

Vem Tranquilo: Rotas Eficientes baseado na Dinâmica Urbana Futura com *Deep Learning* e Computação de Borda

Allan M. de Souza, Leandro A. Villas

¹Instituto de Computação - Universidade Estadual de Campinas - UNICAMP

Abstract. *Several urban factors can impact in a route planning decision, such as travel time, fuel consumption, scenery, and even public safety. Moreover, many of these factors have a space-time correlation, i.e., their dynamics may vary according to day and time for different regions. However, current literature solutions do not consider this future urban dynamics during route planning task, which potentially degrades system efficiency. However, advances in edge computing coupled with deep learning techniques can provide robust and accurate future knowledge, thereby enabling more efficient decision making. Thus, this paper presents VTq a route planning system that is able to predict future dynamics of urban factors and consider such dynamics during route planning to plan a more reliable and efficient route. The results have shown that VTq is able to make predictions with an average error of 0.001 and also improve the route planning efficiency by at least 50% for most users when compared to state-of-the-art solutions.*

Resumo. *Muitos fatores podem impactar em uma decisão de planejamento de rota, como tempo de viagem, consumo de combustível, cenário, e até mesmo segurança. Além disso, muitos desses fatores possuem um correlação espaço-temporal, ou seja, suas dinâmicas podem variar de acordo com dia e hora para diferentes regiões. Contudo, as soluções da literatura não consideram a dinâmica urbana futura no planejamento de rotas, o que potencialmente degrada a eficiência do sistema. Entretanto, os avanços da computação de borda juntamente com as técnicas de deep learning podem fornecer um conhecimento futuro robusto e preciso, assim, permitindo tomadas de decisões mais eficientes. Sendo assim, esse trabalho apresenta VTq um sistema para planejamento de rota que é capaz de prever a dinâmica futura dos fatores urbanos e considerar tal dinâmica durante o planejamento de rota para planejar rotas mais seguras e eficientes. Os resultados mostraram que o VTq é capaz de realizar previsões com um MSE de 0.001 e melhorar a eficiência do planejamento de rota em pelo menos 50% para a maioria dos usuários quando comparado com soluções do estado da arte da literatura.*

1. Introdução

Planejamento de rota é um requisito essencial nos sistemas de transporte inteligentes (*Intelligent Transportation Systems* – ITS) para otimizar a gerência e eficiência do tráfego. Diversos fatores urbanos podem influenciar no planejamento de rota, incluindo, tempo de viagem, distância, consumo de combustível e emissões de CO₂, segurança, qualidade do asfalto e até mesmo o cenário e segurança [Taha and AbuAli 2018]. Neste contexto, diversos trabalhos foram propostos visando evitar congestionamentos [de Souza et al. 2017], minimizar tempo de viagem [Pan et al. 2017] e consumo de combustível [Doolan and Muntean 2017, de Souza et al. 2015], e até mesmo maximizar a segurança pública [Galbrun et al. 2016]. Entretanto, esses trabalhos empregam o mesmo planejamento (i.e., utilizam as mesmas métricas) para todos os usuários, o que está longe de ser o cenário ideal, pois diferentes usuários podem ter preferências diferentes para um mesmo planejamento de rota. Por exemplo, usuários mais cautelosos podem preferir uma rota mais segura, porém mais longa, enquanto usuários com

tempo limitado tendem a preferir rotas mais rápidas [de Souza et al. 2019, Amirgholy et al. 2017]. Assim, como permitir tal dinamicidade nos serviços de recomendação de rota é um problema em aberto na literatura.

Além disso, é importante destacar que muitos desses fatores urbanos possuem uma correlação espaço-temporal (i.e., podem variar ao longo do dia em diferentes regiões), por exemplo, congestionamentos ocorrem em regiões recorrentes em horários de pico de dias úteis, porém essas mesmas regiões podem não apresentar congestionamentos durante os mesmos horários nos finais de semana. Um outro exemplo são fatores de risco à segurança, onde uma mesma região pode fornecer oportunidades diferentes para atividades criminais ao longo do dia, consequentemente aumentando a incidência criminal nessa região durante um determinado período. Contudo, nenhuma solução de recomendação de rota considera a dinâmica futura dos fatores urbanos durante o planejamento de rota, assim, precisando realizar um re-planejamento de rota frequente para garantir uma rota mais confiável e eficiente, o que resulta no aumento custo da rede e computacional da solução devido a grande número de troca de mensagens e re-planejamentos de rotas [Taha and AbuAli 2018]. Sendo assim, saber a dinâmica futura desses fatores urbanos antecipadamente é essencial para prover um planejamento de rota mais eficiente e confiável, além de reduzir o custo de rede e computacional do sistema.

Nesse cenário, as redes veiculares (*Vehicular Ad hoc Networks* – VANETs) e redes de quinta geração (5G) [Campolo et al. 2017]; algoritmos de aprendizado de máquina e *deep learning* [Ye et al. 2018]; e a computação de borda (*Multi-access Edge Computing* – MEC) [Liu et al. 2017a] são essenciais para permitir o desenvolvimento de soluções para lidar com as limitações apresentadas. Particularmente, as VANETs permitem o sensoriamento do ambiente urbano utilizando diversos sensores presentes nos veículos e infraestruturas, além de permitir a comunicação entre veículos, infraestruturas de comunicação, pessoas e objetos inteligentes. Por outro lado, algoritmos de aprendizado de máquina são capazes de extrair informações úteis e prever o comportamento futuro dos fatores urbanos utilizando as informações obtidas através do sensoriamento do ambiente urbano, enquanto a computação de borda disponibiliza recursos computacionais na borda da rede (i.e., veículos e servidores de borda instalados em infraestruturas de comunicação ou em pontos de acesso), aumentando recursos de armazenamento e processamento dos sistemas, consequentemente permitindo um tempo de resposta muito baixo ou até mesmo em tempo real.

Sendo assim, apresentamos VTq (**Vem Tranquilo**) um ITS para planejamento de rota personalizável e eficiente considerando a dinâmica futura dos fatores urbanos baseado em *deep learning* e computação de borda. Para isso, VTq trata os seguintes desafios: (i) Como permitir um planejamento de rota dinâmico e eficiente para diferentes fatores urbanos? (ii) Como prever a dinâmica futura dos fatores urbanos com precisão? e (iii) Como planejar uma rota em tempo hábil considerando as dinâmicas dos fatores urbanos? Para uma predição eficiente, VTq implementa uma rede neural capaz de aprender padrões e explorar a intercorrelação em longas séries de dados temporais. Por fim, para permitir um planejamento adaptativo e considerar as predições durante o planejamento de rota, VTq implementa um algoritmo de aprendizado por reforço, o qual aprende planejar a rota mais eficiente utilizando técnicas de recompensa e punição. Os resultados mostraram que VTq é capaz de realizar predições com um MSE de apenas 0.001, além de recomendar rotas até 50% mais eficientes para a maioria dos usuários em comparação as soluções da literatura.

