Aproximando grafos das ruas de cidades às de cidades planejadas com Graph Machine Learning

Daniel Souza de Campos[1]

[1]Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG)

danielcampos@dcc.ufmg.br

Abstract. *Street networks are critical to local and national transport. In recent decades, there has been a movement towards the expansion of the world's urban mesh, which has shifted from expansion through grid layouts to longer and less connected streets. Because they are naturally modeled as graphs, cities have attracted attention from the field of Graph Machine Learning. Thus, the objective of the work is to try to suggest new streets based on the inherent structure of planned cities using machine learning for graphs, an unexplored task. Experiments showed that the combination of Node2Vec with LightGBM applied on the proposed division of planned and not planned cities did not have satisfactory results.*

Resumo. *Redes de ruas são críticas para o transporte local e nacional. Nas últimas décadas, existiu um movimento de expansão da malha urbana mundial que mudou de expansão por meio de disposições em grade para ruas mais longas e menos conectadas. Por serem naturalmente modeladas como grafos, as cidades tem atraído atenção da área de Graph Machine Learning. Assim, o objetivo do trabalho é tentar sugerir novas ruas baseadas na estrutura inerente de cidades planejadas utilizando de aprendizado de máquina para grafos, uma tarefa ainda não explorada na literatura. Experimentos mostraram que a combinação do Node2Vec com LightGBM aplicada sobre a divisao de cidades planejadas e não planejadas proposta não teve resultados satisfatórios.*

1. Introdução

Redes de ruas são críticas para o transporte local e nacional. Ter uma infraestrutura bem planejada é um grande diferencial para a economia e bem estar geral da população de uma cidade[Burghardt et al. 2022]. Pode-se dizer que a disposição atual das ruas é um reflexo de padrões de desenvolvimento fomentados por necessidades históricas, limitações geográficas e políticas de expansão [Burghardt et al. 2022, Uhl et al. 2022].

Nas últimas décadas, existiu um movimento de expansão da malha urbana mundial que mudou de expansão por meio de disposições em grade(*grid*) para ruas mais longas e menos conectadas. O primeiro tipo aumenta a trafegabilidade, mobilidade e acessibilidade [Hu et al. 2008] e é característica de cidades bem planejadas [Rui et al. 2013]. O segundo aumenta a necessidade de automóveis particulares [Barrington-Leigh and Millard-Ball 2020].

Existem vários estudos com relação a organização das ruas de uma cidade. Alguns focam na análise da expansão da malha urbana ao longo dos anos até onde a existência de

dados permite [Burghardt et al. 2022] [Barrington-Leigh and Millard-Ball 2020]. Já outros tentam resgatar como se deu essa expansão ao completar dados comparando mapas de várias épocas [Uhl et al. 2022]. Diversos trabalhos tentam identificar quais são as principais vias de uma cidade do ponto de vista da teoria de redes complexas [Caoa et al. 2016], transporte de informações[Scellato et al. 2006], robustez [Masucci and Molinero 2016], vulnerabilidade [Jenelius and Mattsson 2015][Li and Liu 2020] e trafegabilidade[Hu et al. 2008].

Dessa forma, dada a importância geral de ruas para as cidades, recentemente surgiu o interesse da comunidade na aplicação e desenvolvimento de modelos de *Machine Learning* para ajudar a entender peculiaridades da estrutura da rede de ruas das cidades. Mais especificamente, ao modelar a rede de ruas como um grafo, é possível usar de *Graph Machine Learning* (GML) para tentar extrair informações que possam ser difíceis de conseguir via métricas propostas manualmente. Algumas tarefas envolvendo GMLs e rede de ruas de cidades que já foram exploradas são: estimar limites de velocidade em ruas [Jepsen et al. 2018, Jepsen et al. 2022], categorizar tipos de ruas [Jepsen et al. 2018, Gharaee et al. 2021, Wang et al. 2020], similaridade de trajetórias [Zhang et al. 2020] e quantificação da homogeneidade espacial de sub-redes de ruas [Xue et al. 2022].

O objetivo desse trabalho é usar a classificação binária de cidades realizada no POC1 como dados de treino para modelos de GML capazes de sugerir novas ruas. Assim, as estruturas de outras cidades poderiam se aproximar da estrutura inerente das cidades consideradas planejadas. Essa é uma tarefa ainda não explorada na literatura.

