



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS QUIXADÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO
MESTRADO ACADÊMICO EM COMPUTAÇÃO

JOSÉ SOARES DA SILVA NETO

**AVALIANDO MODELOS DE PREDIÇÃO DE PRÓXIMO SENSOR PARA
TRAJETÓRIAS DE VEÍCULOS ROUBADOS**

QUIXADÁ

2021

JOSÉ SOARES DA SILVA NETO

AVALIANDO MODELOS DE PREDIÇÃO DE PRÓXIMO SENSOR PARA TRAJETÓRIAS
DE VEÍCULOS ROUBADOS

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

Orientadora: Profa. Dra. Ticiane Linhares Coelho da Silva

Coorientador: Prof. Dr. Régis Pires Magalhães

QUIXADÁ

2021

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

S58a Silva Neto, José Soares da.
Avaliando Modelos de Predição de Próximo Sensor para Trajetórias de Veículos Roubados / José Soares da Silva Neto. – 2021.
66 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Campus de Quixadá, Programa de Pós-Graduação em Computação, Quixadá, 2021.

Orientação: Profa. Dra. Ticiane Linhares Coelho da Silva.

Coorientação: Prof. Dr. Régis Pires Magalhães.

1. Aprendizagem Profunda. 2. Trajetórias. 3. Trajetórias Semânticas. I. Título.

CDD 005

JOSÉ SOARES DA SILVA NETO

AVALIANDO MODELOS DE PREDIÇÃO DE PRÓXIMO SENSOR PARA TRAJETÓRIAS
DE VEÍCULOS ROUBADOS

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

Aprovada em: __/__/__.

BANCA EXAMINADORA

Profa. Dra. Ticiania Linhares Coelho da
Silva (Orientadora)
Universidade Federal do Ceará (UFC)

Prof. Dr. Régis Pires Magalhães (Coorientador)
Universidade Federal do Ceará (UFC)

Profa. Dra. Vania Bogorny
Universidade Federal de Santa Catarina (UFSC)

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

Prof. Dr. Vinicius Monteiro de Lira
Institute of Information Science and Technologies
"Alessandro Faedo" (ISTI-CNR)

À Deus, que sempre esteve ao meu lado. À
minha minha família, que me inspira todos os
dias.

AGRADECIMENTOS

À Deus, que fez dos meus sonhos seus planos, me deu coragem para seguir acreditando e pôs pessoas maravilhosas no caminho.

Aos meus pais e irmãos, que apoiaram de forma incondicional esse projeto. À minha esposa Vanina e aos meus filhos Heitor e Helena, pelo amor e compreensão nos momentos de ausência.

À minha orientadora Ticiania Linhares, pelo incentivo generoso que me ajudou a continuar acreditando e lutando até o fim.

Aos amigos e amigas do programa de Pós-Graduação em Computação da UFC Campus Quixadá e do Insight Data Science Lab, cujo o apoio e colaboração foram fundamentais para esta conquista.

Aos professores e pesquisadores José Macedo, Regis Magalhães, Lívia Almada, Vinícius Lira, Pedro Alves e Paulo César Fernandes que contribuíram profundamente neste trabalho de pesquisa.

Por fim, ao Doutorando em Engenharia Elétrica, Ednardo Moreira Rodrigues, e seu assistente, Alan Batista de Oliveira, aluno de graduação em Engenharia Elétrica, pela adequação do *template* utilizado neste trabalho para que o mesmo ficasse de acordo com as normas da biblioteca da Universidade Federal do Ceará (UFC).

“A única forma de encontrar os limites do possível é indo além dele, para dentro do impossível.”

(Arthur C. Clark)

RESUMO

O crescimento da disponibilidade de dados de trajetórias impulsionou o surgimento de diversas técnicas para a predição da mobilidade humana nos últimos anos. Tais dados são coletados de inúmeras formas, seja por aplicações instaladas em *smartphones*, seja por sistemas de vigilância de tráfego instalados nas redes de ruas. Neste último caso, esses sistemas produzem trajetórias a partir de sensores externos (*External Sensor Trajectories* - EST) que são utilizadas em pesquisas e aplicações para a predição da próxima localização. Sistemas de vigilância registram, entre outras características, eventuais restrições de roubo ou furto do veículo. Conhecer a dinâmica da movimentação dos veículos roubados no espaço urbano é uma informação crucial para agências governamentais de segurança. No entanto, a predição da próxima localização (neste trabalho, sensor) de um veículo roubado é desafiador, por conta da baixa regularidade de transição entre os locais, além da heterogeneidade e escassez de dados de trajetórias. Diante disso, este trabalho propõe um modelo de rede neural semanticamente enriquecido, analisa a eficácia de modelos classificadores de aprendizado de máquina, investiga quais são os melhores atributos para os modelos de previsão em EST para veículos roubados e como diferentes níveis de representação do dado espacial podem afetar a predição. Como avaliação do desempenho dos modelos, são realizados experimentos com um conjunto de dados de uma aplicação real. Os resultados demonstraram a efetividade dos modelos classificadores de aprendizado de máquina enriquecidos com dados semânticos na predição da próxima localização em EST, que a categoria de atributos relacionados a Pontos de Interesse contribuíram mais que os dados criminais para a maioria dos modelos testados e que representações do dado espacial com maior granularidade favorecem a solução proposta.

Palavras-chave: Aprendizagem Profunda. Trajetórias. Trajetórias semânticas.

ABSTRACT

The growing availability of trajectory data has driven the emergence of different techniques for predicting human mobility in recent years. Such data are collected either by applications installed on smartphones or by traffic monitoring systems in street networks. In this case, these systems are trajectories from external sensors (External Sensor Trajectories - EST) used in research and applications to the next location prediction. Record surveillance systems, among other property, restrictions on vehicle theft or theft. Knowing the dynamics of the movement of stolen vehicles in urban space is crucial information for government security agencies. However, the next location prediction (in this work, sensor) of a stolen vehicle is challenging due to the low regularity of transitions and the heterogeneity and scarcity of trajectory data. Therefore, this work offers a semantically enriched neural network model, analyzes the effectiveness of different machine learning models, investigates the best attributes for EST prediction models for stolen vehicles, and how different levels of spatial data representation can affect the prediction. We evaluate our model on a real dataset. The results demonstrated the effectiveness of machine learning models enriched with semantic data in predicting the next location in EST. The category of attributes related to Points of Interest contributed more than criminal data to most of the models tested, and that data representations with greater granularity favor the proposed solution.

Keywords: Deep Learning. Trajectories. Semantic trajectories.

LISTA DE FIGURAS

| | |
|---|----|
| Figura 1 – Observações de um veículo sem restrição de Roubo | 19 |
| Figura 2 – Observações de um veículo com restrição de Roubo | 20 |
| Figura 3 – Distribuição Observações do Conjunto de Veículos sem restrição de Roubo em Dias da Semana e Hora do Dia | 21 |
| Figura 4 – Distribuição Observações do Conjunto de Veículos com restrição de Roubo em Dias da Semana e Hora do Dia | 22 |
| Figura 5 – Exemplo de uma trajetória obtida por uma rede de sensores externos | 25 |
| Figura 6 – Técnica de mapeamento de pontos | 30 |
| Figura 7 – Técnica de mapeamento de KDE | 30 |
| Figura 8 – Técnica de mapeamento de pontos | 32 |
| Figura 9 – Representação de uma rede neural recorrente - <i>Recurrent Neural Network</i> (RNN) | 34 |
| Figura 10 – Representação de uma rede neural LSTM | 35 |
| Figura 11 – Conjunto de Dados de Trajetórias de Sensores Externos. | 44 |
| Figura 12 – Identificação de eventos criminais nas <i>grid cells</i> dos sensores. | 47 |
| Figura 13 – Anotação das Distâncias para Pontos de Interesse. | 47 |
| Figura 14 – Frequência dos sensores no conjunto de rótulos. | 50 |
| Figura 15 – Processo de Preparação dos dados e de Treinamento dos Modelos. | 51 |
| Figura 16 – Identificação do Menor Caminho entre dois Sensores. | 52 |
| Figura 17 – Técnica da Janela Deslizante. | 54 |
| Figura 18 – Arquitetura do Modelo BiLSTM Utilizado | 55 |
| Figura 19 – Análise do Erro de Proximidade para o grupo Espaço-Tempo. | 59 |
| Figura 20 – Análise do Erro de Proximidade para o grupo Espaço-Tempo-Crime. | 59 |
| Figura 21 – Análise do Erro de Proximidade para o grupo Espaço-Tempo-Poi. | 60 |
| Figura 22 – Análise do Erro de Proximidade para o grupo Todos os Atributos. | 60 |

LISTA DE TABELAS

| | |
|--|----|
| Tabela 1 – Análise da Acurácia (Acc) e Erro Absoluto do Médio do Erro de Proximidade | |
| - MAE | 58 |
| Tabela 2 – Análise do Nível da Representação Espacial | 61 |

LISTA DE QUADROS

| | |
|--|----|
| Quadro 1 – Comparação entre os trabalhos relacionados e o proposto | 42 |
| Quadro 2 – Detalhes do conjunto de dados criminais | 44 |
| Quadro 3 – Detalhes do conjunto de dados de pontos de interesse | 45 |
| Quadro 4 – Configuração de Hiperparâmetros | 53 |
| Quadro 5 – Grupos de Atributos | 56 |

LISTA DE ALGORITMOS

| | |
|---|----|
| Algoritmo 1 – Algoritmo Split Estadístico | 49 |
|---|----|

LISTA DE ABREVIATURAS E SIGLAS

| | |
|--------|---|
| API | <i>Application Programming Interface</i> |
| BiLSTM | <i>Bidirectional Long Short-Term Memory</i> |
| CVP | Crime Violento Contra o Patrimônio |
| EST | <i>External Sensor Trajectory</i> |
| GBRT | <i>Gradient Boosting Regression Trees</i> |
| ICTAI | <i>IEEE International Conference on Tools with AI</i> |
| KDE | <i>Kernel Density Estimator</i> |
| KNN | <i>K-Nearest Neighbors</i> |
| LSTM | <i>Long Short-Term Memory</i> |
| MAE | <i>Mean Absolut Error</i> |
| MDM | <i>Mobile Data Management</i> |
| POI | <i>Point of Interest</i> |
| RF | <i>Random Forest</i> |
| RNN | <i>Recurrent Neural Network</i> |

SUMÁRIO

| | | |
|----------------|---|-----------|
| 1 | INTRODUÇÃO | 16 |
| 1.1 | Questões de Pesquisa | 21 |
| 1.2 | Contribuições | 22 |
| 1.3 | Publicações e Submissões | 23 |
| 1.4 | Estrutura da Dissertação | 23 |
| 2 | DEFINIÇÃO DO PROBLEMA E CONCEITOS PRELIMINARES | 24 |
| 2.1 | Pontos e Trajetórias | 24 |
| 2.2 | Trajetoórias de Sensores Externos (External Sensor Trajectory) - EST | 24 |
| 2.3 | Definição do Problema | 26 |
| 3 | FUNDAMENTAÇÃO TEÓRICA | 27 |
| 3.1 | Aspectos Espaciais Relacionados ao Crime | 27 |
| 3.1.1 | <i>Pontos de Interesse</i> | 28 |
| 3.1.2 | <i>Hotspots de Crimes e Kernel Density Estimation - Kernel Density Estimator (KDE)</i> | 29 |
| 3.2 | Modelos de Aprendizado de Máquina | 30 |
| 3.2.1 | <i>K-Nearest Neighbors Classifier - KNN</i> | 31 |
| 3.2.2 | <i>Random Forest</i> | 31 |
| 3.2.3 | <i>Gradient Boost Machine</i> | 32 |
| 3.2.4 | <i>Redes Neurais Recorrentes - RNN</i> | 33 |
| 3.2.4.1 | <i>Long Short-Term Memory (LSTM)</i> | 34 |
| 3.2.4.2 | <i>Long Short-Term Memory (LSTM) Bidirecional (Bidirectional Long Short-Term Memory (BiLSTM))</i> | 36 |
| 4 | TRABALHOS RELACIONADOS | 37 |
| 4.1 | Trajetoórias de Sensores Externos ou Esparsas | 37 |
| 4.2 | Trajetoórias com Dados Semânticos | 40 |
| 4.3 | Resumo dos Trabalhos Relacionados | 41 |
| 5 | MATERIAIS E MÉTODOS | 43 |
| 5.1 | Introdução | 43 |
| 5.2 | Conjunto de Dados de Trajetórias | 43 |
| 5.3 | Conjunto de Dados Semânticos | 44 |

| | | |
|---------|--|----|
| 5.3.1 | <i>Dados Criminais</i> | 44 |
| 5.3.2 | <i>Pontos de Interesse</i> | 45 |
| 5.4 | Procedimentos | 45 |
| 5.4.1 | <i>Pré-processamento</i> | 45 |
| 5.4.2 | <i>Engenharia de Features e Enriquecimento Semântico</i> | 45 |
| 5.4.2.1 | <i>Dado Espacial</i> | 46 |
| 5.4.2.2 | <i>Dado Temporal</i> | 46 |
| 5.4.2.3 | <i>Enriquecimento Semântico com Dados Criminais</i> | 46 |
| 5.4.2.4 | <i>Enriquecimento Semântico com Dados de Pontos de Interesse</i> | 47 |
| 5.4.3 | <i>Imputação de Dados</i> | 48 |
| 5.4.4 | <i>Segmentação das Trajetórias</i> | 48 |
| 5.4.5 | <i>Análise dos Sensores Representativos</i> | 50 |
| 5.4.6 | <i>Métricas de Avaliação</i> | 51 |
| 6 | EXPERIMENTAÇÃO | 53 |
| 6.1 | Configuração do Experimento | 53 |
| 6.1.1 | <i>Janela Deslizante</i> | 54 |
| 6.1.2 | <i>Treinamento dos Modelos</i> | 54 |
| 6.1.2.1 | <i>Preparação dos Dados de Entrada para os Classificadores de Aprendizado Supervisionado</i> | 54 |
| 6.1.2.2 | <i>Preparação dos Dados de Entrada para a BiLSTM</i> | 55 |
| 6.2 | Grupos de Atributos | 56 |
| 6.3 | Resultados | 57 |
| 7 | CONCLUSÕES E TRABALHOS FUTUROS | 62 |
| | REFERÊNCIAS | 63 |

1 INTRODUÇÃO

Com o recente crescimento da disponibilidade de dados de trajetórias, surgiram métodos modernos de análise de padrões de mobilidade de objetos e pessoas. No estudo da mobilidade uma tarefa recorrente é a predição da próxima localização. Tal tarefa pode ser definida como inferir a localização que um objeto em movimento irá visitar baseado no conjunto de suas trajetórias históricas. Essa informação é considerada o alicerce de inúmeras aplicações como transporte inteligente, serviços compartilhados, controle de tráfego, planejamento urbano, sistemas de recomendação, entre outros.

Dados de trajetórias fornecem informações sobre o local e a identificação do objeto em um dado momento e também disponibilizam outros dados como duração do movimento, distância percorrida e aceleração. Tais objetos podem ser pessoas, veículos, animais ou fenômenos naturais.

A mobilidade pode ser capturada de forma ativa ou passiva. Exemplos de registro ativo são atividades em redes sociais em que viajantes compartilham suas experiências com seus amigos através de fotos e comentários que apresentam marcações geográficas e temporais. Já o registro passivo pode ser gerado sem intenção dos usuários quando transportam seus telefones móveis. Nesse caso, as trajetórias são representadas pela sequência de células das torres de telefonia com a marcação temporal da transição entre elas (ZHENG, 2015).

De forma análoga, o movimento de veículos pode ser capturado a partir de equipamentos de GPS instalados nos veículos como táxis, ônibus e embarcações, permitindo que suas posições sejam conhecidas em tempo real. Outra forma de registro do movimento de veículos é através da observação de suas passagens em sistemas de vigilância de tráfego.

Existem diversos tipos de análises sobre a mobilidade. Bucher (2017) apresenta três níveis de predição: 1) posição do objeto; 2) predição de trajetória; e 3) predição da próxima localização. Já as técnicas existentes para predição da próxima localização, tema de estudo deste trabalho, podem ser divididas em duas categorias: baseada em padrões e baseada em modelos (YAO *et al.*, 2017). Os métodos baseados em padrões buscam extrair padronizações da movimentação do usuário a partir de dados históricos e utilizam padrões explícitos definidos anteriormente para inferir a próxima localização. Essa técnica não captura toda a regularidade do dado, pois extrai informações apenas dos padrões já conhecidos. Por outro lado, os métodos baseados em modelos capturam a regularidade do movimento do usuário a partir de métodos estatísticos e realizam predições com os modelos desenvolvidos. Diversos modelos preditivos

foram propostos, dentre eles estão os modelos baseados em cadeias de Markov, como os apresentados nos trabalhos de Mathew *et al.* (2012), bem como em Zhang *et al.* (2016), fatoração matricial a exemplo de Duong-Trung *et al.* (2016) e mais recentemente, modelos de redes neurais recorrentes (RNN), tais como (YAO *et al.*, 2017) e (CRUZ *et al.*, 2019).

Outro fator que diferencia as diversas abordagens para predição da próxima localização é o tipo de trajetória utilizada. Grande parte dos trabalhos utiliza dados brutos de GPS, onde o equipamento que fornece a localização acompanha o objeto analisado, como os trabalhos de (FENG *et al.*, 2018), (ROCHA *et al.*, 2016) e (TRASARTI *et al.*, 2017). Porém, (CRUZ *et al.*, 2019) propõe a predição do movimento de objetos cujas trajetórias são capturadas por sensores externos, como câmeras e sensores de vigilância de tráfego localizados ao longo da rede viária. Nessa abordagem, chamada pelos autores de *External Sensor Trajectory (EST)*, a trajetória de um objeto é capturada pela rede de sensores, assumindo que esta rede consegue identificar unicamente cada objeto em movimento.

Tais redes de sensores externos integram sistemas que ao registrarem a passagem de veículos, fornecem diversos outros dados como por exemplo, histórico de multas, características do veículo ou eventuais restrições de roubo e furto. Neste contexto, esse trabalho propõe um modelo para predição da próxima localização de veículos com restrição de roubo ou furto utilizando aprendizado profundo a partir de suas trajetórias observadas por sensores localizados em uma rede viária. Essa predição é útil para agências governamentais, em especial para aplicações de patrulhamento policial no rastreamento de criminosos e fornecimento de informações sobre a dinâmica do crime de roubo de veículos.

No entanto, a análise de trajetórias oriundas de dispositivos externos apresenta vários desafios (CRUZ *et al.*, 2019):

1. Grande volume de dados: Os sensores realizam uma enorme quantidade de registros de passagem de veículos. O gerenciamento das tarefas de computar as últimas observações dos veículos e realizar predições em tempo real se torna complexo à medida que o número de veículos aumenta.
2. Diversos tipos de trajetórias: As trajetórias de táxis, veículos particulares e veículos utilizados em serviços de entregas podem apresentar padrões bem distintos, inclusive com influência do aspecto temporal. Por exemplo, veículos particulares tendem a apresentar a rota partindo da residência para o trabalho nas manhãs dos dias úteis e terem a viagem no sentido contrário no fim do dia. Entretanto, o comportamento de táxis ou de outros

veículos de aluguel dependem dos destinos de seus passageiros.

