k, s_q))$ e 0 $\leq$ MS($n_k$,$s_q$) $\leq$ 1. Por exemplo, $MS(b, abc) = \sum_{i=2}^{2} \frac{i^2}{\sum_{j=1}^{3} j^2}$ e $MS(bc, abc) = \sum_{i=2}^{3} \frac{i^2}{\sum_{j=1}^{3} j^2}$.

Nas consultas em que o nó mais similar não é identificado, casos que ocorrem quando os padrões identificados nos movimentos recentes do objeto não foram mapeados no modelo

preditivo, a predição é realizada a partir do nó *root*, desta forma garantimos que todas as consultas de predições sejam realizadas.

### *4.4.2 Prevendo a próxima célula de parada*

Uma vez que o nó mais apropriado, $n \in \mathscr{T}_u$, é encontrado, o próximo passo é determinar dentre, as próximas células de parada mapeadas na tabela preditiva de $n$, qual célula possui a maior probabilidade de transição em relação ao $t_{now}$. Se $C$, denota o conjunto das células de parada cujas transições ocorrem partindo de $n$, de acordo com as equações 4.5, 4.6 e 4.7 dizemos que a *próxima* célula de parada de um objeto em movimento é dada pela máxima probabilidade da transição encontrada:

$$\max_{c \in C_n} Pro(n, c, t_{now}) \tag{4.10}$$

### *4.4.3 Prevendo o tempo de saída*

O próximo problema é prever *quando* um objeto em movimento vai deixar seu local atual, conhecido em $T_u^H$. Dois cenários são possíveis: o primeiro é quando $u$ está localizado em uma célula de parada mapeada na PST $\mathscr{T}$, e o segundo é quando $u$ está localizado numa célula normal. No primeiro caso, temos de explorar a informação dos clusters temporais associados com a transição $(n,t)$, maximizando a equação 4.7. No segundo caso assumimos que $u$ está em transição para uma célula de parada, daí o tempo de saída predito será igual ao $t_{now}$. Lembrando que a probabilidade temporal associada a uma transição é determinada considerando dentre seus clusters temporais, o que tem o tempo representativo de saída mais próximo de $t_{now}$, em vista disto, podemos prever que os tempos de saída de $u$ passam a ser $\mu_c$, onde o cluster $c$ é selecionado de acordo com a solução de $\min_{c \in C_{(n,r)}} |\mu_c - t_{now}|^2$.

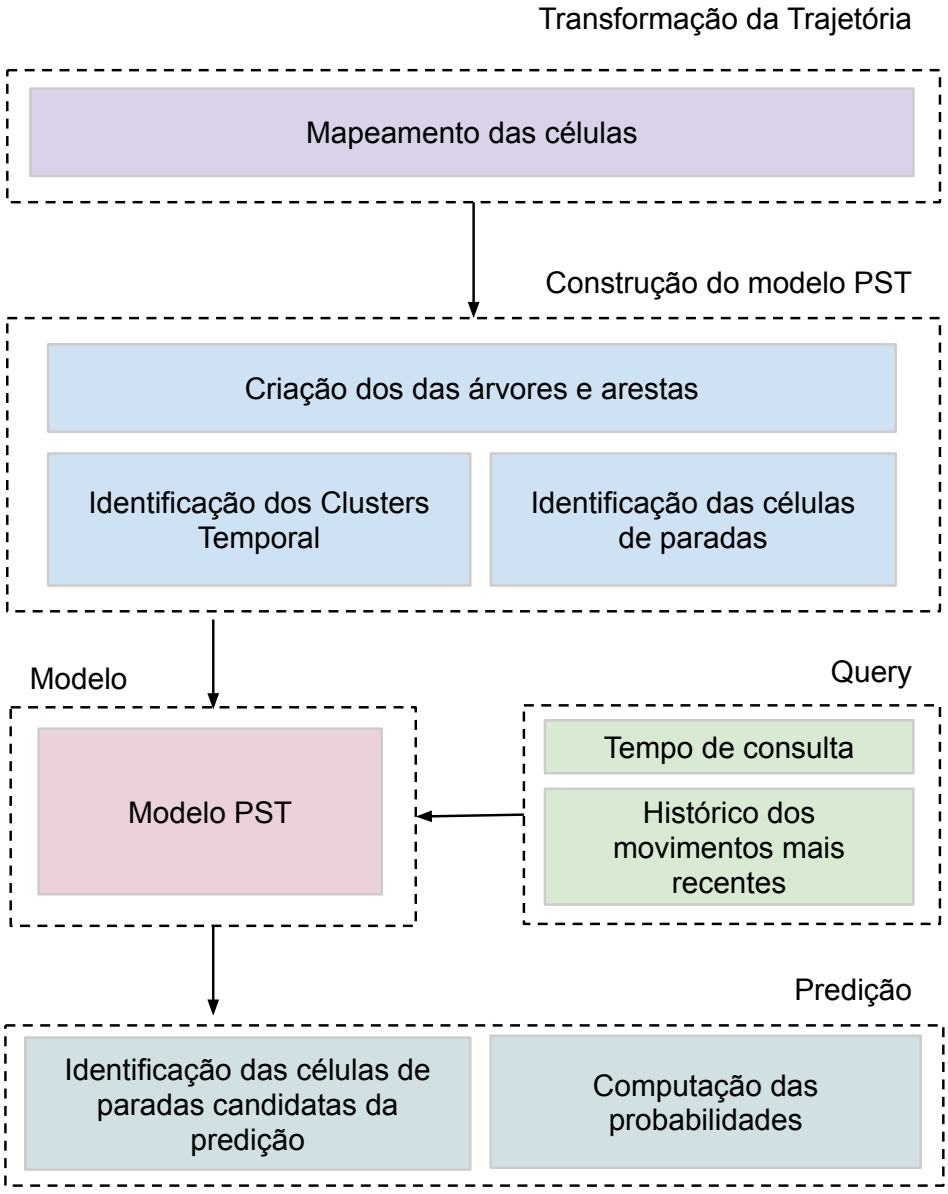
## 4.5 Framework

Nesta seção, vamos descrever a estrutura usada para resolver o problema de predição considerando o modelo preditivo e o algoritmo de predição descritos nas Seções 4.3 e 4.4. Para isto, apresentamos os principais componentes do framework, bem como o *workflow* através do qual realizamos todo o processamento.