O restante do trabalho é dividido da seguinte forma. Seção 2 descreve os trabalhos relacionados apresentando suas vantagens e limitações. Seção 3 descreve VTq. Seção 4 apresenta a análise dos resultados. Por fim, Seção 5 apresenta conclusões e trabalhos futuros.

2. Trabalhos Relacionados

Serviços de navegação (*Vehicular Navigation Systems – VNS*) como o INRIX, TomTom, Google Maps e Waze possuem informações de tráfego em tempo real, assim são capazes planejar rotas mais rápidas [de Souza et al. 2019]. Além disso, muitos ITS foram propostos para realizar um planejamento de rota considerando minimizar congestionamentos [Pan et al. 2017], reduzir emissões de CO_2 , e até mesmo aumentar a segurança [Taha and AbuAli 2018].

[Pan et al. 2017] apresentaram o DIVERT, um ITS distribuído para planejamento de rotas com o objetivo de evitar congestionamentos. Para permitir um planejamento de rota em tempo real, DIVERT faz o *offloading* do cálculo de rotas nos veículos. Para construir um conhecimento preciso sobre as condições de tráfego, os veículos compartilham informações de mobilidade através da rede veicular, em seguida essas informações são reportadas para um servidor central para geração do conhecimento sobre a mobilidade urbana. Para evitar o problema de criar novos focos de congestionamentos durante o planejamento de rota, DIVERT implementa um mecanismo colaborativo para distribuir melhor o tráfego de veículos.

[Doolan and Muntean 2017] propuseram o EcoTrec, um ITS que tem como objetivo a redução de emissões de CO_2 . Para isso, o EcoTrec extrai um conhecimento sobre as emissões de CO_2 nas vias urbanas baseado no consumo de combustível de cada veículo. Após a extração desse conhecimento, EcoTrec compartilha essas informações com os veículos para que eles possam calcular a rota com menos emissões de CO_2 . Entretanto, EcoTrec baseia-se em um algoritmo determinístico para planejar a rota com menos emissões de CO_2 , o que potencialmente pode gerar focos de congestionamentos e degradar a eficiência do sistema.

Visando melhorar a segurança de motoristas e passageiros, [Galbrun et al. 2016] introduziram o SafePaths, um sistema capaz de calcular rotas mais seguras para cada usuário baseado no conhecimento extraído sobre as atividades criminais da cidade, as quais são reportadas pelos departamentos de polícia da cidade. Entretanto, o SafePaths não considera as condições de tráfego durante o planejamento de rotas.

Em nosso trabalho anterior [de Souza et al. 2018], propusemos o #PAS, um ITS para planejamento de rotas que fornece um bom *trade-off* entre mobilidade e segurança, pois nem sempre a rota mais rápida é a mais segura. Dessa forma, soluções que focam em melhorar a mobilidade podem produzir risco à segurança dos motoristas e passageiros, enquanto soluções que focam na rota mais rápida podem direcionar veículos para ruas congestionadas, assim degradando a eficiência da mobilidade urbana. Além disso, o #PAS implementa uma arquitetura híbrida e protocolos de comunicação eficientes para reduzir o *overhead* do sistema e permitir uma boa escalabilidade. O planejamento de rota baseia-se em um algoritmo de caminhos mínimos com restrição de recursos, o qual é uma abordagem eficiente para cálculo de caminhos com múltiplos objetivos.

Entretanto, esses trabalhos não consideram as variações futuras da dinâmica urbana durante o planejamento de rotas, o que pode reduzir a confiabilidade e eficiência da rota planejada em um futuro próximo. Nesse contexto, para não só analisar como a dinamicidade do ambiente urbano pode impactar nas rotas planejadas, mas também para propor um planejamento de rota mais eficiente, propomos VTq, um ITS que realiza predições sobre os fatores urbanos e utiliza essas predições para um planejamento mais robusto eficiente.

3. VTq: Sistema de Recomendação de Rota Ciente da Dinâmica Urbana Futura

VTq foi desenvolvido para lidar com as seguintes limitações apresentadas pelas soluções de planejamento de rota da literatura: (i) planejamento de rota personalizável; (ii) prever a dinâmica futura de fatores urbanos que podem impactar em um planejamento de rota; e (iii) considerar a variação futura dos fatores urbanos durante o planejamento de rota. A

extração do conhecimento sobre os fatores urbanos e sua dinamicidade será possível através do monitoramento, coleta, e exploração dos dados produzidos por dispositivos inteligentes [Dong et al. 2017, Campolo et al. 2017]. Portanto, as tecnologias de comunicação tais como 5G e V2X, redes veiculares, internet das coisas e etc desempenharão um papel essencial.

Para permitir o roteamento personalizável, VTq faz o *offloading* do conhecimento gerado pelo sistema (e.g., fatores urbanos) para os veículos, assim o planejamento de rota personalizada é computado localmente. Além disso, o *offloading* desse conhecimento para os veículos minimiza problemas de privacidade e latência, uma vez que as informações pessoais (e.g., origem, destino, e rota) de cada veículo não são transmitidas pela rede para um servidor central.

Por outro lado, para prever a dinâmica futura dos fatores urbanos, VTq utiliza uma rede neural recorrente (RNN) [Liu et al. 2017b], a qual é uma técnica de *deep learning* eficiente para predição de dados espaço-temporais. Assim, baseado nas informações atuais coletadas do ambiente urbano juntamente com o histórico dessas informações, VTq é capaz de prever como será o comportamento de tal fator urbano na próxima janela temporal (i.e., definida pela aplicação).

Por fim, para permitir não só um algoritmo adaptativo (e.g., o mesmo algoritmo possa ser usado para diferentes objetivos) para o planejamento de rota, além de considerar a variação futura da dinâmica urbana durante o planejamento, VTq implementa um algoritmo baseado em aprendizado por reforço. Assim, utilizando técnicas de recompensa e punição baseado nas ações que podem ser tomadas (e.g., as ruas que podem ser selecionadas para construir um caminho), o algoritmo aprende a planejar a melhor rota possível (de acordo com a preferência de cada usuário) considerando a dinâmica futura do ambiente urbano.

3.1. Formalização do Problema

O cenário é representado por um grafo direcionado $G = (V, E)$, onde V representa o conjunto de interseções do cenário, enquanto E representa o conjunto de vias urbanas $uv \in E$ conectando duas interseções $u, v \in V$. Cada via urbana possui atributos para representar a dinâmica urbana x_{uv}^t no momento t . Assim, baseado no par de origem e destino do veículo, um caminho $P \subseteq E$ deve ser planejado para conectar a origem e destino. Além disso, o custo total de um planejamento de uma rota é igual a soma dos valores dos fatores urbanos de cada via urbana $uv \in P$ no momento em que o veículo passa por tal via, ou seja, $c(P) = \sum_{uv \in P} x_{uv}^t$. Neste cenário, o caminho mais eficiente P é aquele que minimiza o custo total do caminho considerando dinâmica de um fator urbano para cada $uv \in P$ de acordo com seu valor no tempo t .