3. Esparsidade: Os sensores geralmente são instalados apenas nas principais ruas de uma cidade, dessa forma, a viagem de um veículo dificilmente é capturada por completo. Essa característica pode gerar trajetórias com poucos pontos, o que dificulta a tarefa da predição da localização.
4. Incompletude das trajetórias: Além dos sensores não estarem disponíveis em toda a rede viária, tais equipamentos ainda podem falhar por diversos motivos na tarefa de capturar a passagem dos veículos, produzindo trajetórias incompletas.

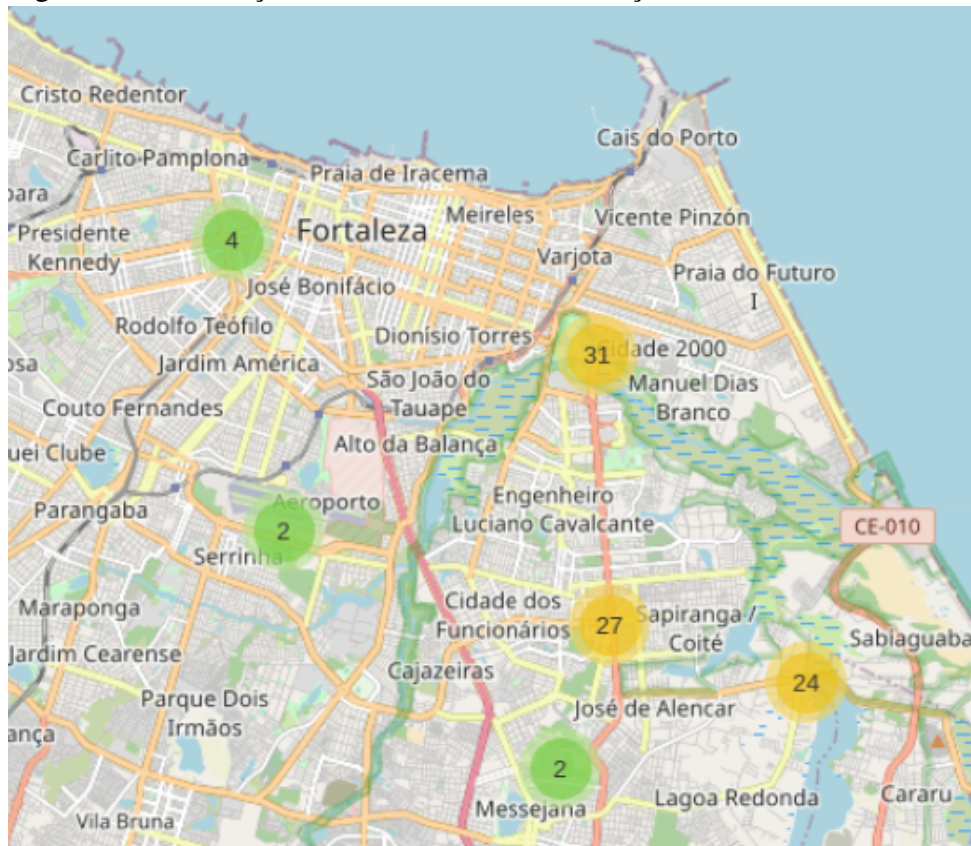
Quando analisadas as trajetórias de veículos roubados, também são incluídos neste trabalho alguns desafios da análise criminal, como os listados por Huang *et al.* (2018):

1. Os motoristas de veículos roubados tendem a se afastar dos sensores, o que aumenta a dispersão dos dados;
2. Conforme declarado em (HUANG *et al.*, 2018), os padrões de crime são dinâmicos, uma vez que os fatores subjacentes às ocorrências de crime podem mudar com o tempo. Os crimes podem ocorrer em diferentes regiões geográficas comparando dias de semana e fins de semana;
3. Os criminosos tendem a evitar alguns Pontos de Interesse, *Point of Interest* (POI), como instalações de segurança. Além disso, o trabalho Oróstica e Poblete (2019) demonstra que o roubo de carros tem forte relação com locais de interesse social como restaurantes e shopping centers.

As Figuras 1 e 2 apresentam a distribuição das observações de um mesmo veículo com e sem a restrição de roubo e furto, respectivamente. Ou seja, no momento das observações da passagem do veículo presentes na Figura 1, tal veículo não apresentava a restrição de roubo. De outro modo, na Figura 2, as observações da passagem do veículo apresentam a restrição. Por meio das imagens, é possível perceber uma diferença na distribuição espacial dessas observações. As observações sem a restrição estão distribuídas por mais regiões da cidade, enquanto as observações durante o período em que o veículo apresentava a restrição de roubo estão mais concentradas. Essa diferença pode ser explicada pelos desafios apresentados anteriormente.

Os desafios da predição da próxima localização de um objeto móvel podem ser melhor entendidos e solucionados a partir do enriquecimento semântico das trajetórias, como demonstrado por Yao *et al.* (2017) e Fan *et al.* (2018). Ao realizar uma análise espacial sobre o crime de roubo de veículos, Weisel *et al.* (2006) afirma que notoriamente existem três principais

Figura 1 – Observações de um veículo sem restrição de Roubo

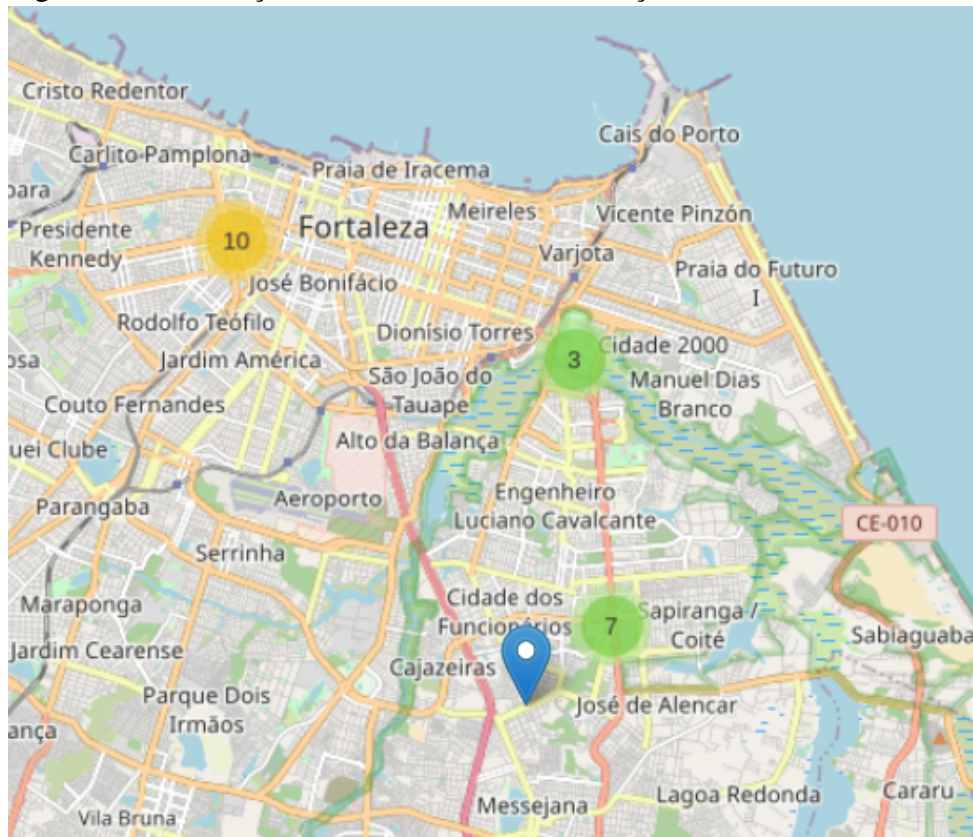


Fonte: Próprio autor (2020)

motivações para esse crime: motivação pelo lucro, no qual o veículo roubado é vendido em regiões afastadas do local do roubo ou o veículo é desmontado para a revenda de peças. Outra motivação é o uso do veículo para o cometimento de outros delitos. Por fim, os veículos são roubados para o uso recreativo. Oróstica e Poblete (2019) demonstram a relação entre pontos de interesse sociais com o crime de roubo de veículo. Dessa forma, espera-se que a inclusão de informações referentes a ocorrências de crime e a proximidade a pontos de interesse social nas trajetórias de veículos roubados possa melhorar predição da próxima localização de veículos com essa restrição.

Para a verificação da relação entre Pontos de Interesse e *hotspots* de eventos criminais com o problema de predição de trajetórias de veículos roubados, nesta pesquisa será proposto o enriquecimento semântico com as seguintes categorias de pontos de interesse: Supermercado, Estacionamento e Oficinas Mecânicas. Essas categorias foram escolhidas devido ao fato de que veículos roubados geralmente são abandonados próximos a esses pontos. Já os *hotspots* criminais avaliados serão os dos seguintes eventos: Crimes Violentos contra o Patrimônio (CVPs), recuperação de veículos roubados e roubo ou furto de veículos. Estas categorias de eventos

Figura 2 – Observações de um veículo com restrição de Roubo



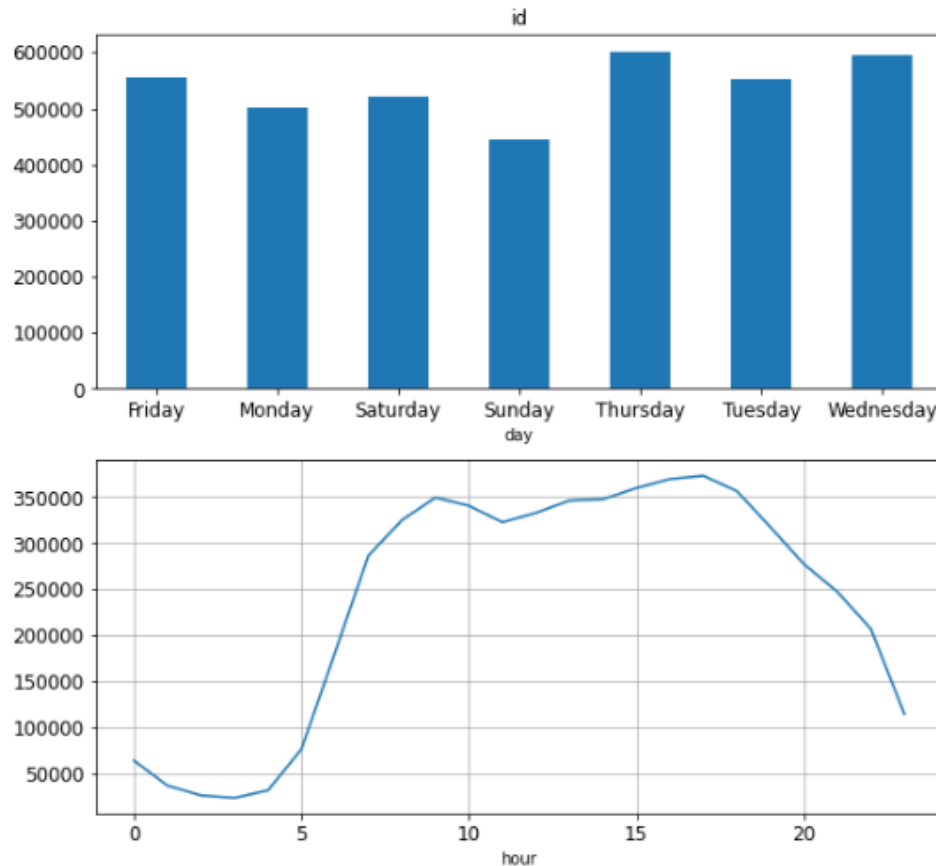
Fonte: Próprio autor (2020)

foram escolhidas devido a relação com as motivações do roubo de veículos já mencionadas.

Uma análise trajetórias de veículos com restrição de roubo e de trajetórias veículos sem restrição podemos observar algumas diferenças do comportamento dos condutores. O tamanho médio das trajetórias sem restrição é 18,9 pontos por trajetória, enquanto no conjunto com restrição em média as trajetórias são compostas por 8,6 pontos. As Figuras 3 e 4 demonstram algumas diferenças na distribuição das observações dos dois conjuntos com relação ao número de observações por dia da semana e por horas do dia. Através das Figuras observa-se que a distribuição das observações dos veículos roubados nos dias da semana é um pouco mais uniforme que a dos veículos não roubados, ambos os conjuntos apresentam uma diminuição nos domingos. A análise do comportamento durante as horas do dia, o conjunto dos veículos não roubados enfatiza os horários de pico entre 07:00h e 08:00h e entre 17:00h e 18:00h, que representam o início e o fim do horário comercial, respectivamente. Enquanto os veículos roubados tem observações relativamente constantes entre 09:00h e 13:00h e apresentam o maior número de observações entre 17:00h e 18:00h.

Diversas estratégias foram apresentadas para a resolução do problema de predição da próxima localização. Como a trajetória gerada pela rede de sensores externos apresenta

Figura 3 – Distribuição Observações do Conjunto de Veículos sem restrição de Roubo em Dias da Semana e Hora do Dia



Fonte: Próprio autor (2020)

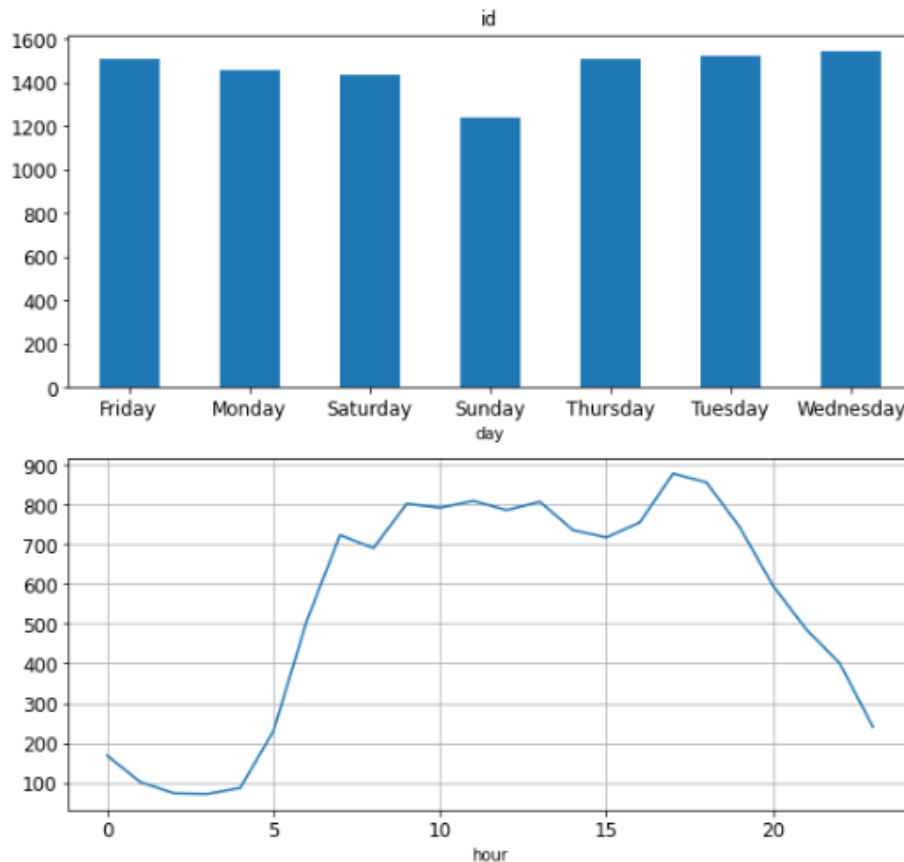
características de uma série temporal, ou seja, é uma sequência de observações consecutivas no tempo (BOX *et al.*, 2015), é viável a aplicação de abordagens eficazes em problemas com dados sequencias, como a utilização de redes neurais recorrentes (RNN) ou, mais especificamente, uma LSTM. A *Long Short-Term Memory* (LSTM) é um tipo de rede neural recorrente densamente conectada que apresenta uma espécie de memória e por isso é utilizada no processamento de sequências e séries temporais (CHOLLET, 2018). Recentemente, a utilização de uma RNN genérica para prever a próxima localização, demonstrou o potencial dessa abordagem para capturar a complexidade da mobilidade em redes de sensores externos (CRUZ *et al.*, 2019). Também podemos modelar EST como uma sequência de etiquetas dos sensores, permitindo aplicar LSTM às previsões de EST para veículos roubados.

1.1 Questões de Pesquisa

A avaliação de nossa proposta será orientada pelas seguintes questões de pesquisa:

1) **QP1:** Qual a eficácia dos modelos de aprendizado supervisionado (KNN, *Random Forest* e

Figura 4 – Distribuição Observações do Conjunto de Veículos com restrição de Roubo em Dias da Semana e Hora do Dia



Fonte: Próprio autor (2020)

Gradient booting) para predição da próxima localização em trajetórias de sensores externos? 2) **QP2:** As informações semânticas propostas neste trabalho podem melhorar a eficácia dos classificadores de aprendizado supervisionado analisados, para predição da próxima localização em trajetórias de sensores externos? 3) **QP3:** Quais atributos mais contribuem para os modelos de predição em EST? 4) **QP4:** Dentre as representações do dado espacial analisadas (*cluster*, *index grid* e *geohash*), qual a melhor para utilizar no modelo RNN de predição de trajetórias de sensores externos?

1.2 Contribuições

O problema abordado nesta dissertação já foi investigado na literatura e foi denominado Predição do Próximo Sensor Externo da Trajetória (*EST Prediction*, em inglês) em (CRUZ *et al.*, 2019; CRUZ *et al.*, 2021), onde a construção das trajetórias não é realizada a partir de dados de GPS, mas sim montadas a partir dos sensores. Ambos trabalhos resolvem o problema usando uma RNN. Esta dissertação apresenta três contribuições principais:

1. Ao contrário de estudos anteriores de previsão de EST, este trabalho analisa a eficácia dos modelos de aprendizado supervisionado em EST, não apenas em modelos RNN;
2. Este trabalho investiga quais informações semânticas ajudam a melhorar a eficácia dos modelos preditivos;
3. Por fim, um estudo experimental é realizado para diferentes representações de nível espacial para trajetórias, bem como é apresentado como cada representação contribui para melhorar o modelo de previsão.

1.3 Publicações e Submissões

Durante o desenvolvimento deste trabalho ocorreu a publicação do artigo *Location prediction: a deep spatiotemporal learning from external sensors data*, publicado no *Journal Distributed and Parallel Databases* e compôs parte da edição especial dos melhores artigos do *Mobile Data Management (MDM) 2019*. O referido artigo apresenta uma abordagem de aprendizado multitarefa para o problema EST. Boa parte dos procedimentos realizados na etapa de pré-processamento das trajetórias da atual pesquisa foram validados nesta publicação.

No momento da escrita dessa dissertação, o artigo intitulado *Predicting the Next Location for Trajectories From Stolen Vehicles* foi aceito para publicação no *IEEE International Conference on Tools with AI (ICTAI) 2021*, onde os experimentos e resultados desta pesquisa são apresentados.