De agora em diante vamos supor que o fluxo de trabalho opera em nível individual, ou seja, para um único usuário, de acordo com as especificidades do problema de predição.

A figura 11 apresenta uma visão geral do workflow. Como podemos ver, os componentes responsáveis pelo processamento são divididos em três fases principais, denominadas, (i) a fase de *transformação trajetória*, (2) a fase de *construção da PST* e, finalmente, (iii) a fase de *predição*.

Figura 11 – Visão geral da estrutura do framework T-PRED. Destacando as principais tarefas para realizar uma predição




Fonte: elaborada pelo autor.

### *4.5.1 Primeira Fase – Transformação da trajetória*

O objetivo desta fase é transformar uma trajetória original, $T_u$, associada a um usuário $u$, em uma trajetória transformada, $T'_u$, cujas amostras são associadas individualmente a uma célula específica de alguma rede de particionamento no espaço. As propriedades da grid depende da função de mapeamento $f$ (como especificado na Definição 4.1.2), enquanto que as células são enumeradas de acordo com a função $g$, especificada na Definição 4.1.3. Para simplificar vamos assumir uma função única $tr : \mathbb{R}^2 \rightarrow \mathbb{N}_0$ que representa a composição de $f$ e $g$, ou seja, $tr(\cdot) = g(f(\cdot))$. No nosso trabalho escolhemos $tr$ para implementar a *Universal Transverse Mercator* (UTM), sistema de coordenadas, proposto por (SNYDER, 1987); este sistema é flexível, na medida em que permite gerar grides uniformes tendo células de tamanhos arbitrários, sendo esta uma propriedade útil sempre que o sistema deve ser ajustado de acordo com as características específicas dos objetos em movimento considerados.

Assim aplicar $tr$ equivale a sobrepor um grid uniforme $\mathscr{G}$ sobre o espaço no qual o usuário está em movimento, logo o espaço é discretizado em um conjunto de células disjuntas $\mathscr{C} \in G$, e a sequência de amostras da trajetória original, $T_u =< s_1, \cdots, s_n >$, $s = (l,t)$, é transformada em uma sequência *transformada* de amostras $T'_u =< s'_1, \cdots, s'_n >$, onde $s'_i = (tr(l_i), t_i)$.

### *4.5.2 Segunda Fase – Construção da Árvore de Sufixo Probabilística*

O objetivo desta fase é gerar um modelo preditivo a partir da trajetória transformada $T'_u$, um ciclo temporal $Time^+$ e um conjunto de limites temporais $\{\sigma, \delta, \tau\}$. Em outras palavras, a partir das informações contidas em $T'_u$, esta fase tem por objetivo gerar uma árvore de sufixo probabilística, denominada PST, a partir do comportamento dos movimentos padrões do usuário, e juntamente com todas as informações necessárias, tanto no domínio espacial quanto temporal, avaliar as probabilidades de transição na fase de predição.

A atividade que gera o modelo preditivo, uma PST, leva em consideração o modelo da PST introduzido na seção 4.3.2, bem como o algoritmo de construção ilustrado no Algoritmo 2. A implementação do algoritmo 2 requer a utilização dos três componentes principais para realizar suas operações.

#### *4.5.2.1 Identificação das Células de Parada*

Uma das principais operações dentro do algoritmo 2 é determinar as células de parada e as transições existentes entre estas em $T'_u$. As células de parada representam as células onde o usuário passa uma quantidade relevante de tempo (de acordo com a Definição 4.1.5). Como consequência, o algoritmo 2 usa o componente responsável por realizar esta operação.

#### *4.5.2.2 Identificação do Cluster Temporal*

Uma vez que o algoritmo 2 gera uma PST, uma operação fundamental a ser incorporada dentro das transições da PST, armazenadas na tabela preditiva de cada nó $n$, são as informações necessárias para calcular as probabilidades de transição no domínio temporal utilizadas numa fase de predição posterior. Mais precisamente, na fase da construção da PST cada transição $((id_y, X)(id_j, Y))$ contém as informações relacionadas aos tempos de saída observados a partir de $id_i$, na zona temporal $X$, para $id_j$ para partição temporal $Y$. Porém estas informações, não são suficientes para calcular $Pro_{temporal}$. Por isso, para este fim, o algoritmo 1 é usado para identificar o *representante* dos tempos de saída associados a cada transição, isto é, as médias dos clusters temporais que podem ser inferidas a partir dos tempos de saída observados quando $T'_u$ é processada.

### *4.5.3 Terceira fase – Predição*

Dado o histórico dos movimentos recentes do usuário $u$, $T_u^H$ e um tempo atual de consulta $t_{now}$; a fase de predição computará a query $q(T_u^H, t_{now})$, conforme especificado na definição do problema em 4.2.1, tirando proveito do modelo preditivo representado pela árvore de sufixo probabilística associada a $u$, $\mathscr{T}_u$.

A fim de realizar uma predição todos os passos descritos e detalhados em 4.4 são executados e a próxima célula de parada, bem como o tempo de saída predito do local atual do usuário são retornados.

# 5 AVALIAÇÃO

## 5.1 Cenário experimental

Nesta seção, descrevemos os experimentos realizados para avaliar a eficácia e o desempenho do TPRED, o *framework* proposto em nossa abordagem que implementa o modelo preditivo e o algoritmo preditivo descrito nas seções 4.3 e 4.4. Lembramos que o *framework* TPRED atua sobre os movimentos de apenas um único objeto conforme as definições do modelo e algoritmo preditivos. Todos os experimentos foram conduzidos num sistema usando um processador i7-3632QM, 8 GB de memória RAM e o sistema operacional Ubuntu 14.04.