3.2. Extrair Conhecimento sobre Fatores Urbanos

Para extração do conhecimento sobre a mobilidade, VTq faz uso das informações produzidas pelos veículos, assim, com ajuda de tecnologias de comunicação tais como, 802.11p e 5G, os veículos podem compartilhar e reportar informações sobre o trânsito para um servidor de borda ou nuvem. Por outro lado, as informações de segurança são obtidas através das bases de dados dos boletins de ocorrência dos departamentos de polícia e/ou de segurança pública, as quais podem fornecer essas informações diretamente para o VTq através da *Internet* ou por relatórios. Além disso, essas informações também podem ser coletadas de redes sociais e sites públicos disponíveis na *Web*. Dessa forma, essas informações podem ser acessadas em tempo real.

Para coleta das informações de mobilidade e segurança, alguns desafios essenciais devem ser solucionados, como, por exemplo: (i) como evitar a sobrecarga da rede durante o

compartilhamento de informações sobre o tráfego? (ii) como extrair o conhecimento sobre as condições de tráfego de maneira eficiente sem produzir um alto *overhead* na rede? e (iii) como representar a segurança nas vias urbanas de maneira eficiente?

Portanto, para superar esses desafios VTq faz uso da arquitetura e protocolos implementados em nossos trabalhos anteriores [de Souza et al. 2018, de Souza et al. 2017]. Em resumo, VTq implementa uma arquitetura híbrida juntamente com um protocolo de compartilhamento de informações que utiliza comunicações V2V e V2I para selecionar o melhor veículo para compartilhar informações de uma região. Assim, reduzindo o número de mensagens transmitidas e garantindo a extração de um bom conhecimento sobre as condições de tráfego (a descrição detalhada dessa arquitetura e protocolo pode ser encontrada em [de Souza et al. 2018]). Por outro lado, para extração do conhecimento sobre segurança VTq extrai as coordenadas geográficas de cada evento criminal recebido, em seguida aplica uma função de densidade para verificar a densidade de crimes em cada via urbana, baseado nas posições das interseções que definem cada via (e.g., as posições de $u, v \in V \mid uv \in E$), uma descrição detalhada desse processo de extração pode ser encontrada em [de Souza et al. 2018].

3.3. Predizendo a Dinâmica Futura dos Fatores Urbanos

Uma arquitetura especial de RNN que pode ser aplicada à predição de dados temporais é a LSTM (*Long Short Term Memory*), a qual tem mostrado resultados substanciais por lembrar informações e explorar intercorrelações por longos períodos [Qiu et al. 2018].

A maioria das abordagens para planejamento de rotas atuais não garantem um caminho eficiente e confiável em relação à mudanças futuras no ambiente urbano (e.g., mobilidade, congestionamento, e segurança) [Taha and AbuAli 2018]. Diante desse cenário, com a grande quantidade de dados disponíveis juntamente com inter-correlação desses dados, a LSTM pode melhorar substancialmente a predição da dinâmica futura dos fatores urbanos explorando longas sequências de dados espaço-temporais (e.g., histórico). Assim, permitindo serviços mais confiáveis e eficiente além de melhorar serviços de planejamento urbano.

VTq implementa uma LSTM que recebe como entrada o histórico dos fatores urbanos de cada via. Considere $X = \{x_0, x_1, \dots, x_m\}$ um vetor que representa um conjunto de entrada, no qual cada elemento $x_i \in X \mid 0 \leq i \leq m$ consiste em um tupla $x_i = \langle timestamp, uv, valor \rangle$, representando data e horário para cada valor estimado de um fator urbano (e.g., tempo de viagem, congestionamento, consumo de combustível, segurança, e etc). Dessa forma, a predição da LSTM é definida como $f = \Psi \circ \Phi$, onde \circ indica que a função Ψ é aplicada à saída da função Φ . Primeiramente, os dados de entrada passam pela função de extração de características $\Phi(\cdot)$, a qual transforma os dados de entradas em características. Em seguida, a função de representação $\Psi(\cdot)$ é utilizada para mapear as características em uma predição [Liu et al. 2017b]. Assim, o processo de predição da dinâmica futura de cada fator urbano é apresentado por:

$$x_{t+1} = \Psi \circ \Phi(X_t) \quad (1)$$

onde, x_{t+1} representa a próxima dinâmica de uma via urbana considerando seu histórico X_t .

3.4. Planejamento de Rota baseado em Aprendizado por Reforço

O motivo principal para a eficiência limitada das soluções para planejamento de rota é a falta de conhecimento de quando e/ou onde um congestionamento irá acontecer, como também quanto tempo o mesmo irá durar. Assim, essas soluções podem guiar veículos para vias urbanas que ficarão congestionadas no futuro próximo [Taha and AbuAli 2018, Pan et al. 2017]

Nesse cenário, como a assistência da LSTM para fornecer um melhor entendimento da dinâmica urbana futura, novas soluções podem ser propostas para fornecer um planejamento

de rota mais robusto. Assim, propomos o uso de um algoritmo de aprendizado por reforço para aprender a planejar a rota mais eficiente baseado na preferência de cada usuário juntamente com a dinâmica futura do ambiente urbano. Portanto, baseado na origem e destino o algoritmo de aprendizado por reforço planeja maximizar/minimizar a recompensa (e.g., rota mais rápida, mais segura, com menos congestionamentos, etc) considerando a variação da dinâmica futura previamente predita pela LSTM.

O algoritmo de aprendizado por reforço é baseado em *Q-learning*. Em resumo, ele é composto por: (i) agente; (ii) ambiente; (iii) estado; (iv) ação; e (v) recompensa. Assim, o algoritmo aprende com a interação entre agente e ambiente, onde a partir de um estado, o agente pode tomar um conjunto de ações e para cada ação é gerada uma recompensa (positiva ou negativa) e um novo estado. Assim, o objetivo desse algoritmo é maximizar/minimizar a recompensa final. O algoritmo 1 apresenta o processo de aprendizado por reforço executado pelo veículo (e.g., o agente) para determinar qual via urbana será selecionada para compor sua rota (e.g., a ação) baseado não só no estado corrente do ambiente, mas também baseado na política de aprendizado (e.g., tempo de viagem, segurança, etc) e recompensa esperada de acordo com a predição da dinâmica futura.