1.4 Estrutura da Dissertação

Este trabalho apresenta a seguinte organização: O Capítulo 2 apresenta a definição formal do problema. Em seguida, o Capítulo 3 discute os conceitos básicos necessários para a compreensão deste trabalho. O Capítulo 4 descreve trabalhos relacionados. O Capítulo 5 detalha a proposta deste trabalho. O Capítulo 6 apresenta as experimentações realizadas. Por fim, o Capítulo 7 apresenta as conclusões e trabalhos futuros.

2 DEFINIÇÃO DO PROBLEMA E CONCEITOS PRELIMINARES

A finalidade deste trabalho é desenvolver um modelo para predição do próximo sensor que um veículo roubado irá passar, dada a trajetória deste veículo constituída a partir de sensores externos e os instantes de tempo que indicam o momento em que o veículo passou pelo sensor. A seguir, são apresentadas as definições de trajetórias brutas, trajetórias de sensores externos, similar à apresentada em (CRUZ *et al.*, 2019), e o problema que o presente trabalho busca resolver.

2.1 Pontos e Trajetórias

Bogorny *et al.* (2014) apresentam o conceito trajetórias brutas como uma lista de pontos localizados no espaço e no tempo. Os autores definem Pontos e Trajetórias da seguinte forma:

Definição 2.1.1 *Ponto.* Um ponto p é uma tupla (x, y, t) , onde x e y são as coordenadas espaciais que representam o local e t é o timestamp em que o ponto é coletado.

A partir da definição acima, trajetórias brutas são delineadas como uma lista ordenada de pontos, como na Definição 2.1.2

Definição 2.1.2 *Trajetoária.* Uma trajetória T é uma lista ordenada de pontos $(p_1, p_2, p_3, \dots, p_n)$, onde $p_i = (x_i, y_i, t_i)$ e $t_1 < t_2 < t_3 < \dots < t_n$.

Diversas características físicas podem ser vinculadas a uma trajetória, como aceleração, velocidade, alteração da direção, entre outras. Além disso, uma trajetória pode ser segmentada em várias subtrajetórias.

2.2 Trajetórias de Sensores Externos (External Sensor Trajectory) - EST

No cenário considerado neste trabalho, os objetos em movimento se deslocam sobre a rede de ruas de uma cidade e suas trajetórias são registradas por sensores localizados ao longo dessa rede em posições fixas, pré-definidas e conhecidas. Assumindo que cada objeto em movimento pode ser identificado unicamente, por exemplo pela placa de um veículo, ao registrar a passagem de um objeto em movimento cada sensor produz a tupla $o = (m, s, t)$ em que:

- m é o identificador do objeto em movimento;

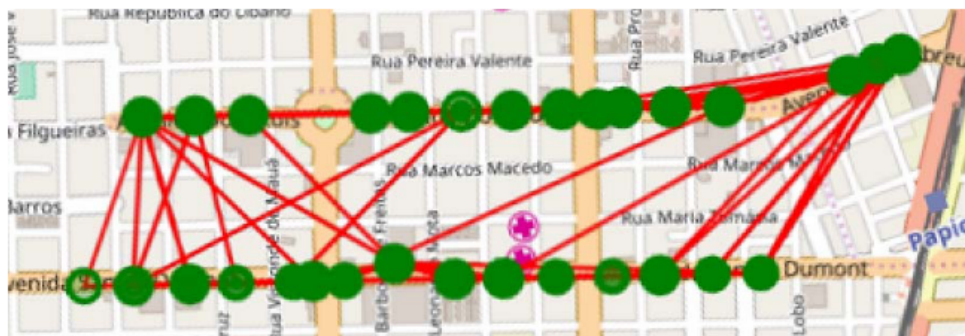
- s é o identificador do sensor;
- t é um *timestamp*.

Vale ressaltar que como as posições dos sensores são fixas e compostas por latitude e longitude, os itens s e t juntos formam uma observação bruta de GPS (x_s, y_s, t) , onde x_s e y_s são as coordenadas espaciais do sensor s . Dessa forma, baseado nos conceitos apresentados em (CRUZ *et al.*, 2019), trajetórias de sensores externos são definidas como:

Definição 2.2.1 (Trajetórias de Sensores Externos) *Dado um conjunto O de observações geradas por um conjunto de sensores S . Onde $O[m] \subset O$ é o conjunto de observações relacionadas ao objeto em movimento m . Dessa forma, a Trajetória de Sensores Externos (EST) do objeto m é definida como a sequência de observações $es_traj_m = (o_1, o_2, \dots, o_j)$, de modo que $o_i = (x_s, y_s, t_i)$ é o i -ésimo ponto da trajetória composta pela localização espacial do sensor $s = (x_s, y_s) \in S$, e um timestamp $t_i \in \mathbb{N}$ para $i = 1, \dots, j$, em que $o_i \in O[m]$. Além disso, $t_1 < t_2 < \dots < t_j$.*

A Figura 5 mostra uma trajetória gerada por sensores externos. Trata-se da trajetória de um ônibus que realiza a mesma viagem várias vezes. Os pontos em verde representam as posições dos sensores que são conectados pelas linhas em vermelho. A figura também exhibe eventuais falhas, quando alguns sensores estão conectados ao próximo sensor esperado. Diferentemente de trajetórias brutas, onde o objeto em movimento está equipado por um dispositivo de GPS que fornece os dados espaciais de latitude e longitude, as trajetórias de sensores externos são menos densas e uma análise sobre essas trajetórias deve considerar a escassez e a incompletude que podem ser causadas por exemplo, devido à presença de sensores apenas em ruas principais da cidade, além de eventuais falhas destes sensores (CRUZ *et al.*, 2019).

Figura 5 – Exemplo de uma trajetória obtida por uma rede de sensores externos



Fonte: Cruz *et al.* (2019)

2.3 Definição do Problema

O problema que o presente trabalho deseja resolver é definido da seguinte forma:

Definição 2.3.1 (*Predição de Trajetórias em Rede de Sensores Externos para Veículos Roubados*) Seja G a rede de ruas, S o conjunto de sensores externos implantados sobre G , O o conjunto de observações históricas produzidas por S , R o conjunto de veículos com registro de roubo em O , e T_{EST} o conjunto de trajetórias históricas derivadas de O que descrevem o padrão de comportamento dos veículos em R . Dado as últimas observações N de um veículo roubado $r \in R$ produzida por O , o problema consiste na predição do próximo sensor s a ser visitado por r .

3 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são descritos os conceitos fundamentais utilizados no decorrer deste trabalho. Por se tratar de uma análise de trajetórias de veículos com restrição de roubo ou furto, na Seção 3.1 é realizada uma breve discussão sobre alguns aspectos relacionados ao roubo de veículos. Em seguida, a Seção 3.2 descreve os Modelos de Aprendizado de Máquina avaliados nesta pesquisa, desde Modelos de Aprendizado Supervisionado às Redes Neurais Recorrentes, uma das principais abordagens utilizadas na resolução do problema de predição da próxima localização.

3.1 Aspectos Espaciais Relacionados ao Crime

A ocorrência de crimes em uma cidade tem forte impacto negativo no cotidiano dos cidadãos, exigindo dos entes públicos operações em tempo hábil e, quando possível, ações de prevenção que aumentem a sensação de segurança da comunidade. Alguns métodos foram propostos com o objetivo de capturar e explorar o padrão dinâmico do crime e prover predições confiáveis da ocorrência de delitos que possam auxiliar agências governamentais em ações preventivas.

A análise criminal é algo bem mais amplo do que apenas a verificação de ocorrência de ações delituosas. Segundo IACA *et al.* (2014), a análise criminal é um processo no qual um conjunto de técnicas quantitativas e qualitativas é usado para analisar dados valiosos para as agências policiais e suas comunidades. Tal análise pode incluir dados de criminosos, vítimas, questões relacionadas à qualidade de vida, dentre outros aspectos que podem ser utilizados para apoiar ações e estratégias que visam a prevenção e redução de crimes, além de resolução de problemas e avaliação dos esforços policiais.

Uma das áreas de estudo da análise criminal é o mapeamento de crimes. Processo que pode ser definido como a utilização de um sistema de informações geográficas para realizar análises espaciais de problemas de crimes e desordens, bem como outras questões relacionadas à polícia (SANTOS, 2016). A partir do momento em que mapas disponibilizam características geográficas das áreas de atuação, os departamentos de polícia utilizam esses mapas para determinar várias de suas ações. Dessa forma, o mapeamento de crime se tornou ferramenta fundamental na análise criminal.

De forma particular sobre o roubo de veículos, a análise criminal aponta as principais

motivações para este delito, que de acordo com Weisel *et al.* (2006) são: (a) Utilização para cometimento de outros delitos, como roubo a pessoa; (b) Utilização para desmonte e venda de peças; (c) Utilização para uso recreativo; e (d) Utilização para venda do veículo em outro Estado.

Com o intuito de estabelecer uma relação entre as motivações e os atributos a serem aplicados neste trabalho, os seguintes dados são utilizados para enriquecer as trajetórias: (i) o número de ocorrências de crimes violentos contra o patrimônio (CVP) na região ao redor do sensor; (ii) o número de ocorrências de roubo de veículos na região ao redor do sensor; (iii) o número de ocorrências de recuperação de veículos roubados nas proximidades do sensor; Por fim, (iv) menor distância, na rede de ruas, do sensor para cada categoria de ponto de interesse avaliada. Neste caso, as categorias são: supermercados, estacionamentos e oficinas mecânicas.

Uma das questões de pesquisa a serem investigadas é averiguar a relevância desses atributos na criação de um modelo preditivo para o próximo sensor de uma trajetória dada como entrada. Nas próximas subseções, são apresentadas as definições de alguns atributos espaciais utilizados na análise de crimes.

3.1.1 Pontos de Interesse

Pontos de Interesse são pontos georreferenciados que representam locais como lojas, supermercados, parques, entre outros. Além das coordenadas de GPS, outras informações como categoria, popularidade e avaliações podem estar disponíveis. POIs podem ser utilizados de diversas formas na análise espacial de crimes. Huang *et al.* (2018), por exemplo, propõe o *DeepCrime*, um *framework* que captura ocorrências criminais relevantes através de diversos períodos de tempo. Os autores analisam a correlação entre categorias de pontos de interesse (POIs), anomalias no cenário urbano e dependências temporais com a ocorrência de crime. O *DeepCrime* utiliza a distribuição das categorias de pontos de interesse nas regiões de uma cidade e verifica a existência de uma correlação positiva entre regiões com alta densidade de pontos de interesse ligados ao entretenimento como bares e clubes, com regiões com alta densidade de roubos.

Oróstica e Poblete (2019) apresentam uma abordagem que utiliza a proximidade entre categorias de pontos de interesse com locais de ocorrência de roubo de veículos. No modelo, cada ocorrência de roubo de veículo é representado como um vetor em função das distâncias para categorias de pontos de interesse. Cada posição no vetor representa a frequência de uma categoria de POI dentro de um raio de 200 metros.

Neste trabalho, os pontos de interesse são utilizados para anotar as trajetórias com informações sobre a menor distância entre as categorias de POIs apontadas como relacionadas ao crime de roubo de veículo para os sensores que registram a passagem de veículos roubados. Assim, a menor distância entre os sensores das trajetórias dos veículos roubados e as categorias de pontos de interesse analisadas é definida como:

Definição 3.1.1 (Distância para Pontos de Interesse - POIs) *Dado uma Trajetória de Sensores Externos (EST) $es_traj_m = (o_1, o_2, \dots, o_j)$ de modo que o_i é uma tupla $o = (m, s, t)$ representado respectivamente o objeto em movimento m , o sensor s e o timestamp t . A menor distância \bar{d}_k é a distância do sensor s para o POI da categoria k mais próximo de s . Dessa forma, para cada sensor s registrado na trajetória será produzido um vetor adicionado a trajetória semântica:*

$$S_{\langle lat, lon \rangle} = [\bar{d}_{categoria_1}, \dots, \bar{d}_{categoria_k}]$$

3.1.2 Hotspots de Crimes e Kernel Density Estimation - KDE

O mapeamento de *hotspots* é uma abordagem muito utilizada na análise e previsão de crimes. Tal abordagem é centrada na informação espacial e se baseia na hipótese de que os locais de eventos passados contribuem de forma efetiva na predição de eventos futuros. Hart e Zandbergen (2014) definem *hotspots* de crime como áreas de concentração de ocorrência de delitos que demonstram um padrão não aleatório no espaço e no tempo. Para Eck *et al.* (2005), *hotspots* são áreas de concentração de crime, cuja a identificação parte de eventos individuais e permite a descoberta de uma série de violações relacionadas. A análise de *hotspots* de infrações criminais também está presente nos trabalhos de Yu *et al.* (2014), Wang *et al.* (2016) e Oróstica e Poblete (2019).

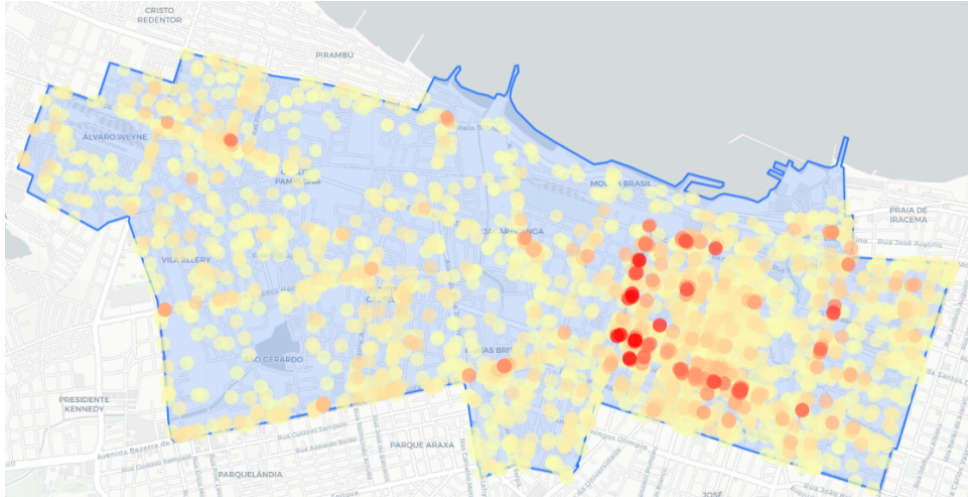
Por outro lado, o *kernel density estimation* (KDE) além de descrever eventos conhecidos em termos de padrões espaciais não aleatórios visualizados como várias formas geométricas, também produz um mapa de risco em perspectiva, que é interpolado de locais específicos de incidentes de crime para uma área de estudo definida (HART; ZANDBERGEN, 2014). Ou seja, o KDE generaliza ou "suaviza" pontos de dados discretos para que uma área de superfície contínua seja produzida e visualizada (BAILEY; GATRELL, 1995; HART; ZANDBERGEN, 2014).

O KDE é uma das técnicas de análise espacial de dados criminais mais utilizada. Isso ocorre devido à aparência estética dos mapas gerados, o que contribui para a interpretabilidade e pela flexibilidade dos parâmetros utilizados. Basicamente, para geração de mapas de densidade

de kernel, os parâmetros utilizados são o raio e a densidade ou volume de ocorrência da área desejada (CHAINEY *et al.*, 2008).

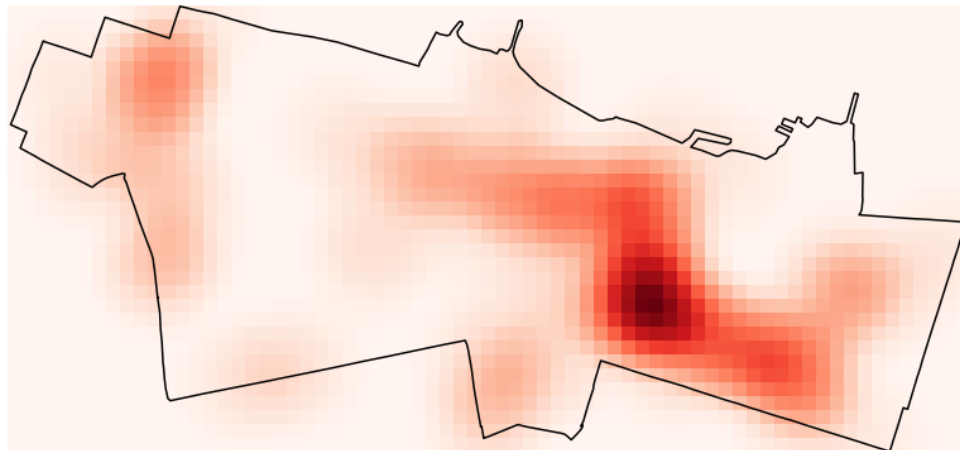
As Figuras 6 e 7 demonstram algumas técnicas de mapeamento baseada em *hotspots*.

Figura 6 – Técnica de mapeamento de pontos



Fonte: (RÊGO *et al.*, 2020)

Figura 7 – Técnica de mapeamento de KDE



Fonte: Elaborado pelo autor.

3.2 Modelos de Aprendizado de Máquina

Aprendizado de máquina pode ser definido como um subcampo da Inteligência Artificial que envolve algoritmos com auto-aprendizagem que derivam conhecimento a partir dos dados para realizar previsões oferecendo uma alternativa mais eficiente para capturar conhecimento dos dados e auxiliar no processo de tomada de decisão (RASCHKA; MIRJALILI, 2017).

De modo geral o Aprendizado de Máquina pode ser dividido em três tipos: Apre-

dizado Supervisionado, Aprendizado Não-Supervisionado e Aprendizado por Reforço. O Aprendizado Supervisionado lida com problemas com as seguintes características: Dados Rotulados, *feedback* direto e objetivam a predição do resultado ou do futuro. O Aprendizado Não-Supervisionado trata de dados não rotulados, não apresenta um *feedback* direto e busca encontrar estruturas ocultas nos dados. E por fim, o Aprendizado por Reforço trata de processos de decisão baseados em um sistema de recompensa e aprendizado de uma série de ações.

O problema investigado nesta pesquisa é uma tarefa de aprendizado supervisionado, mais especificamente um problema de classificação, que é uma subcategoria de aprendizado supervisionado que tem por objetivo prever os rótulos de classe categóricas de novas instâncias, com base em observações anteriores. A seguir são discutidos os algoritmos de Aprendizado de Máquina utilizados nesta pesquisa.

3.2.1 *K-Nearest Neighbors Classifier - KNN*

O *K-Nearest Neighbors* (KNN) é considerado um algoritmo de *lazy learner* ou aprendizado lento em virtude de no processo de treinamento não aprender uma função que descreve os dados, em vez disso memoriza o conjunto de dados (RASCHKA; MIRJALILI, 2017). Devido a essa característica, o KNN pertence a uma subcategoria de modelos não paramétricos, caracterizados por apresentarem um processo de aprendizagem baseada nas instâncias.

O funcionamento do KNN é bem simples e representado pelas seguintes ações: 1) definição dos parâmetros k e métrica de distância; 2) descoberta dos k -vizinhos mais próximos da amostra a ser classificada; 3) atribuição do rótulo da classe por maioria dos votos.