Tabela 1 – Notas dos participantes nas avaliações A, B e C

| Dataset | Usuários | Intervalo de Tempo | Trajetórias | Distância (km) | Duração (horas) |
|---|---|---|---|---|---|
| **Geolife** | 182 | 3 anos | 19.000+ | 1.2 milhões | 48.000+ |
| **Eai** | 660 | 5 meses | 10.000+ | 3.7 milhões | 170.000+ |

Fonte: elaborada pelo autor.

### *5.1.1 Conjunto de dados*

O *framework* foi avaliado usando dois conjuntos de dados distintos, cujas características são indicadas na Tabela 1. O primeiro conjunto de dados considerado é o **Geolife**[1], um conjunto de dados disponível publicamente e usado em diferentes tópicos de pesquisa sobre serviços baseados em localização (ZHENG *et al.*, 2010). Esse conjunto de dados contém uma ampla variedade de usuários em movimentos ao ar livre, como rotinas da vida diária e atividades recreativas, locomovendo-se da mais diversas formas. O segundo conjunto de dados foi obtido a partir dos movimentos dos usuários coletados pelo aplicativo **Eai**, um aplicativo móvel criado para apoiar as atividades dos alunos nos campus universitários[2]. Os dados coletados abrangem, além dos campus, a cidade de Fortaleza e algumas cidades do interior do estado Ceará.

### *5.1.2 Criação dos conjuntos de treinamento e de testes*

Nos experimentos usamos uma abordagem de avaliação de treinamento e teste, método *hold-out*, no qual o conjunto de dados de treinamento é utilizado para gerar o modelo preditivo, enquanto que o conjunto de testes é utilizado para avaliar a qualidade das predições; a

[1] http://goo.gl/PT7Th

[2] https://play.google.com/store/apps/details?id=br.ufc.appeai&hl=pt_BR

validação cruzada não foi adotada devido a característica temporal da solução. O conjunto de treinamento criado considera todas as trajetórias originais do conjunto de dados, juntamente com cada trajetória inicial modificada aleatoriamente em 20% das suas células de parada adicionadas de um valor aleatório - compreendido entre 30 e 60 minutos - no tempo de saída de cada nova célula de parada recém adicionada randomicamente. Essa pertubação nos dados é necessária para alterar diretamente os padrões de mobilidade e temporais existentes no movimentos dos usuários. O conjunto de teste é criado através da seleção aleatória de 20% das trajetórias originais de cada usuário. Cada trajetória selecionada representará o histórico dos movimentos recentes do usuário em movimento e o tempo de consulta, necessário para execução do algoritmo preditivo, será aleatoriamente selecionado, dentre os tempos contidos nos dados coletados na última célula de parada existente na consulta que será utilizada para requisitar a predição.

### *5.1.3 Discretização do grid e dos ciclos temporais do TPRED*

TPRED utiliza um método de discretização espacial para particionar o espaço em células disjuntas conforme mencionado na Seção 4.1. Para isto empregamos uma abordagem baseada no sistema de coordenadas UTM (*Universal Transversa de Mercator*) (SNYDER, 1987) para a discretização espacial no *framework*. No que diz respeito ao ciclo temporal e as partições temporais, definidas na Definição 4.3.1, TPRED usa o ciclo temporal de uma semana com base em partições temporais que abrange cada dia individualmente, ou seja, $Time^{+} = \{Seg, Ter, Qua, Qui, Sex, Sab, Domi\}$.

### *5.1.4 Baselines*

A seguir descrevemos os dois *baselines* utilizados nos experimentos para avaliar os benefícios decorrentes do modelo preditivo proposto e utilizado pelo TPRED: (i) *Localização mais frequente* (*MFL*) e (ii) TPRED *sem o ciclo temporal*.

**Localização Mais Frequente** (*MFL*): este *baseline* considera somente o domínio espacial e baseia-se no cálculo das probabilidades condicionais. Vamos supor aqui que $\langle s_1, \ldots, s_{i-1} \rangle$ representa a sucessão de células de parada observadas numa trajetória que contém os movimentos mais recente de um objeto em movimento $u$, $T_u^H$; vamos levar em consideração também a trajetória transformada dos movimentos passados de $u$, $T'_u$. Então, MFL calcula a probabilidade

de $u$ se mover para uma célula parada $s_i$ como:

$$P(s_i|T_u^H) = \frac{|\{t \in T_u' | t = \langle s_1, \ldots, s_{i-1}, s_i \rangle\}|}{|\{r \in T_u' | r = \langle s_1, \ldots, s_{i-1} \rangle\}|}. \tag{5.1}$$

**TPRED sem o Ciclo Temporal** (TPRED-*NTC*): este *baseline* é uma variante particular do TPRED quando o ciclo temporal não é usado; assim os nós da árvore são rotulados, somente, de acordo com as células de parada.

### *5.1.5 Métricas de desempenho*

**Acurácia Espacial**: esta medida expressa a fração de células de parada que foram preditas corretamente pelo algoritmo e é definida como:

$$Acc_{espacial} = \frac{\text{\# número de células de parada preditas corretamente}}{\text{\# número de células de parada a serem preditas}} \tag{5.2}$$

**Erro Temporal**: vamos considerar todos os casos em que uma célula parada foi corretamente predita por uma abordagem; para cada um desses casos, tempos $E_t$ e $P_t$ que representam, respectivamente, o tempo de saída esperado e o tempo de saída predito. Consequentemente, definimos o erro temporal de uma predição temporal, como:

$$Err_{temporal} = |E_t - P_t|. \tag{5.3}$$

Logo, quanto menor for a média do erro temporal sobre todas as predições realizadas corretamente, melhor é o desempenho da média da predição temporal.