O algoritmo recebe como entrada: (i) a taxa de aprendizado usada no algoritmo de *Q-learning*; (ii) o grafo representando o cenário urbano $G = (V, E)$; (iii) o conjunto de predição da dinâmica futura X_t para o próximo intervalo Δ_t ; e (iv) o par de origem e destino do veículo. Assim, o veículo pode criar a tabela-Q e a matriz de recompensa R para cada intervalo de tempo $t \in \Delta_t$ baseado no grafo G . Cada matriz de recompensa R_t é uma matriz de adjacência com os valores preditos de cara via urbana no tempo t , enquanto a tabela-Q é usada para armazenar o planejamento de rota mais eficiente. Dessa forma, o veículo define o estado atual S_i como sua posição corrente e t como o tempo para iniciar sua rota.

Para planejar a rota em direção ao destino, o veículo verifica as ações \mathcal{A} disponíveis no tempo t baseado no estado S_i e na matriz de recompensa R_t . Em seguida o veículo seleciona uma ação $a_t \in \mathcal{A}_t$ no tempo t , e então, atualiza a tabela-Q de acordo com o estado S_i , com a ação selecionada a_t e recompensa esperada R_t no tempo t . Portanto, a função de atualização do algoritmo *Q-learning* pode ser formulada como a seguinte função iterativa para maximizar a recompensa esperada

$$Q_{t+1}(S_i, a_t) = (1 - \alpha) \cdot Q_t(s_i, a_t) + \alpha \left[R_t + \gamma \max_{a_t \in \mathcal{A}_t} Q_t(S_i, a_t) \right] \quad (2)$$

onde α é a taxa de aprendizado e γ é o fator de desconto. Por fim, o veículo atualiza seu estado S_i como o novo estado $S_i + a_t$ e t com o tempo para executar a ação a_t no estado S_i . Tal processo é executado até Q convergir. Sendo assim, para seguir o planejamento de rota calculado, o veículo seleciona a via urbana com maior recompensa no tempo t , quando o mesmo atingir tal via baseado na matriz de recompensa calculada.

3.5. Visão geral do Sistema

A Figura 1 apresenta uma visão geral da arquitetura e funcionamento do VTq. Primeiramente, quando um veículo for iniciar seu trajeto, uma requisição da predição dos fatores desejados é realizada para o servidor de borda responsável pela região em que o veículo está. Dessa forma, os dados da última predição são retornados para o veículo utilizando a LSTM localizada no servidor de borda. Esses dados são representados por uma lista de tuplas contendo a via urbana $uv \in E$ e os valores x_{uv}^t . Assim, ao receber essas informações o veículo atualiza os valores do grafo que representa o cenário e o algoritmo de aprendizado por reforço (Algoritmo 1) planeja a rota. Para cada via $uv \in E$ a ser selecionada (e.g., ação) no tempo t , uma recompensa é gerada baseado no fator urbano desejado x_{uv}^t , assim para cada caminho construído entre

Algorithm 1: Planejamento de rota utilizando aprendizado por reforço

Input : α // Taxa de aprendizado para o algoritmo de Q-learning
 $G = (V, E)$ // Grafo representando o cenário urbano
 X_t // Conjunto que representa a variação futura na dinâmica urbana
origem // Origem da rota
destino // Destino da rota

Output: Tabela-Q para construir o caminho mais eficiente considerando a dinâmica futura do ambiente urbana

```
1  $R_{0, \dots, \max(t)} \leftarrow \text{constróiMatrizRecompensa}(G, X, \Delta_t)$ ;  
2  $Q \leftarrow \text{constróiTabelaQ}()$ ;  
3 while  $Q$  não convergir do  
4    $S_i \leftarrow \text{origem}$ ;  
5    $t \leftarrow \text{tempoParaIniciarViagem}()$ ;  
6   while  $S_i \neq \text{destino}$  do  
7      $A_t \leftarrow \text{getAcoesDisponiveis}(S_i, R_t)$ ;  
8      $a_t \leftarrow \text{selecionaAção}(A_t)$ ;  
9      $Q.\text{atualiza}(S_i, a_t, \gamma, R_t)$ ;  
10     $S_i \leftarrow S_i + a_t$ ;  
11     $t \leftarrow t + \text{tempoParaExecutar}(a_t, S_i)$ ;  
12  end  
13 end
```

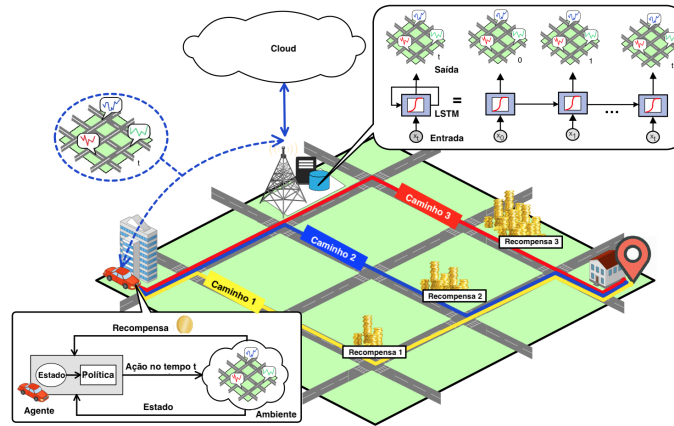


Figura 1. Visão geral do sistema apresentando arquitetura e interação entre LSTM e planejamento de rota.

origem e destino, uma recompensa total com o custo da rota $c(P) = \sum_{uv \in P} x_{uv}^t$ é obtida, assim a rota mais eficiente é aquela com a maior recompensa possível. É importante salientar que o compartilhamento dessas informações entre veículo e servidor de borda não introduz latência nem reduz a eficiência do sistema devido aos protocolos para compartilhamento e requisição de informações implementados pelo VTq (mais detalhes sobre esse processo pode ser encontrado em [de Souza et al. 2017, de Souza et al. 2018]).

4. Análise dos Resultados

Essa seção apresenta os resultados da avaliação do VTq em relação à precisão das previsões de mobilidade e de segurança, e também em relação à qualidade do planejamento de rota baseado nas preferências (mobilidade e segurança) de cada usuário. Portanto, a Seção 4.1 descreve as bases de dados utilizadas para validar as previsões da LSTM. A Seção 4.2 apresenta as ferramentas e metodologia utilizada para a validação do VTq. A Seção 4.3 descreve os resultados das previsões do VTq em relação às soluções para previsão de dados temporais utilizadas pela literatura, enquanto a Seção 4.4 apresenta a avaliação da performance do planejamento de rota por aprendizado por reforço. Por fim, a Seção 4.5 compara os resultados do planejamento de rota produzido pelo VTq em relação às soluções da literatura.