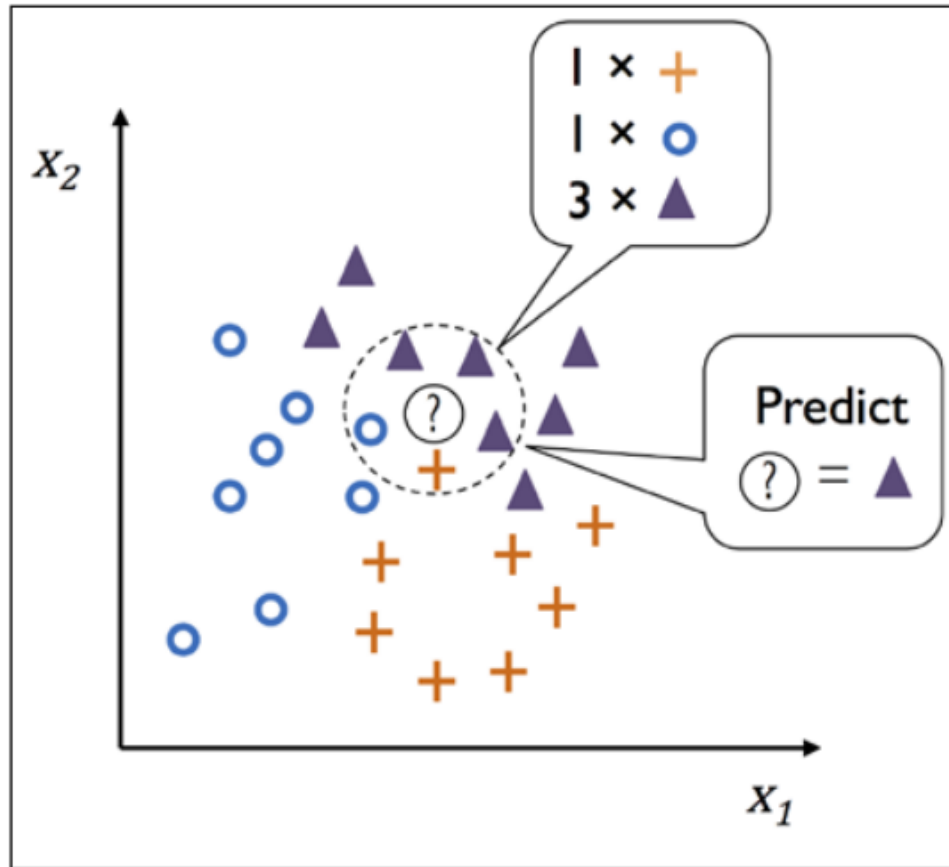
O parâmetro k deste algoritmo é utilizado para determinar o número de vizinhos a serem considerados na votação para determinar a classe de uma nova amostra. A Figura 8 demonstra o funcionamento deste algoritmo.

A utilização do KNN para predição da próxima localização é discutida em (ZENG *et al.*, 2019), onde o algoritmo foi utilizado para verificar se o problema pode ser bem avaliado baseado apenas na semelhança das amostras.

3.2.2 *Random Forest*

Random Forest proposto por Breiman (2001) adiciona uma camada de aleatoriedade ao *bagging*, *Bootstrap Aggregating*. Ele usa a seleção aleatória de *features* em cada nó para dividir o conjunto de *features*.

Figura 8 – Técnica de mapeamento de pontos



Fonte: (RASCHKA; MIRJALILI, 2017)

O *Random Forest* é uma combinação de preditores de árvore, como cada árvore depende dos valores de um vetor aleatório amostrado de forma independente e com a mesma distribuição para todas as árvores, o erro de generalização das árvores converge para um limite conforme o número de árvores na floresta aumenta. O erro de generalização de uma floresta depende da força das árvores individuais na floresta e na correlação entre eles.

A utilização de uma seleção aleatória de *features* para dividir cada nó resulta em baixas taxas de erro e mais robustez em relação ao ruído. Estimativas internas monitoram erros, força, e correlação, e estes são usados para mostrar a resposta aumentando o número de *features* usadas na divisão. Estimativas internas também são usadas para medir a importância das *features*.

3.2.3 Gradient Boost Machine

Gradient Boost Machine (GBM) (FRIEDMAN, 2001), também conhecido como *gradient tree boosting* ou *gradient boosted regression tree*, constrói modelos de regressão aditivos ajustando sequencialmente uma função parametrizada simples (aprendizado básico) aos pseudo-residuais atuais por mínimos quadrados em cada iteração. Os pseudo-residuais são o gradiente da

função *loss* sendo minimizado para os valores do modelo em cada ponto de dados de treinamento avaliado na etapa atual. GBM é uma técnica de aprendizado de máquina que produz resultados competitivos de última geração e altamente robustos em uma ampla gama de aplicações de regressão e classificação. *Boosting* é um problema de otimização numérica em que o objetivo é minimizar a perda do modelo adicionando classificadores fracos usando um procedimento semelhante ao gradiente descendente.

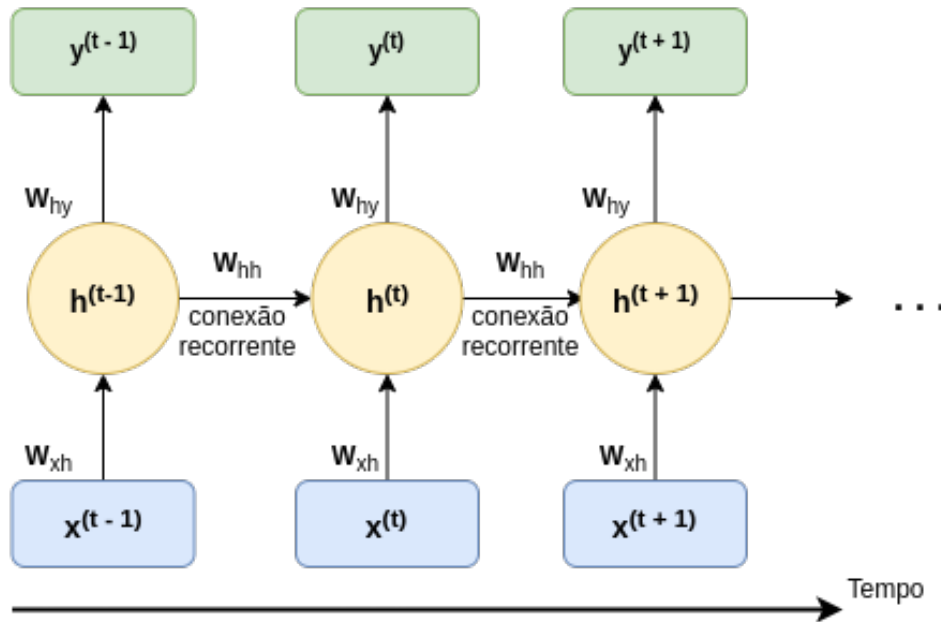
A implementação original desse algoritmo foi chamada de *Multiple Additive Regression Trees (MART)*. A precisão da aproximação e a velocidade de execução do aumento do gradiente podem ser substancialmente melhoradas incorporando a randomização ao procedimento (FRIEDMAN, 2002). Uma subamostra dos dados de treinamento é retirada aleatoriamente do conjunto de dados de treinamento em cada iteração. Essa subamostra selecionada aleatoriamente é então usada no lugar da amostra completa para ajustar o classificador base e calcular a atualização do modelo para a iteração atual. Esta abordagem aleatória é chamada *Stochastic Gradient Boosting*, e é frequentemente usada para aumentar a robustez contra o excesso de capacidade do classificador base.

3.2.4 Redes Neurais Recorrentes - RNN

Redes Neurais Recorrentes (RNN) são umas das abordagens mais eficazes para a predição de dados sequencias. Isso se deve ao fato de proverem o aprendizado através da relação entre valores consecutivos. A RNN processa uma sequência iterando através dos elementos e mantendo na saída atual informações dos dados vistos anteriormente, através de um *loop* interno, fazendo com que os dados pontuais circulem na rede (CHOLLET, 2018). A Figura 9 demonstra o funcionamento de uma RNN, percebe-se que a cada etapa de tempo t , o neurônio recebe a entrada (t) e saída da etapa anterior ($t - 1$). Dessa forma, o fluxo de informações adjacentes permite que a rede tenha uma memória dos valores passados. Para o processamento da saída, cada neurônio da RNN recebe dois conjuntos de pesos: um para a entrada, $x^{(t)}$ e outro para a saída da etapa anterior, $y^{(t-1)}$ (GÉRON, 2019). Esses conjuntos estão representados na Figura 9 como os vetores W_{xh} e W_{hy} , respectivamente. Na transição entre camadas ocultas h , a matriz recorrente W_{hh} é repassada para a camada seguinte.

A computação da ativação da unidades ocultas da RNN é realizado pela equa-

Figura 9 – Representação de uma rede neural recorrente - RNN



Fonte: Adaptada de (GÉRON, 2019)

ção (3.1):

$$h^{(t)} = \phi_h(W_{xh}x^{(t)} + W_{hh}h^{(t-1)} + b_h) \quad (3.1)$$

Em que:

- ϕ_h é uma função de ativação, como por exemplo a função ReLU ¹.
- W_{xh} é a matriz de peso entre a entrada $x^{(t)}$ e a camada oculta h .
- W_{hh} é a matriz de peso associada à aresta recorrente.
- $h^{(t-1)}$ é estado oculto da camada anterior.
- b_h é a unidade de polarização ou termo de bias.

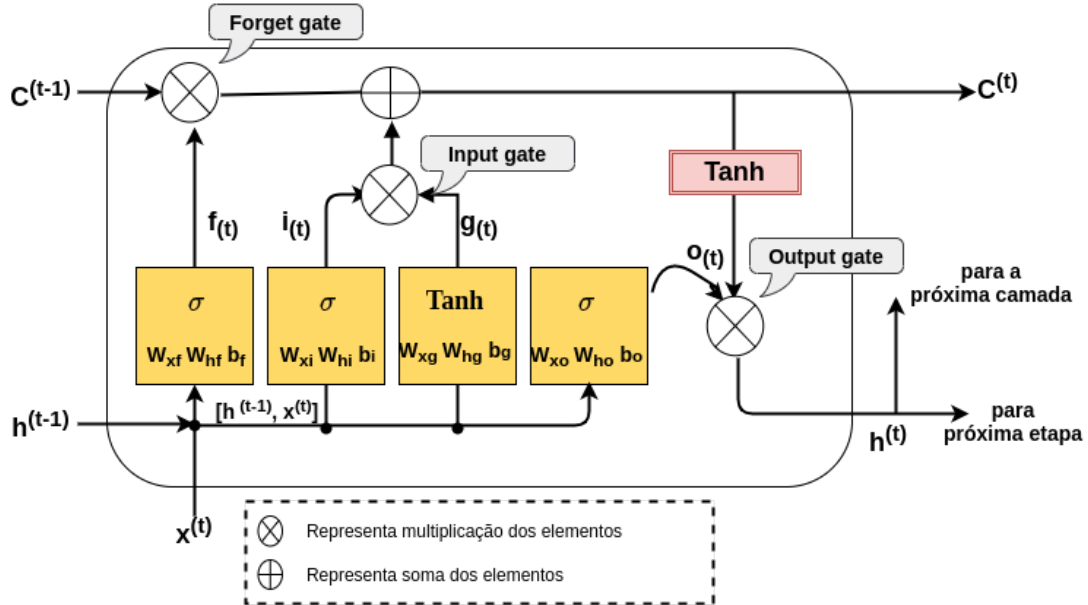
3.2.4.1 Long Short-Term Memory (LSTM)

LSTM é um tipo especial de RNN projetado com o objetivo de resolver o problema de dependências de longo prazo (*vanishing gradient*) apresentado em uma RNN padrão. Esse tipo de rede neural vem sendo aplicado em diversas abordagens para predição da mobilidade (YAO *et al.*, 2017; KONG; WU, 2018; WU *et al.*, 2017). Diferentemente da RNN padrão que apresenta uma camada oculta simples, a LSTM possui um módulo de repetição mais sofisticado.

¹ A função ReLU é definida como: $f = \max(0, x)$

O módulo LSTM possui portões para controlar os estados das células anteriores muito antigas (WU *et al.*, 2017). A Figura 10 apresenta a representação de uma célula LSTM.

Figura 10 – Representação de uma rede neural LSTM



Fonte: Adaptada de (GÉRON, 2019)

A LSTM utiliza dois portões para controlar o estado da célula. O *forget gate* é utilizado para determinar quais informações do estado anterior $C^{(t-1)}$ serão mantidas na etapa atual $C^{(t)}$. Por outro lado, o *input gate* decide a quantidade de informação da entrada atual $x^{(t)}$ será mantida no estado de longa duração $C^{(t)}$. Já o *output gate* é utilizado para controlar a informação do estado de longo prazo $C^{(t)}$ que será repassada para o estado de curto prazo $h^{(t)}$. A seguir, a equação (3.2) demonstra as operações realizadas na célula LSTM como descritas em Géron (2019):

$$\begin{aligned}
 i_{(t)} &= \sigma((W_{xi}^T \cdot x^{(t)} + W_{hi}^T \cdot h^{(t-1)} + b_i)) \\
 f_{(t)} &= \sigma((W_{xf}^T \cdot x^{(t)} + W_{hf}^T \cdot h^{(t-1)} + b_f)) \\
 o_{(t)} &= \sigma((W_{xo}^T \cdot x^{(t)} + W_{ho}^T \cdot h^{(t-1)} + b_o)) \\
 g_{(t)} &= \tanh((W_{xg}^T \cdot x^{(t)} + W_{hg}^T \cdot h^{(t-1)} + b_g)) \\
 c^{(t)} &= f_{(t)} \otimes c^{(t-1)} + i_{(t)} \otimes sg_{(t)} \\
 y^{(t)} = h^t &= o_{(t)} \otimes \tanh(c^{(t)})
 \end{aligned} \tag{3.2}$$

Em que:

- W_{xi}, W_{xf}, W_{xo} e W_{xg} são matrizes de peso para cada uma das camadas conectadas com o vetor de entrada $x^{(t)}$.

- W_{hi}, W_{hf}, W_{ho} e W_{hg} são matrizes de peso para cada uma das camadas conectadas com o estado da etapa anterior $h^{(t-1)}$.
- b_i, b_f, b_o e b_g são os termos de *bias*.

O modelo de predição proposto neste trabalho investiga o uso da LSTM.

3.2.4.2 LSTM Bidirecional (BiLSTM)

Em casos em que todas as etapas de tempo da sequência estão disponíveis, uma alternativa é a utilização de uma rede LSTM Bidirecional - (BiLSTM). De modo geral, com BiLSTMs uma rede neural recorrente é treinada duas vezes. Primeiramente, é realizado o treinamento na sequência como os dados aparecem na entrada, ou seja, como uma camada LSTM *forward*. Em seguida, é fornecida uma cópia da sequência de entrada invertida no tempo, como uma camada *backward*. Dessa forma, um contexto adicional é fornecido à rede, o que pode melhorar sua precisão através do aprendizado com informações passadas e futuras.

Na BiLSTM, dado n o número de etapas da sequência de entrada, a LSTM *forward* \vec{f} realiza a leitura da entrada como está ordenada (de x_1 para x_n), para calcular a sequência de estados ocultos *forward* da rede $(\vec{h}_1, \dots, \vec{h}_n)$. De outro modo, a LSTM *backward* \overleftarrow{f} lê a sequência na ordem inversa (de x_n para x_1), para calcular a sequência dos estados ocultos *backward* da rede $(\overleftarrow{h}_1, \dots, \overleftarrow{h}_n)$.

Para computar o estado final h_j , para cada entrada x_j é realizada a concatenação do estado oculto *forward* \vec{h}_j com estado oculto *backward* \overleftarrow{h}_j , como demonstra a Equação 3.3.

$$h_j = [\vec{h}_j; \overleftarrow{h}_j] \quad (3.3)$$

A seguir, são apresentados os trabalhos relacionados a este.

4 TRABALHOS RELACIONADOS

Objetos em movimento estão presentes em toda parte e relacionados a diversos aspectos do cotidiano das pessoas. A análise de mobilidade tem recebido relevante atenção científica, o que fez surgir inúmeras abordagens para a predição da próxima localização. Essa atenção por parte dos pesquisadores muito se justifica pelo aprimoramento de sistemas de comunicação sem fio e pelo aumento da procura por serviços baseados em localização por parte dos usuários. São apresentados neste Capítulo alguns dos mais recentes trabalhos que objetivam a predição do próximo local em trajetórias externas ou que compartilham a características de serem esparsas, além de trabalhos que propõem o uso de dados semânticos. Para melhor organização, os trabalhos foram divididos em duas Seções: Trajetórias de Sensores Externos ou Esparsas e Trajetórias com Dados Semânticos.

4.1 Trajetórias de Sensores Externos ou Esparsas

Cruz *et al.* (2019) apresentam o *EST Prediction*, um estudo profundo sobre os desafios da análise de trajetórias derivadas de sensores externos como o problema da incompletude e escassez de dados. Os autores analisaram uma base de dados real com trajetórias de uma rede de 272 sensores e um total de 22.338.916 observações. O objetivo geral do estudo é propor um modelo para predição do próximo sensor a registrar a passagem de um determinado objeto em movimento. Para lidar com os problemas de incompletude e escassez os autores demonstram diversos métodos de imputação de dados e realizam comparações entre eles. A abordagem utilizada para a construção do modelo é baseada em Recurrent Neural Networks (RNN). A principal diferença do *EST Prediction* para proposta desta pesquisa é que aquele não recebe informações semânticas relacionados às trajetórias, além de não considerar restrições de roubo ou furto dos veículos.

Liu *et al.* (2019) apresentam o *Travel Time Difference Model (TTDM)*, um modelo para predição da próxima localização que utiliza o tempo decorrido entre os segmentos da trajetória para melhorar o desempenho da predição. O TTDM explora o conjunto de trajetórias históricas para construção de um grafo ponderado, em que os locais contidos nas trajetórias são considerados como nós e as transições entre os locais como arestas. Nessa abordagem, os autores calculam a média do tempo de viagem para cada segmento em diversos *slots* de tempo e utilizam essa informação como peso das arestas, em seguida determinam um conjunto

de locais candidatos a serem a próxima localização de uma trajetória de busca. Finalmente, realizam a predição do próximo local com base no menor tempo de viagem entre os locais candidatos. Na fase de predição, o TTDM realiza a integração entre a estratégia baseada no tempo de viagem com o Modelo de Markov apresentado em Chen *et al.* (2015), tal integração é realizada através de interpolação linear. Durante os experimentos foram utilizados dois conjuntos de dados, o primeiro com trajetórias obtidas por um sistema de vigilância de tráfego que registra a passagem de veículos (VRP) e um segundo que obtém a localização de uma frota de táxis. No conjunto de VRP, os locais das trajetórias são os locais de implantação dos dispositivos do sistema de vigilância, enquanto no conjunto de trajetórias de táxis o espaço geográfico da cidade é discretizado em células do mesmo tamanho. Apesar de utilizar trajetórias de sensores externos em suas análises e enriquecer a predição com a diferença temporal das transições entre os locais, o TTDM difere da proposta deste trabalho por não considerar outras informações semânticas como pontos de interesse e tipo de via dos sensores, além disso não considera a presença de restrição de roubo nos veículos do estudo.