## 5.2 Avaliação experimental

Nesta seção, vamos estudar a eficácia e o desempenho do TPRED quando comparado com os dois *baseline* mencionados anteriormente.

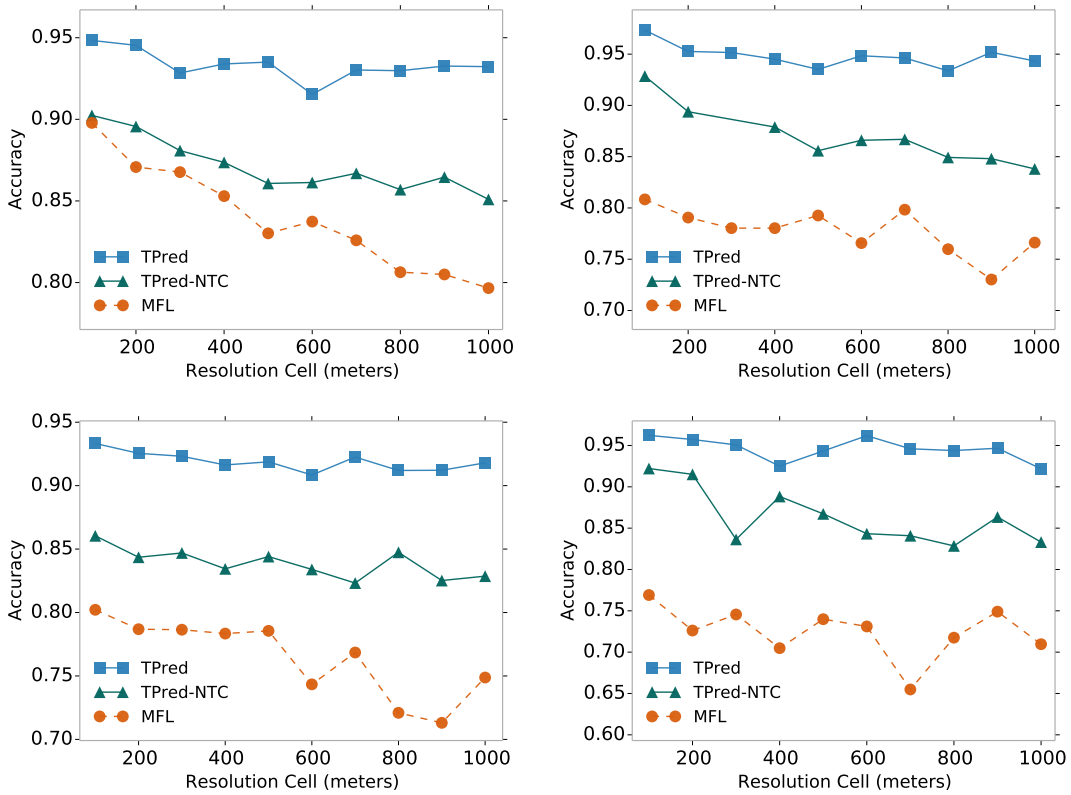
### *5.2.1 Acurácia Espacial*

Nesta seção, vamos analisar como o TPRED se comporta em termos da precisão espacial, e compará-lo com os dois *baseline* introduzidos na Seção 5.1.4. Para isto, estudamos como a acurácia espacial varia em relação aos principais parâmetros que podem influenciar o *framework*, isto é, a *resolução das células*, a *quantidade de células de parada* nas consultas e a quantidade mínima de tempo necessária para que uma célula seja considerada uma célula de

parada (isto é, a condição temporal de permanência), $\sigma$. Os outros limites temporais (condição de consistência $\delta$, Definição 4.1.5, e o limite temporal de transição inter-amostras $\tau$, Definição 4.1.7), são sempre mantido com valores fixos em 30 minutos.

No primeiro conjunto de experimentos, estudamos como a precisão espacial é influenciada ao alterarmos a *resolução das células*, considerando o intervalo $[100, 1000]$ medido em metros e utilizando o tempo de permanência $\sigma$ igual a 10 e 30 minutos. A quantidade de células de parada é variável. Os resultados são mostrados na Figura 12. A partir da figura podemos ver que a nossa abordagem sempre supera os *baselines* definidos. Além disso, destacamos a importância do uso de um ciclo temporal adequado, que nos experimentos foi definido um ciclo semanal que ajuda o TPRED a alcançar uma melhor precisão espacial em relação ao TPRED-NTC, uma vez que as partições temporais, que representam os dias individualmente, ajudam a melhorar os padrões dos movimentos dos usuários no que diz respeito ao domínio temporal.

Figura 12 – Análise da precisão espacial. Os resultados sobre o conjunto de dados do aplicativo Eai são mostrados na coluna à *esquerda*, enquanto que os resultados sobre o conjunto de dados Geolife são reportados na coluna à *direita*. As imagens do *topo* se referem aos experimentos onde $\sigma = 10$ minutos, e as imagens de *baixo* mostram os experimentos quando $\sigma = 30$ minutos.

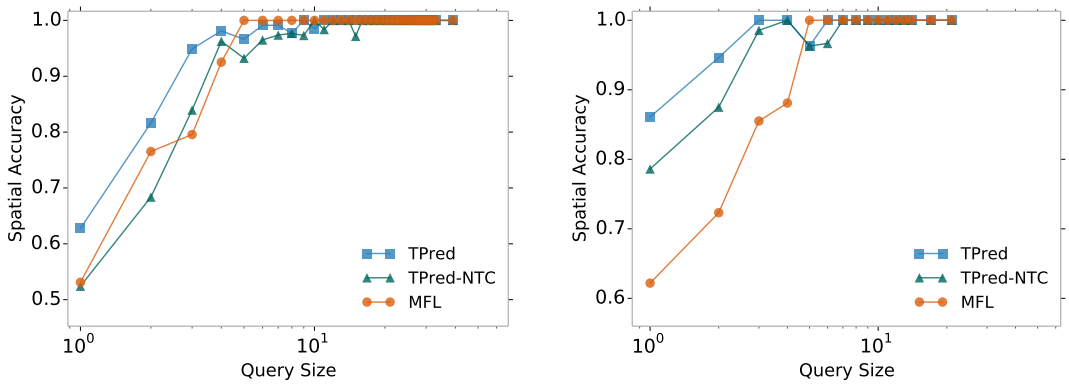



Fonte: elaborada pelo autor.