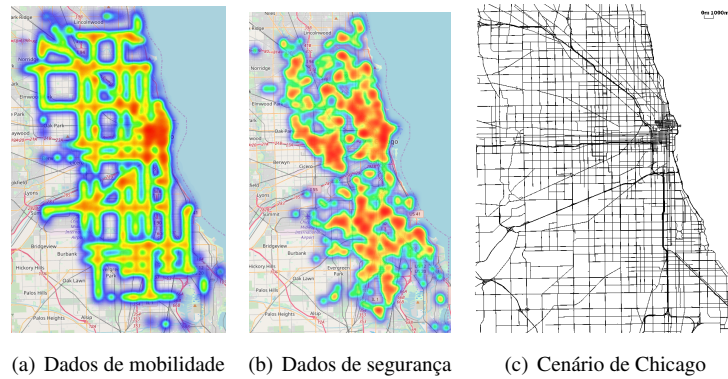


Figura 2. Exemplo das informações das base de dados utilizada e do cenário.

4.1. Base de dados

Para avaliar a eficiência das previsões e do planejamento, utilizamos dados reais de mobilidade e segurança pública da cidade de Chicago, EUA. Esses dados foram obtidos através do portal de dados abertos de Chicago¹. Os dados encontram-se entre Fevereiro de 2018 até Dezembro de 2019. A Figura 2 mostra um exemplo dos dados utilizados para o dia 20 de Outubro de 2019 tanto para mobilidade quanto para segurança.

Base de dados de mobilidade urbana: é composta por aproximadamente 1000 vias urbanas com estimativas de tráfego (e.g., velocidade média corrente de cada via) a cada 5 minutos. Cada estimativa é baseada em informações coletadas em tempo real dos *probes* de GPS dos ônibus da cidade.

Base de dados de segurança: composta pelo conjunto de crimes que aconteceram no período avaliado. Para anualizar as mesmas vias presentes na base de dados de mobilidade urbana, foi utilizado as coordenadas de início e fim das interseções de cada via presente na base de dados de mobilidade. Assim, baseado no conjunto de crimes dentro de uma janela temporal de 30 minutos, a densidade criminal de cada via urbana foi calculada utilizando uma função de densidade Gaussiana (KDE). Seja C o conjunto de crimes que aconteceram na cidade durante a janela temporal definida. Assim, a densidade criminal de cada via urbana $uv \in E$ é calculada da seguinte forma:

$$\theta(p) = \frac{1}{|C|} \cdot \sum_{c \in C} \frac{1}{h\sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{\|c-p\|}{h} \right)^2} \quad (3)$$

onde $\|c - p\|$ é a distância Euclidiana entre os pontos p e c e h é o hiper-parâmetro do algoritmo. O parâmetro h define a dispersão da gaussiana e controla a suavidade da densidade, tal parâmetro foi definido baseado em [Scott 1992]. Uma vez que a função de densidade é estimada, a densidade criminal de cada via $uv \in E$ é calculada baseado na média da densidade de $\theta(u) + \theta(v)$.

4.2. Metodologia

A plataforma de avaliação e simulação é composta pelo simulador de mobilidade urbana (SUMO² — *Simulation of Urban Mobility*), versão 0.30.0, o simulador de rede OMENet++³ versão 5.0 e para rede veicular foi utilizado o Veins⁴ versão 4.6. Além disso, foi utilizado o *framework* de *deep learning* TensorFlow versão 1.12 para implementação da LSTM, enquanto

¹<https://data.cityofchicago.org/>

²<http://sumo.dlr.de/>

³<http://omnetpp.org/>

⁴<http://veins.car2x.org>

para implementação do algoritmo de aprendizado por reforço e os outros preditores foi utilizado o SciKit Learn versão 0.20.2. O cenário urbano representa 50 km² da cidade de Chicago, o qual foi obtido através do OpenStreetMaps⁵ (veja Figura 2(c)). Para obter o mesmo perfil de mobilidade, as velocidades médias das vias do cenário são atualizadas de acordo com as informações fornecidas pelas bases de dados.

Para análise da predição dos fatores urbanos, foi usado uma divisão de dados de treino e dados de teste com 75% dos dados para treino e 25% dos dados para teste utilizando uma *cross-validação* de 10-*folds*. A rede implementada possui duas camadas LSTM com 128 células cada uma, e uma camada *Densa* com distribuição temporal para prever o valor de saída. Além disso, a rede neural foi treinada com 100 *epochs*, com *batches* de tamanho 128 e um *unroll sequence* de tamanho 24. Por outro lado, para o algoritmo de *Q-learning* foi utilizado um coeficiente de aprendizado (α) igual a 0.01 e um fator de desconto (γ) igual a 1, para priorizar as recompensas futuras. Tais parâmetros foram definidos baseado em uma análise exploratória e obtiveram performance superior aos outros parâmetros avaliados.

Cada predição representa a dinâmica da mobilidade e da segurança do cenário para a próxima hora com uma granularidade de 5 minutos. Sendo assim, para a análise da mobilidade as predições representam a velocidade média de cada via urbana, enquanto para as predições de segurança cada predição representa a densidade criminal (descrita pela Equação 3) de cada via.

4.3. Avaliação da Predição da Dinâmica Urbana Futura: Mobilidade e Segurança

Para analisar a performance das predições realizadas pelo VTq comparamos seu desempenho com os seguintes preditores (i) **RL**: preditor que utiliza uma regressão linear para prever o próximo valor na série temporal; (ii) **CNN**: preditor que utiliza uma rede neural com convolução para realizar a predição dos dados futuros da série temporal; e (iii) **LSTM**: preditor baseado em redes neurais recorrentes implementado pelo VTq para prever os valores futuros da série temporal.

Para a avaliação dos preditores foi utilizado o erro médio quadrado (MSE), erro médio absoluto (MAE), e o coeficiente de correlação, os quais são as métricas utilizadas para uma análise de regressão. A Tabela 1 apresenta os resultados de cada métrica para cada preditor.

Como pode ser observado, a LSTM implementada pelo VTq apresenta melhores resultados em todas as métricas avaliadas para ambas base de dados. Em particular, a LSTM apresenta uma redução no MSE e MAE em pelo menos 75% e 50%, respectivamente. Além disso, a LSTM aumenta a correlação em pelo menos 27% quando comparado com as outras soluções. Entretanto, devido a falta de precisão nas predições, as soluções RL e CNN podem introduzir falsos positivos e falsos negativos durante o planejamento de rota, ou seja, veículos podem ser direcionados a vias congestionadas ou perigosas devido a erros nas predições.