O DeepMove, proposto em Feng *et al.* (2018), é um *framework multi-modal* para a predição da próxima localização, em que autores propõe prever a mobilidade humana para trajetórias grandes e esparsas. O DeepMove apresenta dois módulos: 1) O módulo recorrente que busca capturar dependências complexas contidas na trajetória mais recente. 2) O módulo de atenção histórica desenvolvido para capturar a regularidade e a periodicidade do movimento em longos registros históricos. Na etapa de predição os dois módulos são concatenados em um único vetor e enviado como entrada para camada de saída de uma RNN para predição do local na próxima janela de tempo. Os autores analisam três *datasets* reais, sendo que um deles apresenta trajetórias esparsas de check-ins, enquanto os outros dois apresentam trajetórias densas fornecidas por aplicações móveis e por registros de chamadas de celulares. Uma das contribuições da proposta é auxiliar na interpretabilidade de quais atividades históricas são enfatizadas na mobilidade futura. Diferentemente do DeepMove, este trabalho lida com trajetórias de sensores externos e considera informações semânticas destas trajetórias. A inclusão de dados semânticos são apontados pelos autores do DeepMove como trabalhos futuros para entender a motivação da movimentação dos usuários.

Com o objetivo de sanar deficiências nas abordagens tradicionais de predição da próxima localização para trajetórias baseadas em sequência espaço-temporal, Zeng *et al.* (2019) propõem o modelo chamado SSTLP. A principal característica da proposta do SSTLP é a cons-

trução de um modelo que combina a probabilidade de transição entre locais com a probabilidade baseada na distância. A primeira é calculada a partir das transições de um local para outro. A segunda é determinada através de distribuição normal e similaridade cosseno da distância entre os locais. Finalmente, a probabilidade de transição e a probabilidade baseada na distância são ajustadas para determinar o próximo local. A solução proposta no SSTLP é focada na descoberta do padrão do usuário ao qual a trajetória dada como entrada pertence, diferentemente da proposta deste trabalho, não se pretende investigar padrões individuais de trajetórias. Além disso, o SSTLP não trata do enriquecimento semântico das trajetórias, abordando somente as características espaço-temporais e a similaridade entre diversos usuários.

Devido a promissora performance demonstrada pela aplicação de Redes Neurais Recorrentes - RNN, Liu *et al.* (2016) apresentam uma nova abordagem chamada Spatial Temporal Recurrent Neural Networks (ST-RNN). A solução proposta é capaz de modelar os contextos de tempo e espaço em camadas separadas com estruturas específicas para diferentes intervalos de tempo e para diferentes distâncias geográficas. A arquitetura é composta por uma matriz de transição temporal e uma matriz de distâncias geográficas utilizadas como entrada para a rede neural recorrente. Na etapa de predição o resultado é obtido através do produto interno das representação do usuário, indicando que um determinado usuário u tende a ir a um local v no tempo t . A proposta desse trabalho além de focar nas informações espaço-temporais como o ST-RNN, propõe a utilização de uma RNN que recebe informações extras.

Kong e Wu (2018), com o objetivo de investigar o problema da predição do movimento de usuários nos próximos minutos ou horas, propõem um modelo de memória espaço-temporal de longo prazo (HST_LSTM). O HST_LSTM modela o contexto da visita através da identificação de áreas de interesse (AOI) que representam zonas funcionais do espaço geográfico, como áreas comerciais e centro de lojas, ou seja, são áreas que concentram vários pontos individuais de uma determinada categoria. Além disso, a arquitetura proposta realiza uma modificação nas operações internas da LSTM, adicionando fatores espaço-temporais nos três portões da rede: *input gate*, *forget gate* e *output gate*. A abordagem reforça a importância da informação espaço-temporal na predição dos pontos de interesse, porém diferentemente do proposto nesse trabalho, não considera múltiplas informações semânticas anotadas nas trajetórias.

4.2 Trajetórias com Dados Semânticos

O problema da predição da próxima localização em redes de sensores externos também é abordado em Fan *et al.* (2018). Os autores propõem um modelo baseado em aprendizado profundo com o objetivo de prever a próxima localização de um veículo a partir de suas trajetórias históricas e outras informações contextuais. Tais trajetórias são obtidas por dispositivos de vigilância de tráfego implantados ao longo da rede de ruas de uma cidade. Na abordagem apresentada, informações de contexto como tipo de veículo, direção do caminho e fluxo do tráfego, fornecidas pelos dispositivos, são incorporados com dados de climáticos e as informações espaço-temporais das trajetórias. Os resultados demonstraram que a inclusão de informações contextuais aprimora a predição. Este trabalho difere da abordagem apresentada por Fan *et al.* (2018) pelo fato de analisar trajetórias de veículos com restrição de roubo ou furto. Além disso, as informações semânticas utilizadas no enriquecimento das trajetórias em nosso trabalho são pontos de interesse e dados relacionados com crimes no espaço urbano.

Um dos trabalhos que propõe o enriquecimento semântico da trajetória para a predição da próxima localização é o *semantics-enriched recurrent model (SERM)* (YAO *et al.*, 2017). O SERM é um modelo espaço-temporal que aprende com *embeddings* de múltiplos fatores (usuário, localização, tempo, palavra-chave) que são concatenados através de um *framework* unificado. Os autores analisam dois *datasets*, o primeiro com dados de *check-ins* na cidade de Nova York e o segundo com dados de *tweets* da cidade de Los Angeles. Para a informação da localização é realizada a discretização do espaço geográfico das cidades em células de 500m X 500m. A informação semântica é um *bag-of-words* cujas palavras-chave são mapeadas para um vetor *one-hot encoding*. Em seguida é realizada a concatenação de todos os *embeddings*, após a concatenação o resultado é dado como entrada para uma rede neural recorrente - RNN. Por fim, um vetor com as preferências dos usuários é acrescentado à saída da RNN para a predição da próxima parada do usuário. A proposta desse trabalho difere do SERM pelo fato de utilizar trajetórias obtidas por sensores externos aos objetos em movimento analisados, portanto mais esparsas e incompletas. Outra diferença é que a preferência dos usuários não será utilizada, pois exige um número expressivo de trajetórias por usuário, o que não está disponível no caso deste trabalho. Além disso, o enriquecimento semântico proposto utiliza outras *features* como distância para pontos de interesse e para ocorrências de crimes.

Ghosh e Ghosh (2018) também abordam o problema de predição da próxima localização com dados de trajetórias esparsas com a agregação de múltiplos fatores. Na proposta são

abordados três diferentes cenários e conseqüentemente três problemas distintos: No primeiro cenário, a localização atual do usuário, bem como sua trajetória recente são conhecidas. Neste caso, a tarefa é a predição do próximo local. No segundo cenário a informação está incompleta, diante disso, a tarefa é atualizar o estado do sistema refinando a probabilidade de transição. No último quadro a localização atual não está disponível, em tal caso, a tarefa é prever a localização atual com base nos trajetórias históricas. O enriquecimento semântico é realizado com a proximidade entre os locais das trajetórias e pontos de interesse, informações da estrutura da rede de ruas e indexação dos locais por meio de uma estrutura baseada em *hash*. Além disso, o modelo ainda utiliza outras informações de contexto como dados do sensor acelerômetro, atividades dos usuários nos pontos de interesse e acessos em redes sem fio. Na experimentação, a técnica dos autores simula a escassez dos dados realizando progressivas reduções de 10%, 20% e 30%. A proposta de Ghosh e Ghosh (2018) reforça que o enriquecimento semântico contribui para a predição da próxima localização, inclusive citando como trabalhos futuros a realização de testes em redes neurais recorrentes, como proposto neste trabalho.

Finalmente, em Wu *et al.* (2017) é proposto o modelo STS-LSTM. A proposta busca resolver o problema da perda de informação semântica, presentes na rede de ruas, que ocorrem em abordagens tradicionais. O STS-LSTM é dividido em duas etapas principais. O passo inicial é a extração de *features* espaço-temporais para converter as trajetórias em pontos fixos e discretos ao longo da rede de ruas. Em seguida é construído um modelo baseado em LSTM para predição do próximo ponto de parada. As trajetórias utilizadas no STS-LSTM são mais densas que as utilizadas na proposta desse trabalho, outra diferença entre as abordagens é que o presente trabalhos utiliza diversas informações semânticas para anotar as trajetórias.

4.3 Resumo dos Trabalhos Relacionados

O Quadro 1, apresentado a seguir, explica resumidamente a relação entre os trabalhos relacionados e o que está sendo proposto neste trabalho.

Quadro 1 – Comparação entre os trabalhos relacionados e o proposto

| Artigo | Tipo Modelo | Tipo de Dado | Particularidades |
|---------------------------|------------------------------------|--|---|
| Cruz <i>et al.</i> (2019) | Classificação/Redes Neurais | Dados de Sensores | RNN Espaço-Temporal |
| Fan <i>et al.</i> (2018) | Classificação/Redes Neurais | Dados de Sensores | RNN com dados semânticos ¹ |
| Liu <i>et al.</i> (2019) | Modelo Probabilístico | Dados de Sensores e Trajetórias Brutas | Cadeias de Markov |
| Feng <i>et al.</i> (2018) | Classificação/Redes Neurais | Check-ins e redes sociais | RNN com mecanismo de atenção |
| Yao <i>et al.</i> (2017) | Classificação/Redes Neurais | Check-ins e redes sociais | RNN com dados semânticos ² |
| Zeng <i>et al.</i> (2019) | Modelo Probabilístico | Check-ins | Baseado na Probabilidade de transição dos locais |
| Liu <i>et al.</i> (2016) | Classificação/Redes Neurais | Check-ins | RNN Espaço-Temporal |
| Ghosh e Ghosh (2018) | Modelo Probabilístico | Redes sociais | Cadeias de Markov com dados semânticos ³ |
| Kong e Wu (2018) | Classificação/Redes Neurais | Trajetórias brutas | RNN Espaço-Temporal |
| Wu <i>et al.</i> (2017) | Classificação/Redes Neurais | Trajetórias brutas | RNN com dados semânticos ⁴ |
| Proposta | Classificação/Redes Neurais | Dados de Sensores | Classificadores/RNN com dados semânticos⁵ |

Fonte: Elaborado pelo autor.

Descrição dos Dados Semânticos Utilizados nos Trabalhos Relacionados.

- ¹ Tipo de Veículo (pequeno, médio, grande), Direção do Movimento (sul, leste, oeste, sudeste, nordeste, sudoeste e noroeste), Condições de Tráfego, Condições Climáticas.
- ² Texto, *bag of keywords*, de redes sociais.
- ³ Categorias de POIs que indicam atividades realizadas por usuários (universidade, local de trabalho, edifícios residenciais, etc.).
- ⁴ *Tags* semânticas que indicam o tipo da via na rede de ruas.
- ⁵ Categorias de POIs (Supermercado, Oficinas mecânicas e Estacionamentos) e Dados georreferenciados de ocorrências criminais. Trajetórias de Carros Roubados.

5 MATERIAIS E MÉTODOS

5.1 Introdução

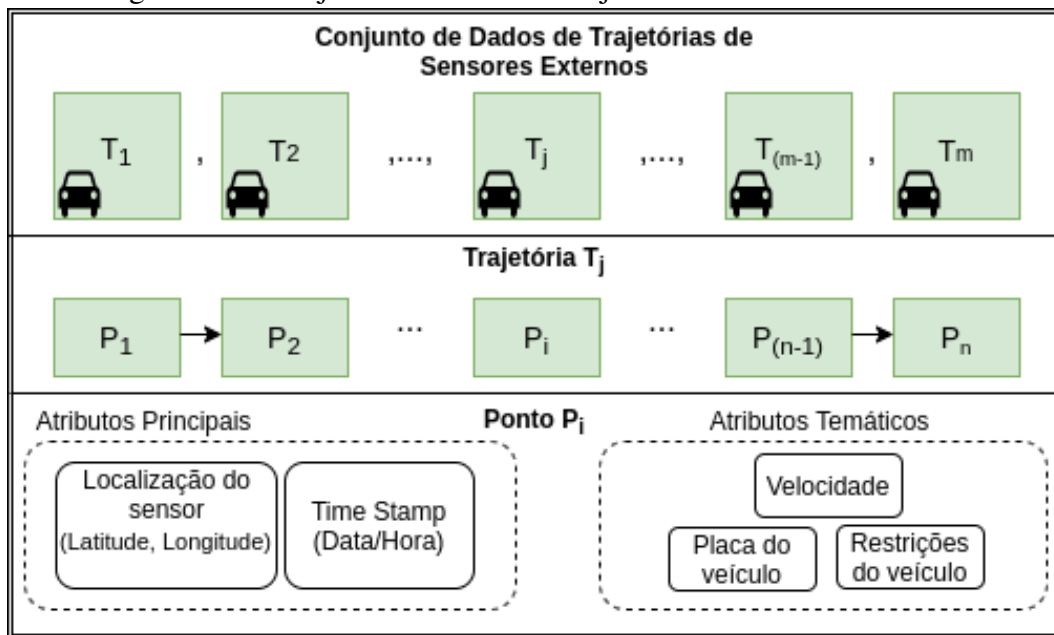
Neste Capítulo são apresentados todos os procedimentos realizados durante a pesquisa. Este estudo pode ser caracterizado de forma geral como um estudo transversal, com abordagem quantitativa e objetivo exploratório. Na investigação, são utilizados dados primários reunidos por meio de *Application Programming Interface* (API)s disponíveis para coleta de dados georreferenciados, além de dados secundários fornecidos pela Secretaria de Segurança Pública do Estado do Ceará.

5.2 Conjunto de Dados de Trajetórias

O cenário da coleta de dados é a cidade de Fortaleza, capital do Estado do Ceará, localizada no nordeste do Brasil. Os dados de trajetórias são observações de passagem de veículos nos meses de janeiro a junho de 2019, obtidas por 489 sensores instalados às margens da rede de ruas da cidade. Tais equipamentos são utilizados em uma aplicação real de monitoramento de tráfego que diariamente registra em torno de 2 milhões de observações. No entanto, este estudo analisa apenas as observações de veículos que apresentam restrições de roubo ou furto, o que potencializa os desafios da análise de EST já mencionados como a esparsidade e incompletude.

Figura 11 apresenta os componentes do conjunto de dados de trajetórias de sensores externos utilizado neste estudo. O conjunto contém 31.280 observações de 3.611 veículos distintos. Além dos atributos básicos para a formação das trajetórias, estão presentes alguns atributos temáticos como a placa de identificação do veículo, a velocidade e um atributo que indica eventuais restrições como, por exemplo, restrição de roubo ou furto. A cidade de Fortaleza foi escolhida como cenário deste estudo devido a realização do projeto SINESP BIG DATA E INTELIGÊNCIA ARTIFICIAL PARA SEGURANÇA PÚBLICA, desenvolvido pela Secretaria Nacional de Segurança Pública do Estado do Ceará em parceria com a Universidade Federal do Ceará - UFC. Por restrições de confidencialidade, os dados não estão disponíveis para o público geral.

Figura 11 – Conjunto de Dados de Trajetórias de Sensores Externos.



Fonte: adaptado de (KAMW, 2019)

5.3 Conjunto de Dados Semânticos

Os dados semânticos estão divididos em duas categorias: (I) Dados Criminais e (II) Pontos de Interesse.

5.3.1 Dados Criminais

Os dados de ocorrência de crimes, fornecidos pela Secretaria de Segurança Pública e Defesa Social do Estado do Ceará através do projeto SINESP BIG DATA E INTELIGÊNCIA ARTIFICIAL, estão divididos em três categorias: (I) Crime Violento Contra o Patrimônio (CVP)s, roubo ou furto de veículo automotor e recuperação de veículos roubados. Explicitamente, os registros de ocorrências contém o seguintes atributos: 1) Tipo de Evento: descreve a categoria de crime; 2) Cidade: informa a cidade de registro do evento; 3) Data/Hora: *timestamp* do registro da ocorrência; 4) Latitude e 5) Longitude: descrevem a localização do evento. Tais conjuntos são detalhados no Quadro 2.

Quadro 2 – Detalhes do conjunto de dados criminais

| Categoria | Número de Instâncias |
|--------------------------------|-----------------------------|
| CVPs | 115.163 |
| Roubo ou Furto de Veículo | 3.993 |
| Recuperação de Veículo Roubado | 47.762 |

Fonte: Elaborado pelo autor.

5.3.2 Pontos de Interesse

O conjunto com os pontos de interesse foi coletado utilizando as APIs *Google Maps* (SVENNERBERG, 2010) e *OpenStreetMap* (OPENSTREETMAP CONTRIBUTORS, 2017). Os atributos presentes no conjunto de pontos de interesse são os seguintes: 1) Nome do Bairro; 2) Nome do Logradouro; 3) Nome do Local: nome do estabelecimento comercial; 4) Latitude e Longitude: descrevem a localização do estabelecimento. O Quadro 3 descreve detalhes do conjunto de dados.

Quadro 3 – Detalhes do conjunto de dados de pontos de interesse

| Categoria | Número de Instâncias |
|--------------------|-----------------------------|
| Oficinas Mecânicas | 4.070 |
| Estacionamentos | 457 |
| Supermercados | 675 |

Fonte: Elaborado pelo autor.

5.4 Procedimentos

Abaixo, são listados os passos realizados para preparação do conjunto de dados enviados para os modelos preditivos de sensores externos.

5.4.1 Pré-processamento

Algumas técnicas de limpeza de dados foram aplicadas com o objetivo de eliminar ruído dos dados ocasionados por problemas na coleta ou armazenamento das trajetórias. A primeira técnica de limpeza utilizada foi o filtro de trajetórias através da informação de geometria *bounding box*, para retirada de pontos de fora do espaço geográfico da cidade de Fortaleza-CE. A segunda técnica foi a calibração, técnica que auxilia na identificação de tratamento de *outliers* e adiciona correção estatística em vez de filtrá-los diretamente (WANG *et al.*, 2021). Neste trabalho, a calibração aplicada utilizou as restrições da topologia da rede de ruas corrigindo pontos que se afastavam dos limites das vias, como proposto em Liu *et al.* (2012).

5.4.2 Engenharia de Features e Enriquecimento Semântico

Devido à continuidade do espaço e tempo, uma trajetória não é dada como entrada para modelos preditivos diretamente. A seguir, são descritos as atividades de engenharia de

features para preparação da entrada para os modelos.

5.4.2.1 *Dado Espacial*

Para o dado espacial, as representações contínuas das coordenadas dos pontos, neste trabalho sensores, que compõem a trajetória são pré-processadas para converter os valores das latitudes e longitudes códigos discretos. A codificação discreta escolhida foi o *Geohash* (NIEMEYER, 2008). O *Geohash* é uma forma de codificar uma localização geográfica em uma sequência de caracteres, sua precisão depende do tamanho da sequência e quanto maior o prefixo compartilhado entre duas localizações, mais próximas elas serão. O tamanho da sequência do *Geohash* utilizado neste trabalho é de 50 caracteres, 25 para latitude e 25 para longitude. Dessa forma, cada sensor é mapeado para um código diferente.