No segundo conjunto de experimentos estudamos como o *tamanho da consulta*,

quantidade de células de parada identificadas nos movimentos mais recentes do usuário em movimento, afetam a precisão espacial. Esperamos que, quanto maior o número de células de parada em uma consulta, mais precisos serão os resultados, uma vez que mais informações estão disponíveis. Neste contexto a resolução das células e $\sigma$ foram mantidos fixos, respectivamente, em 100 metros e 10 minutos. Os resultados são exibidos na Figura 13. A partir da figura podemos ver que, em geral, a precisão espacial aumenta para todas as abordagens quando o tamanho da consulta aumenta. Além disso, notamos como TPRED notavelmente supera os *baselines*, quando há poucas células de parada, menos de 4, disponíveis na trajetória recente dos usuários. Isso demonstra a importante capacidade do TPRED em predizer corretamente o próximo local relevante mesmo com poucas quantidade de células de parada na consulta.

Figura 13 – Análise do impacto do tamanho da consulta, quantidade de células de parada identificadas nos movimentos recentes dos usuários, na acurácia espacial. Os resultados dos experimentos sobre os dados do Eai são exibidos no gráfico à (*esquerda*) e os do Geolife no gráfico à (*direita*).



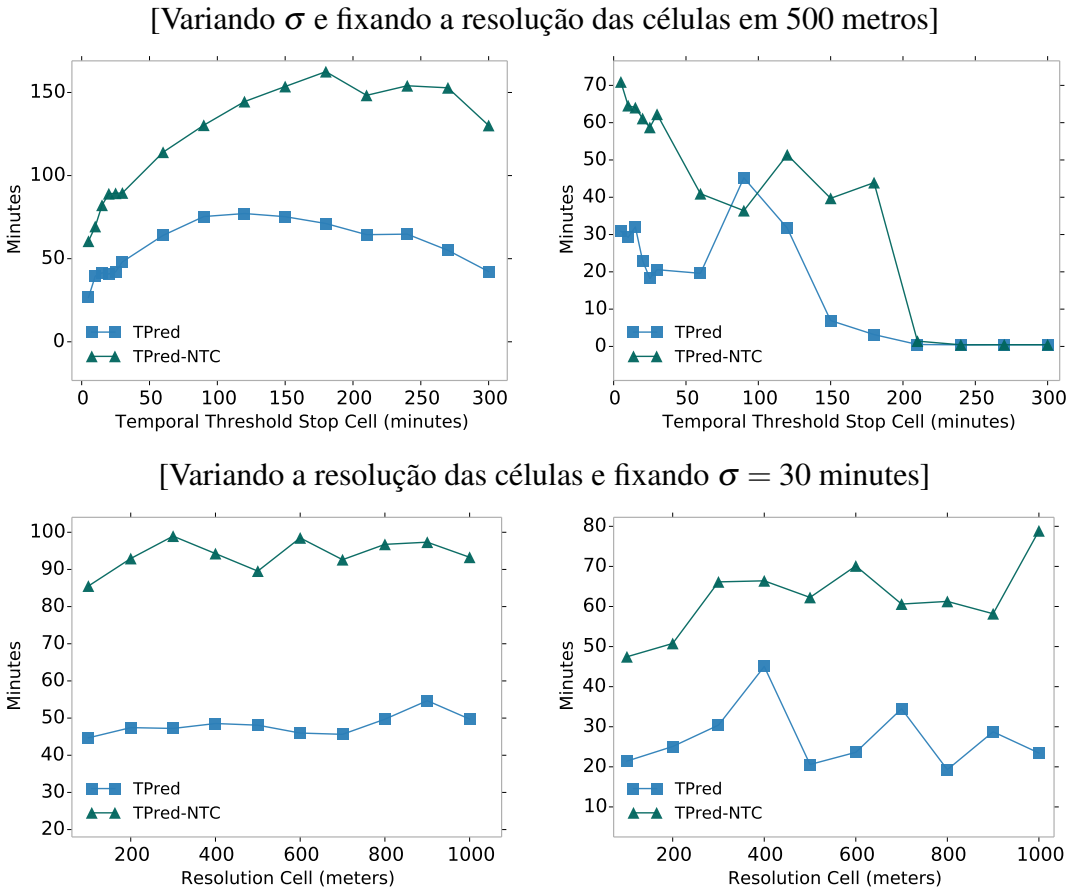

Fonte: elaborada pelo autor.

### *5.2.2 Erro Temporal*

Nesta seção, estudamos como a *média do erro temporal* varia em função dos principais parâmetros que podem influenciá-la, isto é, a *resolução da célula* e o limite temporal de *permanência* mínimo $\sigma$. Para este fim, realizamos duas baterias distintas de experimentos: na primeira variamos $\sigma$ dentro do intervalo de $[10, 300]$ de minutos, mantendo fixa a resolução das células em 500 metros. Na segunda bateria de experimentos variamos a resolução da células no intervalo de $[100, 1000]$ metros e fixamos $\sigma$ em 30 minutos. Em todos os casos o tamanho da consulta é variável, enquanto que $\delta$ foi fixado em 10 minutos e $\tau$ foi mantido em 30 minutos. Os resultados são mostrados na Figura 14.