Tabela 1. Resultados das métricas avaliadas para cada solução com os dados de mobilidade e segurança

Preditores	Mobilidade			Segurança		
	MSE	MAE	Correlação	MSE	MAE	Correlação
RL	0.0053	0.0510	0.75	0.0075	0.0659	0.71
CNN	0.0055	0.0522	0.75	0.0074	0.0674	0.71
LSTM	0.0012	0.0234	0.96	0.0019	0.0334	0.93

O desempenho superior apresentado pela LSTM é resultado da sua capacidade de não só explorar a inter-correlação de dados espaço-temporais, mas também por ser capaz de lembrar

⁵<https://www.openstreetmap.com>

e aprender longas séries de dados. Os resultados apresentados pela LSTM mostram que ela pode ser uma solução eficiente para fornecer informações sobre a dinâmica urbana futura para serviços de planejamento de rota, pois mesmo em previsões erradas, o erro introduzido pela LSTM é muito baixo, conseqüentemente não degradando a eficiência do planejamento. Isso pode ser visto na Figura 3, a qual mostra os resultados das previsões para os dados de segurança comparando todas as soluções. O eixo y apresenta os valores preditos pela solução, enquanto o eixo x apresenta os valores reais. Portanto, quanto mais espalhados os pontos piores são as previsões e maior é o erro introduzido. É importante salientar que resultados de mobilidade foram omitidos pois a performance das soluções foram semelhantes em ambas as base de dados.

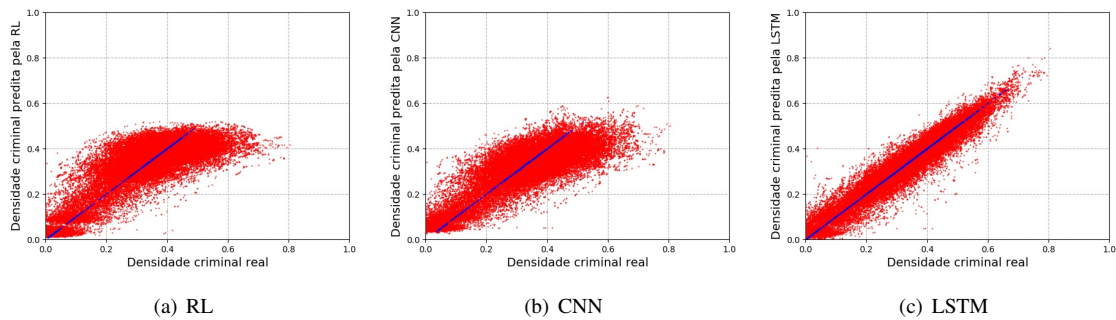


Figura 3. Resultado da dispersão das previsões: previsões (eixo y) vs. dados reais (eixo x)

A Figura 4 apresenta os resultados das previsões de segurança para um conjunto de 170 previsões de Outubro de 2019 para cada solução. Como pode ser observado, as soluções RL e CNN não são capazes de expressar a dinâmica da segurança adequadamente. Além disso, esse comportamento deixa claro a introdução de falsos positivos e falso negativos (veja Figuras 4(a) e 4(b) os valores preditos muito maiores/menores que os valores reais). Por outro lado, a LSTM é capaz de expressar a mesma dinâmica sem a introdução de falsos positivos e falsos negativos (veja Figura 4(c)).

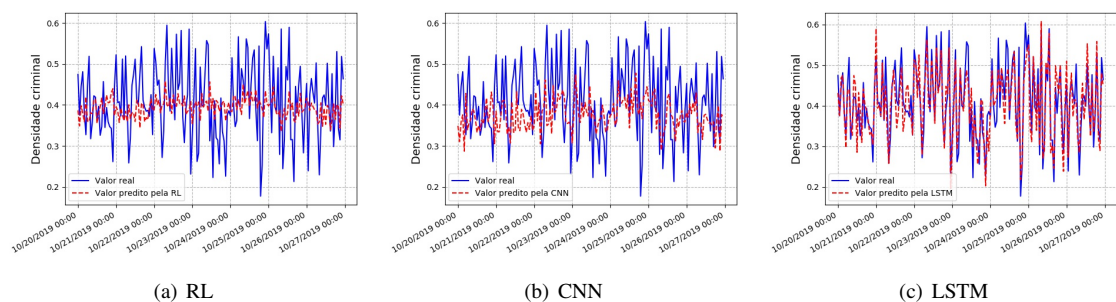


Figura 4. Resultados da comparação do dinâmica predita em relação a dinâmica real do cenário

4.4. Avaliação do Planejamento de Rota Eficiente baseado em Aprendizado por Reforço

Para a análise do planejamento de rota, um conjunto de 1000 rotas foram planejadas utilizando dias e horários selecionados aleatoriamente do conjunto de teste das base de dados de mobilidade e segurança. Cada rota consiste em uma origem e um destino selecionado aleatoriamente do conjunto de vias disponíveis na base de dados. Foi considerado o tempo de viagem (e.g., razão entre velocidade média da via e seu comprimento obtidos pela base de dados e cenário) e a segurança como métricas (preferência) para o planejamento de rota. Ambas as métricas representam a soma total do custo do caminho (e.g., $c(P)$).

Um fator importante que deve ser avaliado é a latência introduzida pelo algoritmo de planejamento de rota, uma vez que para garantir o planejamento ótimo entre uma origem e destino, todos os caminhos possíveis juntamente com a sua variação temporal devem ser explorados. Entretanto, o problema de explorar todos os caminhos entre dois vértices foi provado ser NP-completo, tendo complexidade $O(V!)$ [Joksch 1966]. Sendo assim, para garantir um planejamento em tempo real limitamos a exploração de caminhos do algoritmo de planejamento (e.g., as iterações para convergir o *Q-learning*). A quantidade de caminhos a serem explorados foi definido como 10^3 , 10^4 , 10^5 e 10^6 . Além disso, para cada exploração foi calculado o tempo de CPU utilizando um processador Intel Core i5-3330 de 3.0 GHz.

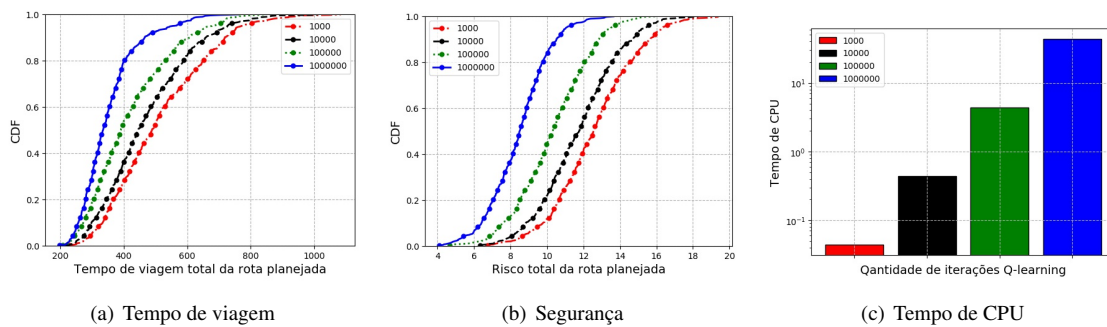


Figura 5. Resultados do planejamento de rota do VTq em relação a quantidade de iterações do algoritmo.

A Figura 5 apresenta os resultados para cada exploração implementada no algoritmo de planejamento. Os resultados do tempo de viagem e segurança estão representados por uma função de distribuição acumulada, enquanto os resultados do tempo de CPU estão representados pelo valor médio.