5.4.2.2 *Dado Temporal*

A informação temporal é transformada em intervalos discretos, aqui chamados de *slot* de tempo. As horas do dia são divididas em múltiplos *slots* de mesmo tamanho. O tamanho escolhido neste trabalho foi de 30 minutos. Assim, cada dia contém 48 *slots* identificados de 0 a 47.

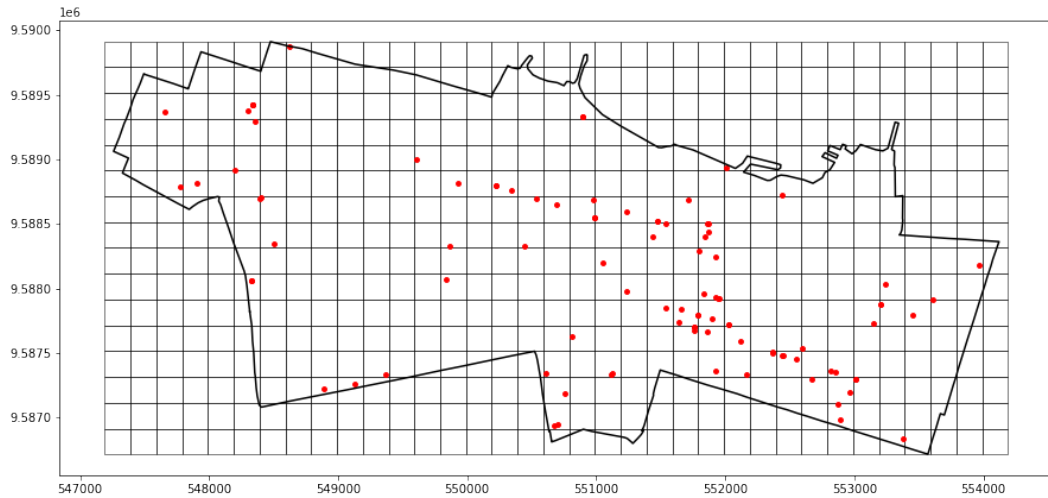
5.4.2.3 *Enriquecimento Semântico com Dados Criminais*

Para anotação dos dados criminais descritos na Seção 5.3.1 nas trajetórias os seguintes procedimentos são realizados:

- a. Construção de uma *grid* uniforme que divide o espaço geográfico que contém a rede de sensores em um conjunto de linhas e colunas uniformemente espaçadas (*grid cells*). O tamanho escolhido para as *grid cells* foi de 500×500 metros;
- b. Identificação da *grid cell* em que cada sensor está localizado;
- c. Contagem de eventos que ocorreram dentro da *grid cell* do sensor, das três categorias de crimes investigadas.

A Figura 12 demonstra o processo de construção de *grid* e identificação dos eventos.

Figura 12 – Identificação de eventos criminais nas *grid cells* dos sensores.

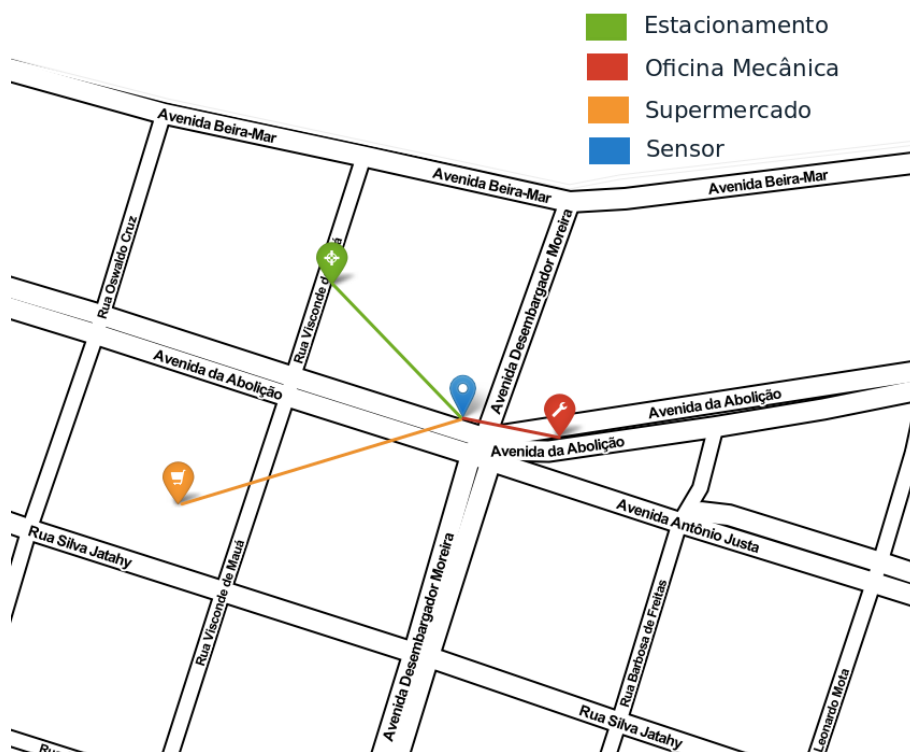


Fonte: Elaborado pelo autor.

5.4.2.4 Enriquecimento Semântico com Dados de Pontos de Interesse

Para a anotação das distâncias para os pontos de interesse, os POIs mais próximos de cada sensor, das três categorias analisadas, são identificados. Em seguida, a distância euclidiana desses pontos para sensor é anotado. Esse processo é representado na Figura 13.

Figura 13 – Anotação das Distâncias para Pontos de Interesse.



Fonte: Elaborado pelo autor.

5.4.3 Imputação de Dados

A imputação de dados de trajetórias de sensores externos é uma maneira de diminuir a incompletude característica deste tipo de trajetória. A incompletude pode ser ocasionada por falha na captura da passagem do objeto em movimento, porém, é importante ressaltar que existe uma incerteza ao analisarmos dados ausentes entre dois sensores consecutivos em uma trajetória, já que o veículo pode simplesmente não ter passado por uma via que contenha sensores (CRUZ *et al.*, 2021).

Assim, com realizado no trabalho de Cruz *et al.* (2019), a imputação é aplicada de acordo com a hipótese que os motoristas geralmente preferem o caminho mais curto. Dessa forma, quando os sensores S_i e S_j aparecem consecutivamente em uma trajetórias, todos os sensores contidos no caminho mais curto entre S_i e S_j são considerados sensores faltantes e incluídos entre esses dois sensores. Apesar de Cruz *et al.* (2019) apresentar duas estratégias de imputação, a imputação completa e a imputação de próximo valor, neste trabalho decidiu-se por utilizar apenas a imputação completa, que realiza a imputação em todas as transições da trajetória, até à localização que será predita.

5.4.4 Segmentação das Trajetórias

A segmentação é o processo de dividir longas trajetórias em várias trajetórias menores baseado em alguns critérios, como critérios espaço-temporais que incluem velocidade, aceleração, localização, entre outros. Ressalta-se que esta segmentação é realizada para preparação do conjunto de dados e não se refere ao processo de divisão do dado em conjuntos de treino e teste para os modelos.

A segmentação aplicada para trajetórias de sensores externos foi proposta por Cruz *et al.* (2019). De modo geral, inicialmente são computadas as trajetórias com as observações de um mesmo veículo para cada dia. Após essa primeira segmentação, uma segunda segmentação é aplicada a partir de dois critérios estatísticos: a média e o desvio padrão do tempo decorrido na transição entre os dois sensores. O processo de segmentação é detalhado no algoritmo 1.

O Algoritmo 1, que apresenta o procedimento de segmentação (*split*) das trajetórias baseado em medidas estatísticas, recebe como entrada as trajetórias diárias dos veículos, o conjunto com as estatísticas de tempo decorrido nas transições de cada par de sensores, o coeficiente M , aplicado à média, e o coeficiente P , aplicado ao desvio padrão do tempo decorrido

Algoritmo 1: Algoritmo Split Estatístico

Entrada: trajetóriasDiárias, estatísticasTempo, M , P
Saída: novoConjuntoDeTrajetórias // novo conjunto de trajetórias após o split estatístico

```

1 novoConjuntoDeTrajetórias := {}; n :=(trajetóriasDiárias.tamanho-1);
2 para  $i := 0$  até  $n-1$  faça
3   trajetória = trajetóriasDiárias[i]
4    $m :=$  (trajetória.tamanho-1)
5   novaTrajetória := {}
6   novaTrajetória.add(trajetória[0]);
7   para  $k := 0$  até  $m-1$  faça
8     sensorInicio = trajetoria.ponto[k]
9     sensorFim = trajetoria.ponto[k+1]
10    tempoDecorrido := trajetotia.ponto[p+1].time - trajetotia.ponto[k].time
11    médiaSegmento := (estatisticaTempo(sensorInicio, sensorFim).média)
12    desvioPadrãoSegmento := (estatisticaTempo(sensorInicio,
13      sensorFim).desvioPadrão)
13    limite := ( $M *$  médiaSegmento) + ( $P *$  desvioPadrãoSegmento)
14    se tempoDecorrido <= limite então
15      novaTrajetória.add(trajetoria.ponto[k+1])
16    senão
17      novoConjuntoDeTrajetórias.add(novaTrajetória)
18      novaTrajetória := {}
19      novaTrajetória.add(trajetória[k+1]);
20      interrompa
21    fim
22  fim
23 fim
24 retorne novoConjuntoDeTrajetórias

```

Fonte: Elaborado pelo autor.

entre dois sensores consecutivos S_k, S_{k+1} . A cada transição entre dois sensores, o *tempo decorrido* (linha 10) é calculado. Em seguida, o *limite* para o seguimento é obtido na linha 13. Para determinar se a trajetória será dividida no segmento atual, verifica-se se o tempo decorrido na transição é menor ou igual ao limite do segmento. Caso o tempo decorrido seja maior que o limite para o segmento, a trajetória é finalizada no sensor S_k , e uma nova trajetória é iniciada a partir do sensor S_{k+1} . Ao fim da execução na linha ??, o algoritmo retorna um novo conjunto de trajetórias sem a presença de comportamentos incomuns para cada segmento do sensor S_k para o sensor S_{k+1} .

Para os experimentos apresentados no próximo Capítulo, $M = 1$ e $P = 2$ foram utilizados, como proposto por Cruz *et al.* (2019), onde os autores definem formalmente o *split* estatístico da seguinte forma:

Seja $\mu_{S_k, S_{k+1}}$ a média e $\sigma_{S_k, S_{k+1}}$ o desvio padrão do tempo decorrido entre dois sensores S_k e S_{k+1} . Considerando a trajetória $(O_1, O_2 \cdot O_{n-1}, O_n)$. Nós dividimos a trajetória $(O_1, O_2 \cdot O_i)$ e $(O_{i+1}, \cdot O_n)$ quando $t_{i+1} > \mu_{S_k, S_{k+1}} + 2\sigma_{S_k, S_{k+1}}$, que significa um comportamento incomum para o caminho de S_k para S_{k+1} .

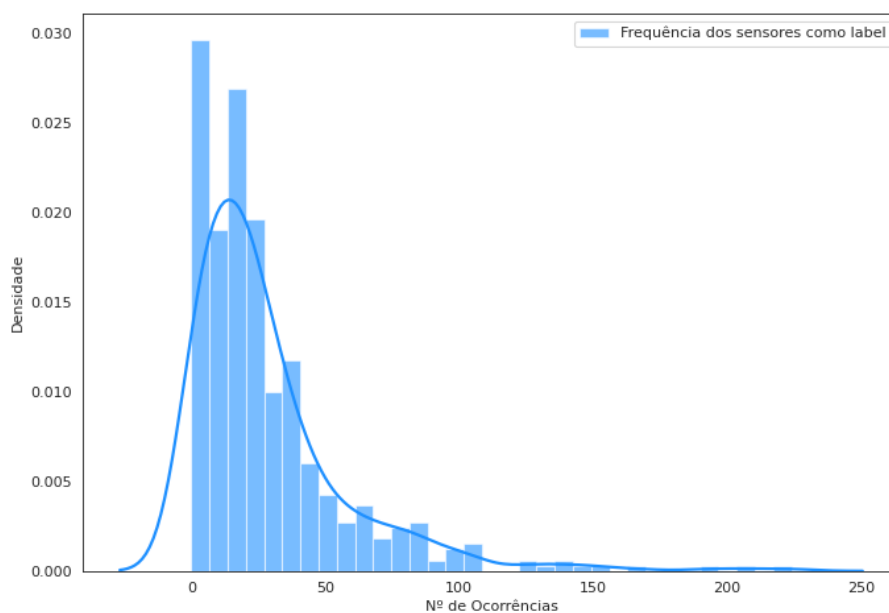
A segmentação por estatísticas de tempo é importante para a identificação das viagens no conjunto de trajetórias utilizado neste trabalho. Como exposto em Wang *et al.* (2021), a análise de uma viagem (ou corrida) de um táxi faz mais sentido do que a análise de movimento de um táxi para um dia inteiro.

Após a aplicação do *split* estatístico, as trajetórias com um único ponto são descartadas, dessa forma, o conjunto de dados passou a ter 3.151 trajetórias de 460 veículos distintos e uma média de 5,6 trajetórias por veículo.

5.4.5 Análise dos Sensores Representativos

Como o problema abordado nesta dissertação apresenta várias classes, foi realizada uma análise da frequência das ocorrências dos sensores no conjunto de rótulos. Em uma distribuição uniforme, em média cada sensor deveria ter 28.68 ocorrências no conjunto. A Figura 14 mostra a Estimativa de Densidade de *Kernel* das ocorrências de cada sensor como rótulo. Nos experimentos foram excluídas as trajetórias que apresentam como rótulo um sensor com frequência abaixo da média. Dessa forma, o número de classes reduz para de 489 para 165.

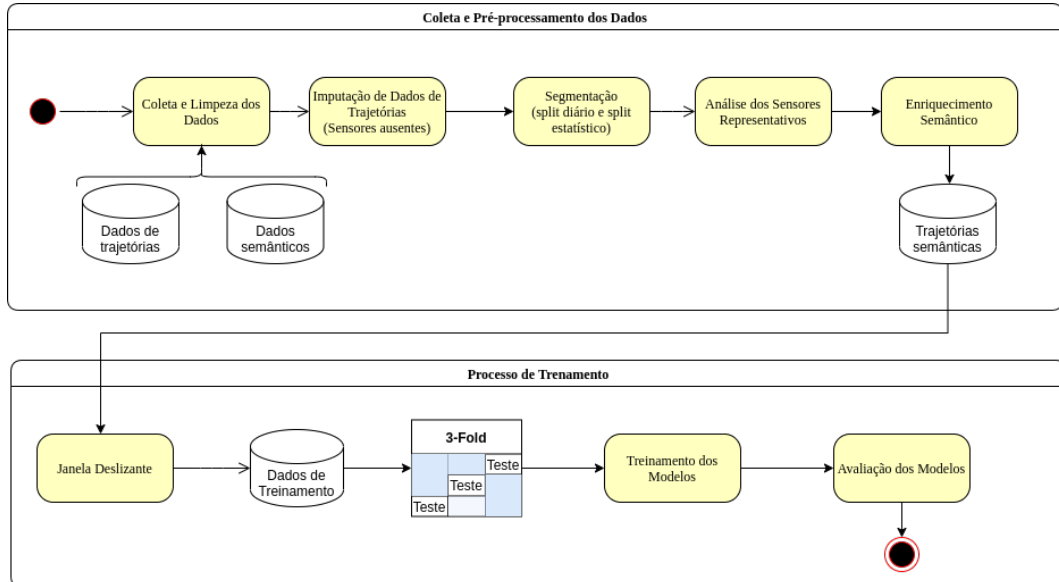
Figura 14 – Frequência dos sensores no conjunto de rótulos.



Fonte: Elaborado pelo autor.

Todo o processo de preparação do conjunto de dados e do processo de treinamento dos modelos é descrito na Figura 15.

Figura 15 – Processo de Preparação dos dados e de Treinamento dos Modelos.



Fonte: Elaborado pelo autor.

5.4.6 Métricas de Avaliação

As métricas de avaliação utilizadas neste trabalho foram acurácia e o erro de proximidade. A acurácia indica a porcentagem de previsões corretas e é descrita nas Fórmulas 5.1 e 5.2.

$$Acurcia = \frac{\sum_{i=1}^N I(l_i, \bar{l}_i)}{N} \quad (5.1)$$

onde l_i é o rótulo real da próxima localização e \bar{l}_i é o rótulo predito, tal como

$$I(a, b) = \begin{cases} 1, & \text{se } a = b \\ 0, & \text{caso contrário} \end{cases} \quad (5.2)$$

O erro de proximidade, ou *closeness error* (*ce*) representado na Fórmula 5.3, mede a proximidade do local observado ao predito. O erro de proximidade é uma forma complementar de estimar a qualidade dos modelos, o que ajuda a entender se os locais previstos são próximos

ou não dos locais esperados.

$$ce(l_i, \bar{l}_i) = d_{RN}(p_i, \bar{p}_i) \quad (5.3)$$

onde p_i e \bar{p}_i são os rótulos de geolocalização l_i e \bar{l}_i , respectivamente. Enquanto $d_{RN}(p_i, \bar{p}_i)$ é o comprimento do caminho mais curto de p_i para \bar{p}_i na rede de ruas. Para o cálculo da $d_{RN}(p_i, \bar{p}_i)$ entre dois sensores, utilizamos a biblioteca *OSMnx* (BOEING, 2017). Através da biblioteca, identificamos o caminho mais curto entre dois sensores na rede de ruas e registramos o comprimento deste caminho em uma matriz de distâncias. A Figura 16 demonstra a identificação do menor caminho na rede de ruas para duas transições entre sensores.

Figura 16 – Identificação do Menor Caminho entre dois Sensores.



Fonte: Elaborado pelo autor.

Os percentis do erro de proximidade foram analisados, pois indicam as porcentagens de previsões com um determinado valor como o limite superior de proximidade. Além disso, é informado o e o erro absoluto médio obtidos nos experimentos com diferentes grupos de atributos.

6 EXPERIMENTAÇÃO

6.1 Configuração do Experimento

Foram avaliados três modelos de aprendizado supervisionado e amplamente utilizados e eficientes no que diz respeito às técnicas consideradas neste trabalho, além de uma arquitetura de aprendizado profundo BiLSTM. Os modelos avaliados foram: KNN, *Gradient Boosting Regression Trees* (GBRT) e *Random Forest* (RF). Para os modelos KNN e RF a implementação utilizada está disponível na biblioteca *Scikit-Learn* (PEDREGOSA *et al.*, 2011), enquanto para o GBRT foi utilizado a classe *XGBClassifier* implantada em *Python* e fornecida pelo pacote *XGBoost* (CHEN; GUESTRIN, 2016).