A partir da figura vemos que, para ambos os conjuntos de dados, o TPRED quase sempre supera TPRED-NTC, com uma notável vantagem para os experimentos realizados sobre o conjunto de dados Eai.

Figura 14 – Análise da média do erro temporal $Err_{temporal}$. Resultados sobre o conjunto de dados Eai são mostrados à *esquerda*, enquanto que os do conjunto de dados do Geolife estão à *direita*.




Fonte: elaborada pelo autor.

### *5.2.3 Desempenho*

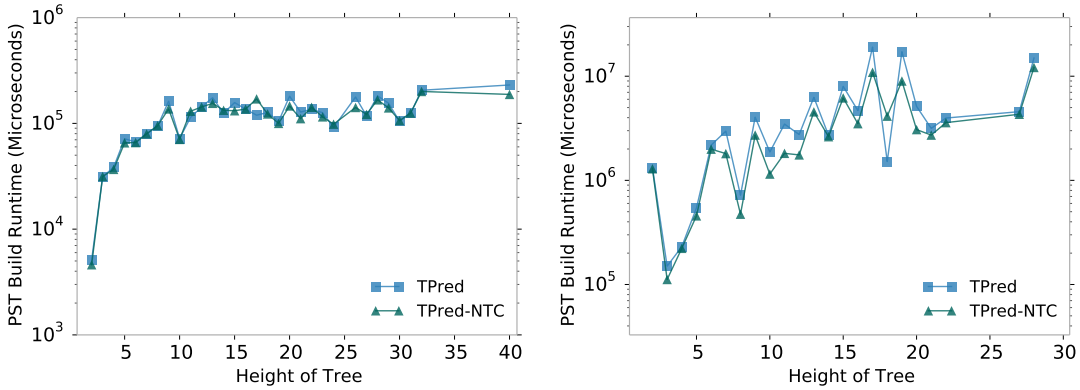
Nesta seção, vamos avaliar o desempenho de tempo de execução do TPRED durante a construção de um modelo preditivo e o processamento das predições requisitadas.

#### *5.2.3.1 Desempenho durante a construção do modelo preditivo*

No seguinte conjunto de experimentos analisamos os tempos médios de execução obtidos na construção dos modelos preditivos. O principal fator por trás do tempo necessário para construir um modelo preditivo é a quantidade de células de parada que caracterizam

os movimentos de um objeto, que por sua vez influencia a altura da árvore. Para este fim, agrupamos as árvores de acordo com a sua altura, e analisamos separadamente os tempos de execução para cada grupo. Os resultados são mostrados na Figura 15; os valores dos parâmetros relevantes são relatados na legenda da figura, enquanto o $\delta$ e $\tau$ foram fixados em 10 e 30 minutos, respectivamente. A partir da figura podemos ver que a altura máxima observada é de 40 e 28

Figura 15 – Avaliação do tempo de execução para construir os modelos preditivos usando $\sigma = 10$ minutos e 500 metros para resolução da célula. Resultados sobre os dados do Eai são mostrados à *esquerda*, enquanto que os resultados sobre os dados Geolife estão à *direita*.




Fonte: elaborada pelo autor.

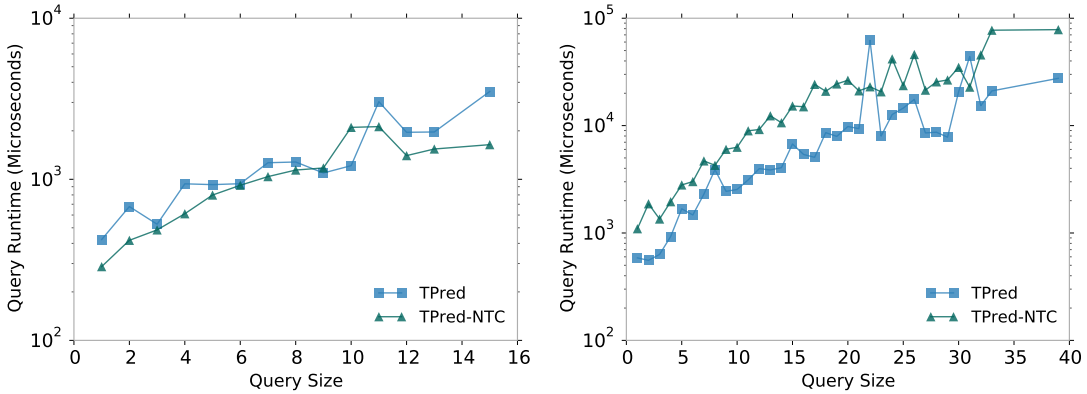
nós para Eai e Geolife, respectivamente. Além disso, vemos como TPRED (mesmo quando se leva em consideração a sua variante, TPRED-NTC) é muito rápido, uma vez que é capaz de construir modelos em menos de 1 segundo para os dados do Eai e em menos de 10 segundos para Geolife. Finalmente, observamos que aumentos na altura das árvores têm o efeito de aumentar ligeiramente os tempos médios de construção, demostrado de forma mais regular sobre o conjunto de dados do Eai.

#### *5.2.3.2 Desempenho quando computado uma predição*

Nos seguintes lotes de experimentos analisamos o desempenho do TPRED ao computar as consultas de predições requisitadas. Nos experimentos que se seguem considerarmos apenas o conjunto de dados Eai (resultados semelhantes podem ser replicados quando se considera o conjunto de dados Geolife) e variamos a quantidade de células de parada em uma consulta na faixa de $[1, 40]$; os valores dos parâmetros utilizados ao gerar os modelos de predição são relatados na legenda da Figura 16. Quanto os outros limites temporais, $\delta$ e $\tau$, estes foram mantidos em 10 e 30 minutos, respectivamente.