Como esperado, quanto menor o espaço de exploração pior é o planejamento da rota. Dessa forma, quanto mais opções de caminhos para serem exploradas melhor é o aprendizado do algoritmo e o planejamento de rota. Especificamente com exploração igual 10^6 , VTq reduz o tempo de viagem em 40% para 90% dos veículos, enquanto melhora em 35% a segurança de 90% veículos quando comparada com os resultados da exploração de 10^3 (veja Figura 5(a) e Figura 5(b)).

Por outro lado, a quantidade de caminhos a serem exploradas também impacta no tempo de CPU do algoritmo, assim, quanto mais opções maior o tempo de CPU. Porém, mesmo para o maior tempo de CPU, VTq apresenta um tempo de planejamento inferior a 40 segundos, o que o torna uma solução viável para o planejamento de rotas. É importante destacar que o planejamento é executado antes do veículo iniciar a viagem, portanto, esse tempo de CPU não se torna uma limitação do sistema. Além disso, um *trade-off* entre tempo e performance do planejamento pode ser definido, por exemplo, a exploração com 10^5 iterações fornece um bom planejamento com tempo de CPU inferior a 3 segundos, fornecendo um bom *trade-off* entre mobilidade e tempo de CPU.

A Figura 6 mostra a comparação do planejamento de rota utilizando os valores preditos pela LSTM e em relação ao planejamento de rota com os dados reais. Com essa avaliação é possível observar a degradação do planejamento de rota por previsões erradas além de permitir a identificação de falsos positivos. Contudo, os resultados mostram que para ambos os planejamentos VTq possui um desempenho muito semelhante independente dos dados utilizado. Portanto, esses resultados mostram que as previsões realizadas pela LSTM não degradam a eficiência do planejamento de rota empregado pelo VTq e nem introduzem falsos positivos e falsos negativos nesse processo.

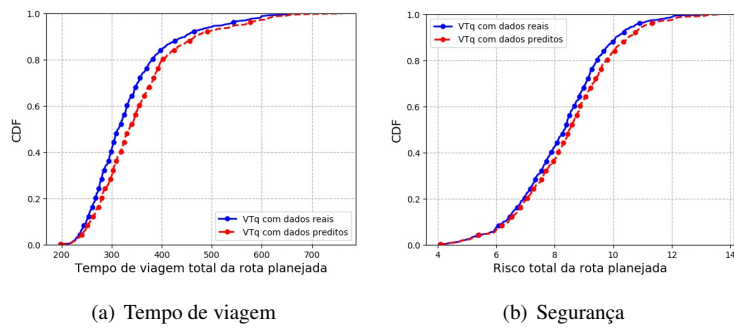


Figura 6. Comparação entre o planejamento de rota com dados reais e dados preditos.

4.5. Comparação do Planejamento de Rotas: VTq vs Trabalhos Relacionados

Para avaliar a performance do VTq, seu desempenho foi comparado com soluções do estado da arte da literatura para planejamento de rotas baseado em mobilidade e segurança, sendo elas: (i) DIVERT [Pan et al. 2017]; (ii) SafePaths [Galbrun et al. 2016]; e (iii) #PAS [de Souza et al. 2018]. Nessa avaliação as métricas de tempo de viagem e risco à segurança foram utilizados para o planejamento de rota além do tempo de CPU do algoritmo de roteamento.

A Figura 7 apresenta a comparação do desempenho do VTq em relação as soluções da literatura. Em particular a Figura 7(a) apresenta os resultados do tempo de viagem, enquanto as Figuras 7(b) e 7(c) apresentam os resultados de risco à segurança e tempo de CPU, respectivamente.

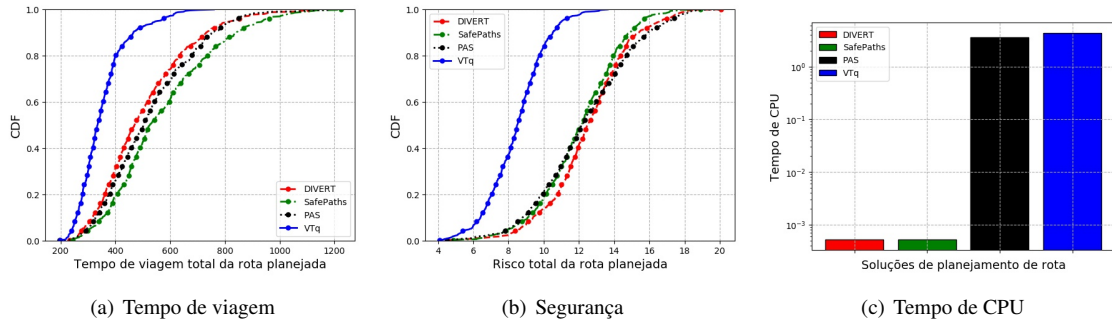


Figura 7. Resultado da comparação do planejamento de rota entre VTq e soluções da literatura.

SafePaths faz o planejamento de rota baseado na rota mais segura, enquanto DIVERT faz o planejamento considerando o tempo de viagem nas vias e o #PAS explora um *trade-off* entre mobilidade e segurança. Sendo assim, DIVERT tem um tempo de viagem menor que #PAS e SafePaths, e possui os piores resultados de segurança. SafePaths fornece uma segurança maior aos motoristas e passageiros quando comparado com #PAS e DIVERT, porém não apresenta um tempo de viagem satisfatório para os usuários. Por fim, #PAS não é penalizado em nenhuma métrica por explorar o *trade-off* entre mobilidade e segurança.

Entretanto, apesar dessas soluções serem desenvolvidas para lidar com os problemas de mobilidade e segurança, nenhuma delas considera a dinâmica futura desses fatores durante o planejamento, conseqüentemente, a rota planejada potencialmente pode deixar de ser a mais eficiente em um futuro próximo, assim degradando a eficiência das rotas planejadas.

Os benefícios de se considerar a variação da dinâmica futura dos fatores urbanos podem ser vistos comparando o desempenho do VTq em relação as outras soluções. Pois, mesmo com soluções específicas para minimizar problemas de mobilidade e segurança, elas não são

capazes de planejar rotas eficientes devido a dinamicidade do ambiente urbano.