O modelo BiLSTM foi implementado em *Python* com o pacote *Keras* (CHOLLET *et al.*, 2015). A função de ativação e *loss function* escolhidas foram *softmax* e *sparse categorical cross-entropy*, respectivamente, por serem mais recomendadas para problemas de classificação *multiclasse*, como otimizador foi utilizado o *Adam*, escolhido empiricamente. A abordagem *Grid Search* baseada na função apresentada em (BROWNLEE, 2016) foi utilizada para obter a combinação de valores para os seguintes parâmetros: *learning_rate*, *bath_size*, *epochs*, *time_embedding_size*, *spatial_embedding_size* e *units*. O Quadro 4 apresenta os valores desses hiperparâmetros. Os experimentos foram executados em uma plataforma *Linux* com uma CPU *i7* de 2,7GHz e 16GB de RAM.

Quadro 4 – Configuração de Hiperparâmetros

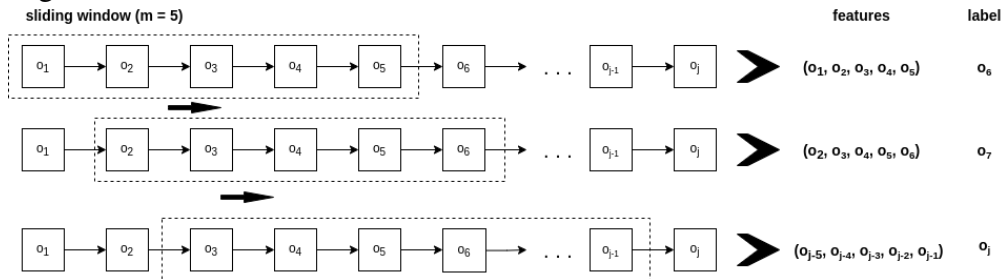
| Modelo | Parâmetro | Variação | Melhor Valor |
|--------|-------------------------------|-----------------------------|------------------|
| KNN | <i>n_neighbors</i> | 1 a 5 | 1 |
| | <i>Distance metric</i> | <i>Euclidean, manhattan</i> | <i>Euclidean</i> |
| RF | <i>n_estimators</i> | 50, 100, 150 | 100 |
| GBRT | <i>max_depth</i> | 3, 6, 9 | 6 |
| | <i>leaning_rate</i> | 0 a 1 | 1 |
| | <i>n_estimators</i> | 50, 100, 150 | 100 |
| | <i>min_child_weight</i> | 0 a 5 | 1 |
| BiLSTM | <i>constant leaning_rate</i> | 0.1, 0.01 | 0.01 |
| | <i>bath size</i> | 32, 64, 128 | 64 |
| | <i>epochs</i> | 400 a 800 | 600 |
| | <i>time_embedding_size</i> | 16, 32, 64 | 16 |
| | <i>spatial_embedding_size</i> | 16, 32, 64 | 32 |
| | <i>units</i> | 32, 64, 128 | 64 |

Fonte: Elaborado pelo autor.

6.1.1 Janela Deslizante

No processo de treinamento foi utilizada a técnica de *Janela Deslizante* para transformar as trajetórias em sequências de comprimento fixo. A janela deslizante examina cada sequência avançando uma posição até chegar ao fim. Os pontos dentro da janela foram definidos como *features*, e a posição imediatamente fora da janela é usada como *label*. A Figura 17 ilustra este processo para uma janela de tamanho $m = 5$.

Figura 17 – Técnica da Janela Deslizante.



Fonte: Adaptado de (WU *et al.*, 2017)

6.1.2 Treinamento dos Modelos

Após a preparação dos dados para o treinamento, o conjunto de dados é dividido em dados de treino e teste utilizando a técnica de *K-Fold cross-validation* com o parâmetro $K = 3$. Para a métrica acurácia foi reportada a média obtida pelo *K-Fold*, enquanto que para a métrica de Erro de proximidade, foi utilizado o melhor *fold* de cada modelo para comparação entre a distância do sensor predito e a o sensor esperado.

6.1.2.1 Preparação dos Dados de Entrada para os Classificadores de Aprendizado Supervisionado

Para os classificadores **KNN**, **RF** e **GBRT** as trajetórias fornecidas como entrada são transformadas em um vetor de tamanho fixo de $m \times d$ dimensões, em quem m é o número de observações anteriores usadas para previsão e d é o número de atributos. Neste trabalho, uma dimensão é utilizada para o *geohash* que representa a localização do sensor, uma dimensão para o *slot* de tempo e as outras dimensões para os atributos semânticos. O valor de m usado nos experimentos foi fixado em 5, e o sexto sensor é a *target*. Por exemplo, quando são analisados todos os seis atributos semânticos propostos, cada linha do conjunto de *features* é uma trajetória

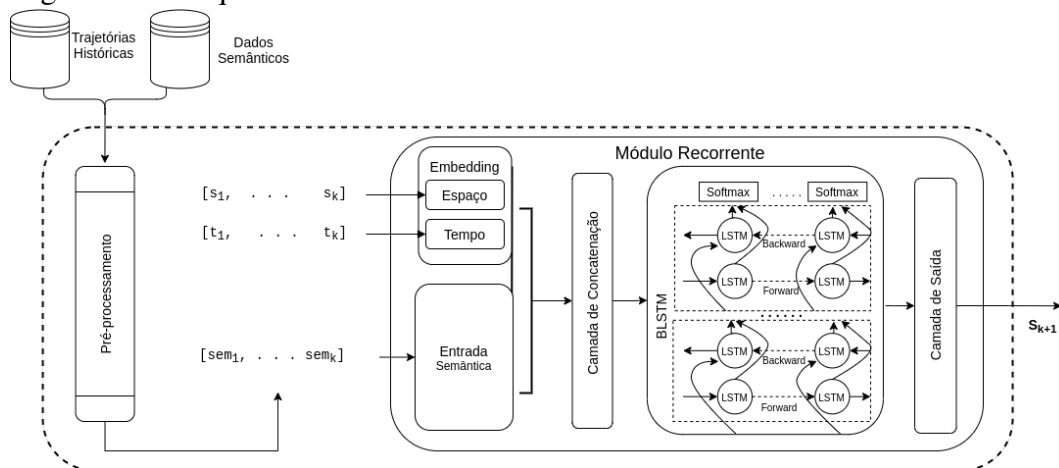
e representada pela matriz $\mathbf{X} \in \mathbb{R}^{5 \times 8}$. Uma operação *flatten* é realizada e a matriz \mathbf{X} é convertida em outra de 40 dimensões, $\mathbf{X} \in \mathbb{R}^{n \times 40}$. Em que n é o número de trajetórias. Similarmente, o sexto sensor, a *target*, é um vetor de tamanho n .

6.1.2.2 Preparação dos Dados de Entrada para a BiLSTM

Para a utilização da BiLSTM com trajetórias semânticas de veículos roubados, é realizada uma adaptação do modelo utilizado em (CRUZ *et al.*, 2019). As diferenças entre a arquitetura citada e a utilizada neste trabalho são basicamente é que esta utiliza uma LSTM Bidirecional enquanto aquela utiliza uma RNN padrão. A motivação para o uso da BiLSTM é a possibilidade de processar todas as informações de entrada disponíveis, tanto das etapas passadas como futuras, ao processar uma etapa de tempo específica.

Outra diferença é a a inclusão de camadas de entrada para os dados semânticos. De forma mais detalhada, o modelo tem (1) uma camada de **Embedding** que aplica uma transformação linear para reduzir a dimensionalidade de vetores *one-hot* espaciais e temporais preservando sua similaridade na nova representação. A camada de *Embedding* é aplicada apenas nos vetores espaciais e temporais, uma vez que os vetores semânticos já têm dimensões de baixo nível. (2) Uma camada de **concatenação** concatena as saídas da Camada de *Embedding* e da entrada semântica em um único vetor. (3) Uma Camada **BiLSTM** é responsável por aprender os padrões sequenciais da saída da Camada de Concatenação. Além disso, (4) uma Camada de **Saída** fornecida por uma camada totalmente conectada com a função de ativação *softmax*. A camada de saída converte o retorno da camada BiLSTM em um vetor que indica a probabilidade de cada rótulo ser o alvo (próximo local). A Figura 18 demonstra a arquitetura da BiLSTM utilizada.

Figura 18 – Arquitetura do Modelo BiLSTM Utilizado



Fonte: Elaborado pelo autor.

6.2 Grupos de Atributos

Para realização das análises sobre a relevância dos atributos semânticos, os atributos foram divididos em quatro grupos:

1. Espaço-Temporal - (ET): este grupo utiliza apenas os dados espaciais e os *slots* de tempo, como na abordagem tradicional de predição em EST.
2. Espaço-Tempo-Crime - (ETC): Neste cenário, as trajetórias são enriquecidas apenas com os atributos criminais.
3. Espaço-Tempo-Pois - (ETP): as trajetórias são enriquecidas apenas com os pontos de interesse.
4. Todos os Atributos - (TA): trajetórias enriquecidas com os atributos criminais e os pontos de interesse.

O Quadro 5 descreve os atributos presentes em cada grupo.

Quadro 5 – Grupos de Atributos

| Grupo | Atributos |
|--|---|
| ET | <i>geohash</i> |
| | <i>slot de tempo</i> |
| ETC | <i>geohash</i> |
| | <i>slot de tempo</i> |
| | <i>Contagem de crimes violentos contra o patrimônio</i> |
| | <i>Contagem de ocorrência de roubo de veículos</i> |
| ETP | <i>Contagem recuperação de veículos roubados</i> |
| | <i>geohash</i> |
| | <i>slot de tempo</i> |
| | <i>distância para supermercados</i> |
| | <i>distância para estacionamentos</i> |
| TA | <i>distância para oficinas mecânicas</i> |
| | <i>localização</i> |
| | <i>geohash</i> |
| | <i>Contagem de crimes violentos contra o patrimônio</i> |
| | <i>Contagem de ocorrência de roubo de veículos</i> |
| | <i>Contagem recuperação de veículos roubados</i> |
| | <i>distância para supermercados</i> |
| | <i>distância para estacionamentos</i> |
| <i>distância para oficinas mecânicas</i> | |

Fonte: Elaborado pelo autor.

6.3 Resultados

Estudo sobre os resultados de **QP1** e **QP2**: A Tabela 1 mostra as métricas acurácia e *Mean Absolut Error* (MAE) do Erro de Proximidade calculadas para modelos KNN, RF, GBRT e BiLSTM para os diferentes grupos de atributos discutidos na Seção 6.2. Todos os experimentos foram realizados usando validação cruzada *k-fold* com $k = 3$, e reporta-se a acurácia média. A partir dos resultados mais baixos obtidos por KNN, é possível notar que o problema deste trabalho é mais complexo do que prever o próximo sensor com base no vetor de trajetória mais semelhante. Além disso, os resultados demonstram que para os modelos KNN, RF e GBRT, a inclusão de atributos semânticos aumenta a acurácia.

Comparando o desempenho do modelo KNN utilizando apenas os dados espaço-temporais com os outros grupos atributos que utilizam dados semânticos, observa-se que a utilização do grupo Espaço-Tempo-Crime melhorou a acurácia em 6,3%. Da mesma forma, o treinamento com o grupo de atributos Espaço-Tempo-Pois aumenta a acurácia em 11,26%. Para este modelo, o melhor resultado para a métrica de acurácia é obtido com todos os atributos, alcançando 52.21%.

O modelo RF também melhorou a acurácia com a inclusão de dados semânticos. Utilizando separadamente os dados semânticos de crimes e pontos de interesse superam em 13% o modelo treinado apenas com o dado espaço-temporal. Além disso, o modelo alcançou o melhor resultado combinando dados espaço-temporais e todos os atributos semânticos. Este resultado reforça a eficácia dos atributos semânticos no aprimoramento dos modelos de predição de EST.

O modelo GBRT obteve o melhor resultado entre os algoritmos analisados e apresentou a melhor acurácia usando todos os dados semânticos, 68,72%. Apesar do pequeno aumento, GBRT também se beneficia do uso de atributos semânticos.

No entanto, ao contrário dos modelos de aprendizado supervisionado, os atributos semânticos não têm impacto positivo na acurácia do modelo BiLSTM. Uma justificativa seria o baixo número de trajetórias do conjunto de dados, de modo que o modelo BiLSTM não tem informações suficientes para aprender com a sequência semântica.

As Figuras 19, 20, 21, e 22 mostram a eficácia dos modelos para a métrica de erro de proximidade. Observe que, como acontece com a acurácia, a inclusão de atributos semânticos melhora o desempenho dos modelos KNN, RF e GBRT em comparação com a abordagem que usa apenas informações de espaço e tempo.

Observa-se ainda que no 90º percentil, o KNN reduz o erro de proximidade de 10km

Tabela 1 – Análise da Acurácia (Acc) e Erro Absoluto do Médio do Erro de Proximidade - MAE

| Grupo de Atributos | KNN | | RF | | GBRT | | BiLSTM | |
|--------------------|-------|---------|-------|---------|--------------|---------------|--------|---------------|
| | Acc | MAE | Acc | MAE | Acc | MAE | Acc | MAE |
| ET | 40,40 | 3241,61 | 53,14 | 2156,41 | 66,34 | 1113,48 | 63,45 | 920,82 |
| ETC | 46,77 | 2195,35 | 66,55 | 895,41 | 66,73 | 825,78 | 62,55 | 927,34 |
| ETP | 51,66 | 1811,23 | 66,96 | 940,61 | 67,63 | 864,87 | 57,91 | 978,48 |
| TA | 52,21 | 1698,54 | 68,18 | 771,96 | 68,72 | 757,00 | 59,56 | 1127,67 |

Fonte: Elaborado pelo autor.

para menos de $6km$. Ou seja, uma redução de 40% do erro de proximidade quando o modelo usa todos os atributos semânticos. Uma análise individual dos grupos de atributos mostra que os dados de pontos de interesse contribuem mais significativamente para o modelo KNN em relação à métrica de erro de proximidade. Da mesma forma, quando os modelos RF e GBRT são treinados com os atributos espaço-temporais e semânticos, esses modelos atingem um erro de proximidade de cerca de $2,5km$, representando uma redução de 68% para esta métrica no modelo RF e 29% para o GBRT quando comparados com a utilização grupo de atributos espaço-tempo.

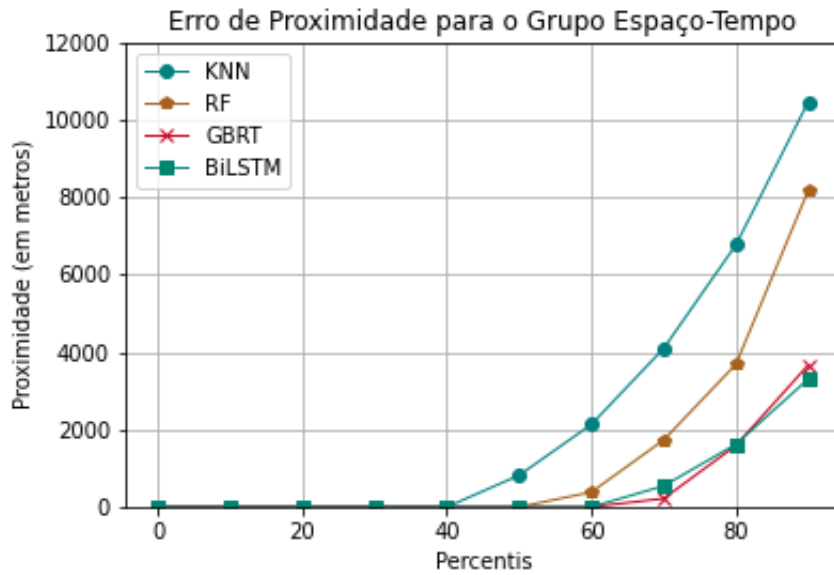
Embora a acurácia do modelo GBRT supere o modelo BiLSTM, quando são treinados apenas com informações espaço-temporais, o modelo BiLSTM obteve um erro de proximidade ligeiramente melhor. No entanto, ao contrário dos outros modelos experimentados, o BiLSTM não se beneficia da inclusão de atributos semânticos.

Análise do resultado para a **QP3**: Para maioria dos modelos, os grupos com atributos semânticos demonstraram os melhores resultados. Principalmente quando treinados usando a distância para os pontos de interesse, esses atributos são os mais eficazes para melhorar o desempenho da acurácia nos modelos de aprendizado de supervisionado. Isso confirma o que já era esperado pela Secretaria de Segurança Pública do Ceará, já que os veículos roubados costumam ser abandonados próximos desses pontos de interesse. Assim, as trajetórias de veículos roubados tendem a passar por sensores próximos a esses locais.

Estudo sobre os resultados do **QP4**: Foram analisadas diferentes representações da informação espacial para o modelo BiLSTM com dados semânticos:

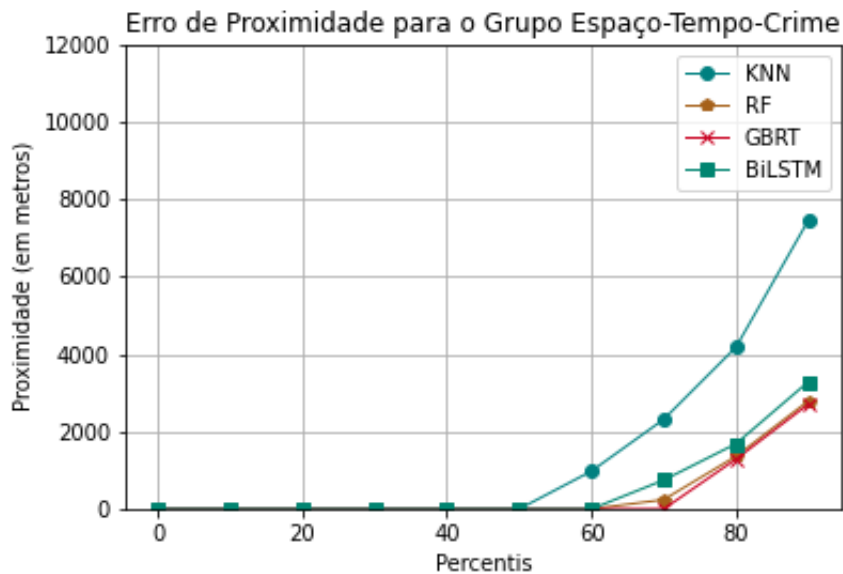
1. **Baseado em cluster**: o algoritmo *DBSCAN* (ESTER *et al.*, 1996) foi aplicado utilizando a matriz de distâncias dos sensores na rede de ruas. O parâmetro *MinPts* foi definido para ser igual a 1, este parâmetro indica o número mínimo de pontos, neste trabalho sensores, necessário para criação dos *clusters*, configurar esse parâmetro com um número maior que um iria definir os sensores distantes de outros como ruído, pois é necessário que o *cluster* encadeie sensores próximos por densidade. Já parâmetro *Epsilon*, que define a distância

Figura 19 – Análise do Erro de Proximidade para o grupo Espaço-Tempo.



Fonte: Elaborado pelo autor.

Figura 20 – Análise do Erro de Proximidade para o grupo Espaço-Tempo-Crime.