Como podemos ver a partir do gráfico à *esquerda* da Figura 16, quanto maior a

Figura 16 – Avaliação do desempenho quando as consultas de predição são realizadas. O gráfico à *esquerda* refere-se aos experimentos sobre os quais os valores dos parâmetros usados são $\sigma = 30$ minutos e resolução da célula fixada em 500 metros. Enquanto que o gráfico à *esquerda* foram usados $\sigma = 10$ minutos e 100 metros para resolução da célula.




Fonte: elaborada pelo autor.

quantidade de células de parada em uma consulta, maior será o tempo necessário para calcular essas consultas. Isto ocorre, principalmente, devido a um aumento do peso computacional na computação da similaridade (Seção 4.4.1) necessário para encontrar os nós adequados nas árvores de sufixo probabilística para realizar as predições. De qualquer forma destacamos que os tempos totais de execução são da ordem de alguns milissegundos.

No gráfico à *direita* temos o mesmo tipo de experimentos realizados no gráfico à *esquerda*, embora a resolução da célula e $\sigma$ sejam menores. E isto nos lembra o efeito desses parâmetros em aumentar a quantidade de células de parada detectadas a partir dos dados, que por sua vez aumenta a altura e tamanho das árvores. A partir do gráfico à *direita* observa-se que os tempos médios de execução são mais elevados, devido ao aumento da altura e tamanho das árvores, mas são, no entanto, menor do que um décimo de segundo.

## 5.3 Resumo

Neste capítulo analisamos o contexto em que os experimentos foram realizados, realizamos uma avaliação experimental e comparamos o desempenho do TPRED com outros dois *baselines*. Observamos que nosso *framework* obteve os resultados mais satisfatórios, aumentando assim a confiabilidade das predições realizadas. Como vimos foram utilizadas diferentes métricas que mostram a superioridade da nossa abordagem proposta.

# 6 CONCLUSÃO

Neste trabalho, propomos um *framework* chamado TPRED baseado em árvore de sufixo probabilística, que considera as informações, nos domínios espacial e temporal para aprender o padrão de mobilidade dos objetos em movimento e, como consequência, computar predições confiáveis a partir dos movimentos recentes de um objeto em movimento e de um tempo de consulta. Verificamos a validade de nossa contribuição através da realização de um extenso conjunto de avaliações experimentais sobre dois conjuntos de dados do mundo real e usamos diferentes métricas para mensurar o nosso desempenho ao compararmos a nossa abordagem com outros dois *baselines*, e demonstramos a sua eficiência e eficácia.

## 6.1 Resultados Alcançados

Como vimos na avaliação experimental o *framework* TPRED é capaz de superar os *baselines* utilizados, tanto nos componentes espaciais quanto nos temporais, nos experimentos que realizamos.

## 6.2 Trabalhos Futuros

Para trabalhos futuros, pretendemos otimizar o TPRED para o processamento de computação altamente distribuída, aproveitando-se de estruturas de computação distribuída populares e eficientes, como *Spark*. Desejamos analisar o impacto na performance ao alterarmos o modelo preditivo usando a abordagem de uma árvore de sufixo probabilística clássica. Pretendemos também definir uma forma melhor para identificarmos as células de parada no domínio espacial, nosso objetivo é eliminar o parâmetro *resolução da célula*. Além disso, pretendemos explorar outros tipos de domínios, além do espaço e do tempo, e outras informações, por exemplo, as redes sociais, as condições climáticas e as condições de tráfego, para impulsionar o modelo preditivo nosso *framework* e a precisão das predições. Finalmente, nós encaramos a possibilidade de gerar modelos preditivos relacionados a grupos de objetos em movimento que são caracterizados por padrões de mobilidade semelhantes; na verdade, consideramos que tais pesquisas podem fornecer uma melhor compreensão sobre padrões de mobilidade similares entre os objetos em movimento, bem como dar ainda mais possibilidades de idealizar aplicações interessantes do mundo real.

# REFERÊNCIAS

ALVARES, L. A.; BOGORNY, V.; KUIJPERS, B.; MOELANS, B.; MACEDO, J. A. F. D.; PALMA, A. T. Towards semantic trajectory knowledge discovery. **Technical Report, Hasselt University, Limbourg, Belgium**, October 2007.

APPEL, A. P. Métodos para o pré-processamento e mineração de grandes volumes de dados multidimensionais e redes complexas. **Tese de Doutorado de Ciências de Computação e Matemática Computacional**, Universidade de São Paulo, Campus de São Carlos, 2010.

BARABASI, A.-L.; CRANDALL, R. E. **Linked: The New Science of Networks**. [*S. l.*]: Perseus Publishing, 2002. v. 71. 280 p. ISBN 0738206679.

BEJERANO, G.; YONA, G. Variations on probabilistic suffix trees: statistical modeling and prediction of protein families. v. 17, p. 23–43, 2001.

BERGROTH, L.; HAKONEN, H.; RAITA, T. A survey of longest common subsequence algorithms. **String Processing and Information Retrieval, 2000. SPIRE 2000. Proceedings. Seventh International Symposium on**, IEEE, p. 39–48, 2000.

BOBADILLA, J.; ORTEGA, F.; HERNANDO, a.; GUTIÉRREZ, a. Recommender systems survey. **Knowledge-Based Systems**, Elsevier B.V., v. 46, p. 109–132, jul 2013. ISSN 09507051.

BOGORNY, V.; KUIJPERS, B.; ALVARES, L. O. St-dmql: A semantic trajectory data mining query language. **International Journal of Geographical Information Science**, v. 54, n. 10, p. 1245–1276, October 2009.

CHEN, D.-B.; XIAO, R.; ZENG, A. Predicting the evolution of spreading on complex networks. **Scientific reports**, v. 4, p. 6108, 2014. ISSN 2045-2322.

DHAR, S.; VARSHNEY, U. Challenges and business models for mobile location-based services and advertising. **Communications of the ACM**, v. 54, p. 121–128, 2011.

GAMBS, S.; KILLIJIAN, M.; NúñEZ, M.; CORTEZ, P. Show Me How You Move and I Will Tell You Who You Are. v. 4, p. 103–126, 2011.