Em particular, VTq é capaz de melhorar a mobilidade em pelo menos 50% para 90% dos veículos, além de melhorar a segurança em pelo menos 30% para 90% dos veículos quando comparados com as soluções para planejamento de rotas baseadas em mobilidade e segurança que não consideram a dinâmica urbana. Além disso, VTq possui um tempo de CPU (com 10^5 iterações) equivalente a solução #PAS (veja Figura 7(c)) que mostrou ser uma solução que não introduz uma latência indesejada no sistema além de ser escalável [de Souza et al. 2018]. Por outro lado, as soluções DIVERT e SafePaths apresentam um tempo de CPU muito inferior em relação ao VTq e #PAS, o que é resultado do algoritmo de caminhos mínimos que possui uma complexidade inferior quando comparado com as demais soluções. Em particular, DIVERT e SafePaths possuem complexidade $O(E + V \log V)$, enquanto o #PAS possui um complexidade de $O(E \cdot \lambda)$, onde λ é o risco máximo de segurança aceitável para a rota, e por fim, VTq apresenta uma complexidade de $O(K \cdot C)$, onde K é a quantidade de caminhos explorados e C é o custo para encontrar cada caminho, onde C é pelo menos $E + V \log V$.

É importante esclarecer que as soluções SafePaths, DIVERT e #PAS podem fornecer a mesma performance que o VTq, mas, para isso, um novo planejamento teria que ser feito quando o veículo chegasse ao final de cada via de sua rota previamente planejada. Porém, isso seria potencialmente impraticável pois a complexidade do sistema aumentaria devido a grande quantidade de mensagens trocadas entre servidor e veículos (para obter conhecimento atualizado sobre a dinâmica urbana) além de reduzir a qualidade da experiência de direção, uma vez que o motorista teria que trocar de rota muito frequentemente.

Uma questão importante é a análise do *overhead* produzido pelo sistema para fornecer informações de mobilidade e obter o conhecimento para o planejamento de rota. Entretanto, por limitações de espaço tal análise foi omitida, mas como VTq é baseado em nosso trabalho anterior (e.g., utiliza a mesma arquitetura e protocolos para obtenção e compartilhamento de conteúdo) uma análise detalhada da performance de rede, *overhead* e escalabilidade pode ser encontrada em [de Souza et al. 2018].

5. Conclusão

Nesse trabalhos apresentamos VTq, um ITS para planejamento de rota eficiente considerando a dinâmica urbana futura baseado em *deep learning* e computação de borda. Para isso, VTq extrai conhecimento sobre os fatores urbanos e realiza previsões da dinâmica futura no ambiente urbano com uma LSTM considerando as informações atuais e seu histórico. Além disso, baseado nas previsões fornecidas pela LSTM, VTq implementa um algoritmo de planejamento de rota eficiente baseado em aprendizado por reforço. O algoritmo de planejamento de rota mostrou ser adaptativo, obtendo um desempenho substancial para diversas métricas que podem ser personalizadas pelo usuário. Para validação dos resultados foram utilizados dados reais da cidade de Chicago, onde os resultados mostraram que: (i) a LSTM fornece um conhecimento preciso sobre a dinâmica urbana; (iii) a computação de borda juntamente com o aprendizado de máquina são capazes de potencializar soluções para melhorar o gerenciamento e planejamento do ambiente urbano; e (ii) o algoritmo de aprendizado por reforço é capaz planejar rotas mais eficientes considerando a dinâmica urbana futura. Como trabalhos futuros pretendemos implantar o algoritmo de *deep Q-learning* para melhorar a performance do algoritmo de planejamento além de técnicas convoluções de grafos em redes neurais recorrentes para melhorar ainda mais as previsões produzidas pelo sistema.

Agradecimentos: Os autores gostariam de agradecer ao apoio financeiro da Fundação de Amparo à Pesquisa (FAPESP) concedido pelo projeto (processo #2018/19639-5) e pela bolsa de pesquisa (processo #2019/24937-8).

Referências

- Amirgholy, M., Golshani, N., Schneider, C., Gonzales, E. J., and Gao, H. O. (2017). An advanced traveler navigation system adapted to route choice preferences of the individual users. *International Journal of Transportation Science and Technology*, 6(4):240–254.
- Campolo, C., Molinaro, A., Iera, A., and Menichella, F. (2017). 5g network slicing for vehicle-to-everything services. *IEEE Wireless Communications*, 24(6):38–45.
- de Souza, A. M., Botega, L. C., Garcia, I. C., and Villas, L. A. (2018). Por aqui é mais seguro: Melhorando a mobilidade e a segurança nas vias urbanas. *Anais do Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*, 36.
- de Souza, A. M., Braun, T., Botega, L. C., Villas, L. A., and Loureiro, A. A. F. (2019). Safe and sound: Driver safety-aware vehicle re-routing based on spatiotemporal information. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–17.
- de Souza, A. M., da Fonseca, N. L. S., and Villas, L. (2017). A fully-distributed advanced traffic management system based on opportunistic content sharing. In *2017 IEEE International Conference on Communications (ICC)*, pages 1–6.
- de Souza, A. M., Yokoyama, R. S., Maia, G., Loureiro, A. A. F., and Villas, L. A. (2015). Minimizing traffic jams in urban centers using vehicular ad hoc networks. In *2015 7th International Conference on New Technologies, Mobility and Security (NTMS)*.
- Dong, P., Zheng, T., Yu, S., Zhang, H., and Yan, X. (2017). Enhancing vehicular communication using 5g-enabled smart collaborative networking. *IEEE Wireless Communications*, 24(6):72–79.
- Doolan, R. and Muntean, G. M. (2017). EcoTrec: A Novel VANET-Based Approach to Reducing Vehicle Emissions. *IEEE Transactions on Intelligent Transportation Systems*, 18(3):608–620.
- Galbrun, E., Pelechris, K., and Terzi, E. (2016). Urban navigation beyond shortest route: The case of safe paths. *Information Systems*, 57:160–171.
- Joksch, H. (1966). The shortest route problem with constraints. *Journal of Mathematical Analysis and Applications*, 14(2):191 – 197.
- Liu, J., Wan, J., Jia, D., Zeng, B., Li, D., Hsu, C., and Chen, H. (2017a). High-efficiency urban traffic management in context-aware computing and 5g communication. *IEEE Communications Magazine*, 55(1):34–40.
- Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., and Alsaadi, F. E. (2017b). A survey of deep neural network architectures and their applications. *Neurocomputing*, 234:11 – 26.
- Pan, J. ., Popa, I. S., and Borcea, C. (2017). Divert: A distributed vehicular traffic re-routing system for congestion avoidance. *IEEE Transactions on Mobile Computing*.
- Qiu, C., Zhang, Y., Feng, Z., Zhang, P., and Cui, S. (2018). Spatio-temporal wireless traffic prediction with recurrent neural network. *IEEE Wireless Communications Letters*, 7(4):554–557.
- Scott, D. W. (1992). *Multivariate density estimation: theory, practice and visualization*. John Wiley Sons, New York, Chicester.
- Taha, A. and AbuAli, N. (2018). Route planning considerations for autonomous vehicles. *IEEE Communications Magazine*, 56(10):78–84.
- Ye, H., Liang, L., Li, G. Y., Kim, J., Lu, L., and Wu, M. (2018). Machine learning for vehicular networks: Recent advances and application examples. *IEEE Vehicular Technology Magazine*, 13(2):94–101.