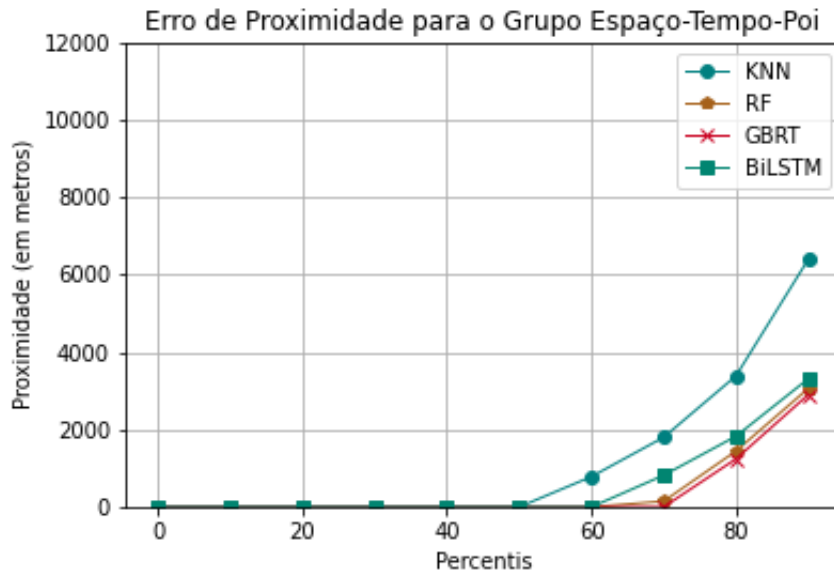
Fonte: Elaborado pelo autor.

para agrupamento dos pontos, foi ajustado empiricamente. Para esse parâmetro a seguinte variação foi testada: (100, 200, 300, 500). A melhor a melhor clusterização foi obtida com $Epsilon = 200$;

Com a aplicação da clusterização, os 489 sensores foram reunidos em 304 *clusters*

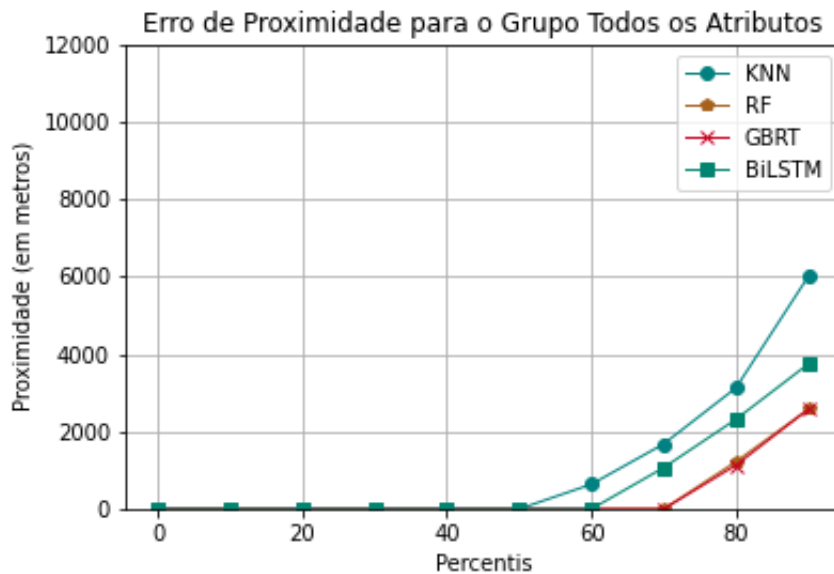
2. **Índice baseado em Grid:** a *grid* virtual que divide o espaço geográfico da rede dos sensores em um conjunto de linhas e colunas. O tamanho da célula foi definido como $500m \times 500m$;

Figura 21 – Análise do Erro de Proximidade para o grupo Espaço-Tempo-Poi.



Fonte: Elaborado pelo autor.

Figura 22 – Análise do Erro de Proximidade para o grupo Todos os Atributos.



Fonte: Elaborado pelo autor.

3. **Geohash:** as latitudes e longitudes dos sensores foram convertidas em uma sequência de caracteres com a precisão de 50 caracteres.

As diferentes representações espaciais foram testadas para verificar o impacto destas representações no problema deste trabalho.

Os resultados da Tabela 2 mostram que nossos dados são sensíveis à utilização de clusterização e representação por *grid*. Esse fato pode ser explicado pelo comportamento de motoristas de veículos roubados. Como os sistemas de vigilância da cidade tendem a ter

altas taxas de interceptação de veículos roubados quando eles aparecem em vários sensores na mesma viagem, os motoristas de veículos roubados tendem a escapar dos sensores, o que torna as trajetórias muito esparsas no tempo e curtas (ou seja, com poucos sensores registrados). Isso pode justificar as trajetórias com restrição de roubo se beneficiarem da representação por *Geohash*, uma vez que não reduzem as trajetórias como abordagens baseadas em *Cluster* ou índice da *Grid*. *Geohash* garante que cada sensor tenha uma representação *hash* exclusiva; isso se torna mais eficaz, dentre as que foram avaliadas nesta pesquisa, representando as transições (bordas) entre as junções (nós) da rede de ruas. Diferente das abordagens de índice da *Grid* ou representação baseada em *Cluster*, onde as trajetórias podem ser representadas pelo mesmo índice ou *id* do *cluster* repetidamente na sequência.

Tabela 2 – Análise do Nível da Representação Espacial

| Grupo de Atributos | Cluster | Índice da Grid | Geohash |
|---------------------------|----------------|-----------------------|----------------|
| Espaço-Tempo | 62,09 | 60,60 | 63,45 |
| Espaço-Tempo-Crime | 57,77 | 60,41 | 62,55 |
| Espaço-Tempo-Poi | 54,48 | 54,06 | 57,91 |
| Todos os Atributos | 52,72 | 52,10 | 59,56 |

Fonte: Elaborado pelo autor.

7 CONCLUSÕES E TRABALHOS FUTUROS

Esta pesquisa avaliou a predição da próxima localização para trajetórias de veículos roubados com enriquecimento semântico. Os experimentos demonstraram que dados criminais e atributos de pontos de interesse relacionados ao roubo de veículos melhoram o desempenho dos modelos de aprendizado supervisionado avaliados. A análise individual dos grupos de atributos mostrou que o grupo de pontos de interesse é o que mais contribui para a previsão. Porém, ao contrário dos modelos de Aprendizado Supervisionado, o modelo BiLSTM proposto não se beneficia da inclusão dos atributos analisados. Este fato pode ocorrer por se tratar de um conjunto de dados com número reduzido de trajetórias. Outros conjuntos de dados de trajetórias que não são de veículos roubados não foram utilizados, pois entende-se que o objetivo de avaliar a contribuição dos atributos semânticos, analisados neste estudo, na predição do próximo sensor em EST seria prejudicado. Dados de trajetórias externas de veículos com restrições de roubo dependem da disponibilização por agências de segurança pública o que dificulta o acesso a outros conjuntos de dados de trajetórias de veículos roubados a serem utilizados nesta dissertação.

Por fim, foram avaliados três níveis de representação dos dados espaciais. O nível espacial de maior granularidade, *Geohash*, foi o mais eficaz dentre os avaliados para o problema EST. Isso pode ser explicado pela característica das trajetórias de veículos roubados de serem compostas por poucos pontos, se compararmos com trajetórias dos veículos sem essa restrição.

Parte dos procedimentos metodológicos deste trabalho já foi reconhecida pela comunidade científica no artigo (CRUZ *et al.*, 2021) e no momento da escrita desta dissertação, os resultados das análises aqui apresentadas foram aceitas para publicação no seguinte artigo: **Predicting the Next Location for Trajectories From Stolen Vehicles** submetido para a conferência ICTAI 2021.

Como trabalhos futuros, pretende-se examinar como aprender com as distâncias na rede de ruas e minimizar o erro de proximidade. Outra possibilidade é investigar a realização de embeddings de sub-trajetórias. Além disso, é necessário investigar se outras *features* como distância, velocidade e *embedding* de subtrajetórias podem contribuir na predição. Planeja-se também aprimorar os dados para considerar a incerteza e a dispersão das trajetórias obtidas dos sensores. Outras técnicas para lidar com a incompletude e esparsidade dos dados também podem ser aplicadas.

REFERÊNCIAS

- BAILEY, T. C.; GATRELL, A. C. **Interactive spatial data analysis**. [S.l.]: Longman Scientific & Technical Essex, 1995. v. 413.
- BOEING, G. Osmnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. **Computers, Environment and Urban Systems**, Elsevier, v. 65, p. 126–139, 2017.
- BOGORNY, V.; RENSO, C.; AQUINO, A. R. de; SIQUEIRA, F. de L.; ALVARES, L. O. Constant—a conceptual data model for semantic trajectories of moving objects. **Transactions in GIS**, Wiley Online Library, v. 18, n. 1, p. 66–88, 2014.
- BOX, G. E.; JENKINS, G. M.; REINSEL, G. C.; LJUNG, G. M. **Time series analysis: forecasting and control**. [S.l.]: John Wiley & Sons, 2015.
- BREIMAN, L. Random forests. **Machine learning**, Springer, v. 45, n. 1, p. 5–32, 2001.
- BROWNLEE, J. How to grid search hyperparameters for deep learning models in python with keras. **Retrieved April**, v. 20, p. 2018, 2016.
- BUCHER, D. Vision paper: Using volunteered geographic information to improve mobility prediction. In: ACM. **Proceedings of the 1st ACM SIGSPATIAL Workshop on Prediction of Human Mobility**. [S.l.], 2017. p. 2.
- CHAINEDY, S.; TOMPSON, L.; UHLIG, S. The utility of hotspot mapping for predicting spatial patterns of crime. **Security journal**, Springer, v. 21, n. 1-2, p. 4–28, 2008.
- CHEN, M.; YU, X.; LIU, Y. Mining moving patterns for predicting next location. **Information Systems**, Elsevier, v. 54, p. 156–168, 2015.
- CHEN, T.; GUESTRIN, C. XGBoost: A scalable tree boosting system. In: **Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. New York, NY, USA: ACM, 2016. (KDD '16), p. 785–794. ISBN 978-1-4503-4232-2. Acesso em: 14 out 2021.
- CHOLLET, F. **Deep Learning With Python**. [S.l.]: Manning Publications Co, 2018.
- CHOLLET, F. *et al.* **Keras**. [S.l.: s.n.], 2015. Disponível em: <https://keras.io>. Acesso em: 14 out 2021.
- CRUZ, L. A.; ZEITOUNI, K.; MACEDO, J. A. F. de. Trajectory prediction from a mass of sparse and missing external sensor data. In: IEEE. **2019 20th IEEE International Conference on Mobile Data Management (MDM)**. [S.l.], 2019. p. 310–319.
- CRUZ, L. A.; ZEITOUNI, K.; SILVA, T. L. C. da; MACEDO, J. A. F. de; SILVA, J. S. da. Location prediction: a deep spatiotemporal learning from external sensors data. **Distributed and Parallel Databases**, Springer, v. 39, n. 1, p. 259–280, 2021.
- DUONG-TRUNG, N.; SCHILLING, N.; SCHMIDT-THIEME, L. Near real-time geolocation prediction in twitter streams via matrix factorization based regression. In: **Proceedings of the 25th ACM international on conference on information and knowledge management**. [S.l.: s.n.], 2016. p. 1973–1976.

- ECK, J.; CHAINEY, S.; CAMERON, J.; WILSON, R. **Mapping crime: Understanding hotspots**. [S.l.]: National Institute of Justice, 2005.
- ESTER, M.; KRIEGEL, H.-P.; SANDER, J.; XU, X. *et al.* A density-based algorithm for discovering clusters in large spatial databases with noise. In: **kdd**. [S.l.: s.n.], 1996. v. 96, n. 34, p. 226–231.
- FAN, X.; GUO, L.; HAN, N.; WANG, Y.; SHI, J.; YUAN, Y. A deep learning approach for next location prediction. In: IEEE. **2018 IEEE 22nd international conference on Computer Supported Cooperative Work in Design ((CSCWD))**. [S.l.], 2018. p. 69–74.
- FENG, J.; LI, Y.; ZHANG, C.; SUN, F.; MENG, F.; GUO, A.; JIN, D. Deepmove: Predicting human mobility with attentional recurrent networks. In: INTERNATIONAL WORLD WIDE WEB CONFERENCES STEERING COMMITTEE. **Proceedings of the 2018 World Wide Web Conference**. [S.l.], 2018. p. 1459–1468.
- FRIEDMAN, J. H. Greedy function approximation: a gradient boosting machine. **Annals of statistics**, JSTOR, p. 1189–1232, 2001.
- FRIEDMAN, J. H. Stochastic gradient boosting. **Computational Statistics & Data Analysis**, Elsevier, v. 38, n. 4, p. 367–378, 2002.
- GÉRON, A. **Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems**. [S.l.]: O’Reilly Media, 2019.
- GHOSH, S. K.; GHOSH, S. Modeling individual’s movement patterns to infer next location from sparse trajectory traces. In: IEEE. **2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)**. [S.l.], 2018. p. 693–698.
- HART, T.; ZANDBERGEN, P. Kernel density estimation and hotspot mapping. In: **Policing: An international journal of police strategies & management**. [S.l.]: Emerald Group Publishing Limited, 2014.
- HUANG, C.; ZHANG, J.; ZHENG, Y.; CHAWLA, N. V. Deepcrime: attentive hierarchical recurrent networks for crime prediction. In: **Proceedings of the 27th ACM International Conference on Information and Knowledge Management**. [S.l.: s.n.], 2018. p. 1423–1432.
- IACA, S.; ELDER, J.; IACA, S.; BRUCE, C. W.; SANTOS, R. B.; RODRIGUEZ, E.; COUNTY, C. L. A.; STEINER, F.; POLICE, A. F.; WYCKOFF, L. Definition and types of crime analysis. **Citeseer**. [S.l.], 2014.
- KAMW, F. S. **Utilizing Big Trajectory Data For Urban Visual Analytics and Accessibility Studies**. Tese (Doutorado) — Kent State University, 2019.
- KONG, D.; WU, F. Hst- lstm: A hierarchical spatial-temporal long-short term memory network for location prediction. In: **IJCAI**. [S.l.: s.n.], 2018. v. 18, n. 7, p. 2341–2347.
- LIU, Q.; WU, S.; WANG, L.; TAN, T. Predicting the next location: A recurrent model with spatial and temporal contexts. In: **Thirtieth AAAI conference on artificial intelligence**. [S.l.: s.n.], 2016.
- LIU, Q.; ZUO, Y.; YU, X.; CHEN, M. Ttdm: A travel time difference model for next location prediction. In: IEEE. **2019 20th IEEE International Conference on Mobile Data Management (MDM)**. [S.l.], 2019. p. 216–225.

- LIU, S.; LIU, C.; LUO, Q.; NI, L. M.; KRISHNAN, R. Calibrating large scale vehicle trajectory data. In: **IEEE. 2012 IEEE 13th International Conference on Mobile Data Management**. [S.l.], 2012. p. 222–231.
- MATHEW, W.; RAPOSO, R.; MARTINS, B. Predicting future locations with hidden markov models. In: **Proceedings of the 2012 ACM conference on ubiquitous computing**. [S.l.: s.n.], 2012. p. 911–918.
- NIEMEYER, G. Geohash. **Retrieved June**, [S.l.], v. 6, p. 2018, 2008.
- OPENSTREETMAP CONTRIBUTORS. **Planet dump retrieved**. [S.l.: s.n.], 2017. Disponível em: <https://www.openstreetmap.org>. Acesso em: 14 out 2021.
- ORÓSTICA, K.; POBLETE, B. Mining the relationship between car theft and places of social interest in Santiago Chile. In: **Companion Proceedings of The 2019 World Wide Web Conference**. [S.l.: s.n.], 2019. p. 811–814.
- PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V. *et al.* Scikit-learn: Machine learning in python. **Journal of machine learning research**, v. 12, n. Oct, p. 2825–2830, 2011.
- RASCHKA, S.; MIRJALILI, V. **Python machine learning**. [S.l.]: Packt Publishing Ltd, 2017.
- RÊGO, L. G. C. do; SILVA, T. L. C. da; MAGALHÃES, R. P.; MACÊDO, J. A. F. de; SILVA, W. C. P. Exploiting points of interest for predictive policing. In: **Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Advances in Resilient and Intelligent Cities**. [S.l.: s.n.], 2020. p. 20–28.
- 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. In: **ACM. Proceedings of the 20th International Database Engineering & Applications Symposium**. [S.l.], 2016. p. 34–42.
- SANTOS, R. B. **Crime analysis with crime mapping**. [S.l.]: Sage publications, 2016.
- SVENNERBERG, G. **Beginning Google Maps API 3**. 2nd. ed. USA: Apress, 2010. ISBN 1430228024.
- TRASARTI, R.; GUIDOTTI, R.; MONREALE, A.; GIANNOTTI, F. Myway: Location prediction via mobility profiling. **Information Systems**, Elsevier, v. 64, p. 350–367, 2017.
- WANG, H.; KIFER, D.; GRAIF, C.; LI, Z. Crime rate inference with big data. In: **Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining**. [S.l.: s.n.], 2016. p. 635–644.
- WANG, S.; BAO, Z.; CULPEPPER, J. S.; CONG, G. A survey on trajectory data management, analytics, and learning. **ACM Computing Surveys (CSUR)**, ACM New York, NY, USA, v. 54, n. 2, p. 1–36, 2021.
- WEISEL, D. L.; SMITH, W. R.; GARSON, G. D.; PAVLICHEV, A.; WARTELL, J. **Motor vehicle theft: Crime and spatial analysis in a non-urban region**. Rockville, MD: National Institute of Justice, 2006.

- WU, F.; FU, K.; WANG, Y.; XIAO, Z.; FU, X. A spatial-temporal-semantic neural network algorithm for location prediction on moving objects. **Algorithms, Multidisciplinary Digital Publishing Institute**, [S.l.], v. 10, n. 2, p. 37, 2017.
- YAO, D.; ZHANG, C.; HUANG, J.; BI, J. Serm: A recurrent model for next location prediction in semantic trajectories. In: ACM. **Proceedings of the 2017 ACM on Conference on Information and Knowledge Management**. [S.l.], 2017. p. 2411–2414.
- YU, C.-H.; DING, W.; CHEN, P.; MORABITO, M. Crime forecasting using spatio-temporal pattern with ensemble learning. In: SPRINGER. **Pacific-Asia Conference on Knowledge Discovery and Data Mining**. [S.l.], 2014. p. 174–185.
- ZENG, J.; TANG, H.; WU, Y.; LIU, L.; HIROKAWA, S. Predict the next location from trajectory based on spatiotemporal sequence. In: IEEE. **2019 8th International Congress on Advanced Applied Informatics (IIAI-AAI)**. [S.l.], 2019. p. 109–114.
- ZHANG, C.; ZHANG, K.; YUAN, Q.; ZHANG, L.; HANRATTY, T.; HAN, J. Gmove: Group-level mobility modeling using geo-tagged social media. In: **Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. [S.l.: s.n.], 2016. p. 1305–1314.
- ZHENG, Y. Trajectory data mining: an overview. **ACM Transactions on Intelligent Systems and Technology (TIST)**, ACM New York, NY, USA, v. 6, n. 3, p. 1–41, 2015.