GAMBS, S.; KILLIJIAN, M.; NúñEZ, M.; CORTEZ, P. Next Place Prediction using Mobility Markov Chains. v. 4, n. 3, p. 103–126, 2012.

GIANNOTTI, F.; NANNI, M.; PEDRESCHI, D. Efficient mining of temporally annotated sequences. **In Proc. SDM'06**, p. 346–357, 2006.

GIANNOTTI, F.; NANNI, M.; PEDRESCHI, D.; PINELLI, F. Trajectory Pattern Mining. **Work**, p. 330–339, 2007.

GONZáLEZ, M. C.; HIDALGO, C. A.; BARABASI, A.-L. Understanding individual human mobility patterns. v. 453, 2008.

HAN, J.; KAMBER, M. **Data mining - concepts and techniques. 1st edition ed.** [*S. l.*]: New York: Morgan Kaufmann Publishers, 2000.

HORNE, J. S.; GARTON, E. O.; KRONE, S. M.; LEWIS, J. S. Analyzing Animal Movements Using Brownian Bridges. **Ecology (Volume 88, Issue 9 September 2007 )**, v. 88, p. 2354–2363, 2007.

KANG, J.; YONG, H.-S. Spatio-temporal discretization for sequential pattern mining. **ICUIMC '08 Proceedings of the 2nd international conference on Ubiquitous information management and communication**, ICUIMC, 2008.

LEI, P.; LI, S.; PENG, W. QS-STT: QuadSection clustering and spatial-temporal trajectory model for location prediction. **Distributed and Parallel Databases**, v. 31, n. 2, p. 231–258, 2013. ISSN 09268782.

LIU, J.; WOLFSON, O.; YIN, H. Extracting Semantic Location from Outdoor Positioning Systems. p. 73, 2006.

MONREALE, A.; PINELLI, F.; TRASARTI, R.; GIANNOTTI, F. WhereNext: a location predictor on trajectory pattern mining. **Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining**, ACM, p. 637–645, 2009.

NEWMAN, M. E. J.; BARABASI, A.-L.; WATTS, D. J. **The Structure and Dynamics of Networks**. [*S. l.*]: Princeton University Press, 2006. v. 11. 624 p. (Princeton studies in complexity, 4). ISSN 13681613. ISBN 0691113572.

NG, E. K. K.; FU, A. W.-c.; WONG, R. C.-W. Projective clustering by histograms. **Knowledge and Data Engineering, IEEE Transactions on**, IEEE, v. 17, n. 3, p. 369–383, 2005.

NOULAS, A.; SCELLATO, S.; LATHIA, N.; MASCOLO, C. Mining User Mobility Features for Next Place Prediction in Location-Based Services. p. 1038–1043, 2012.

ROCHA, C. L.; BRILHANTE, I. R.; LETTICH, F.; MACEDO, J. A. F. D.; RAFFAETà, A.; ANDRADE, R.; ORLANDO, S. Tpred: a spatio-temporal location predictor framework. **Data Mining, OLAP, and Knowledge Discovery (IDEAS '16)**, IDEAS, July 2016.

RON, D.; SINGER, Y.; TISHBY, N. The Power of Amnesia: Learning Probabilistic Automata with Variable Memory Length. v. 25, p. 117–149, 1996.

SNYDER, J. P. Map Projections: A Working Manual. **U.S. Geological Survey Professional Paper 1395**, p. 154–163, 1987. ISSN 00941689.

SPACCAPIETRA, S.; PARENT, C.; DAMIANI, M. L.; MACEDO, J. A. de; PORTO, F.; VANGENOT, C. A conceptual view on trajectories. **Data & knowledge engineering**, Elsevier, v. 65, n. 1, p. 126–146, 2008.

SUN, Z.; HAN, L.; HUANG, W.; WANG, X.; ZENG, X.; WANG, M.; YAN, H. Recommender systems based on social networks. **Journal of Systems and Software**, v. 99, p. 109–119, 2015. ISSN 01641212.

TCHEBICHEF, P. Sur les valeurs limites des intégrales. **Journal de Mathématiques Pures et Appliquées, série 2.**, p. 157–160, 1874.

UKKONEN, E. On-line construction of suffix trees. v. 14, p. 249–260, 1995.

XUE, A. Y.; ZHANG, R.; ZHENG, Y.; XIE, X.; YU, J.; TANG, Y. DesTeller: a system for destination prediction based on trajectories with privacy protection. **Proc. VLDB**, v. 6, n. 12, p. 1198–1201, 2013. ISSN 2150-8097.

XUE, A. Y.; ZHANG, R.; ZHENG, Y.; XIE, X.; HUANG, J.; XU, Z. Destination prediction by sub-trajectory synthesis and privacy protection against such prediction. **Proceedings - International Conference on Data Engineering**, p. 254–265, 2013. ISSN 10844627.

YE, J.; ZHU, Z.; CHENG, H. What's Your Next Move: User Activity Prediction in Location-based Social Networks. p. 9, 2013.

YING, J. J.-C.; LEE, W.-C.; TSENG, V. S. Mining Geographic-Temporal-Semantic Patterns in Trajectories for Location Prediction. v. 5, n. 2, 2013.

YING, J. J.-C.; LEE, W.-C.; WENG, T.-C.; TSENG, V. S. Semantic trajectory mining for location prediction. p. 34–43, 2011.

ZHENG, Y.; XIE, X.; MA, W.-Y. Geolife: A collaborative social networking service among user, location and trajectory. **IEEE Data(base) Engineering Bulletin**, IEEE, June 2010.

ZHU, W.-Y.; PENG, W.-C.; HUNG, C. C.; LEI, P.-R.; CHEN, L.-J. Exploring Sequential Probability Tree for Movement-Based Community Discovery. v. 26, p. 2717 – 2730, 2014. ISSN 1041-4347.