O trabalho está dividido na seguinte forma: Na Seção 2 é apresentado o referencial teórico junto com a revisão dos resultados da POC1 que serviram de base para este trabalho. A Seção 3 apresenta a Metodologia seguida. A Seção 4 apresenta a formalização do problema e os experimentos realizados. A Seção 5 finaliza o trabalho.

2. Referencial Teórico

Nesta seção, serão revisados os resultados da POC1 que serviram como base para este trabalho, ou seja, a agregação e classificação manual de um *dataset* de cidades de acordo com as métricas encontradas durante a POC1. Além disso, também serão apresentados os conceitos de *GML* e as pesquisas relevantes que guiaram a seção de experimentação.

2.1. Revisão da POC1

Com os estudos durante o POC1 sobre a expansão de cidades, ficou entendido que elas se expandem devido, principalmente, ao seu processo de urbanização [Angel 2012]. Assim, o crescimento da rede de ruas segue o crescimento populacional para permitir o transporte de cargas e pessoas [Strano et al. 2012, Barthelemy et al. 2013]. Dessa forma, como é esperado um grande crescimento populacional em áreas urbanas nas próximas décadas [Angel 2012], é necessário que as cidades se preparem para evitar problemas como congestionamento e oferecimento de serviços públicos de má qualidade. Esse preparo também é capaz de evitar correções futuras caracterizadas por grandes obras caras [Barthelemy et al. 2013]. Dessa forma, a decisão de onde e quando estabelecer novas ruas em uma cidade é muito importante pois ela, provavelmente, definirá o traçado de transporte de pessoas pela cidade por muitos séculos [Floater et al. 2014].

As cidades podem ser modeladas naturalmente como um grafo. Existem dois tipos de modelagem principais. A forma primal representa cruzamentos como nós e ruas como arestas e a forma dual representa ruas como nós e cruzamentos como arestas [Agryzkov et al. 2017]. Sob o olhar de redes complexas, a forma primal pode ser vista como uma rede aleatória já que o grau dos nós desses grafos variam pouco ao redor de um centro. Já as redes duais podem ser consideradas *scale-free* pois a distribuição dos graus dos seus nós mostra que existem muito mais nós com grau baixo do que grau alto [Porta et al. 2004, Xie and Levinson 2007]. Existem vantagens e desvantagens de usar uma dessas modelagens [Porta et al. 2004, Porta et al. 2005] e, portanto, é necessário ter cuidado na hora de aplicar métricas de qualidade sobre o grafo.

Existem muitas métricas que podem ser computadas sobre um grafo a fim de caracterizá-lo. Muitas já foram usadas e criadas para analisar os grafos primais e duais das cidades[Boeing 2018, Boeing 2021, Xie and Levinson 2007, Angel 2012, Angel et al. 2021, UNH 2016, Courtat et al. 2011, Porta et al. 2005, Burghardt et al. 2022, Barrington-Leigh and Millard-Ball 2019]. No POC1, foram encontradas 108 métricas que poderiam ajudar a caracterizar o grafo que representa uma cidade. Apesar de toda essa coleção de métricas, existem indícios de que a classificação de uma cidade como planejada está mais relacionada com a conectividade e eficiência de suas ruas [Barrington-Leigh and Millard-Ball 2019, Barrington-Leigh and Millard-Ball 2020, Burghardt et al. 2022].

Dessa forma, foram escolhidas 5 métricas dentre as 108 identificadas para sugerir dois conjuntos de cidades: planejadas e não planejadas. As métricas que basearam essas classificações foram: proporção orgânica [Courtat et al. 2011], coeficiente de malha [Courtat et al. 2011], circuito [Boeing 2021] e densidade de interseção [UNH 2016, Boeing 2018] e proporção de junção de 3 vias. Assim, ao final, das 8914 cidades iniciais fornecidas pelo dataset base [Boeing 2020a, Boeing 2020b, Boeing 2020c], 265 foram escolhidas como planejadas, ou seja, a estrutura do grafo primal dessas cidades foi considerada suficientemente conectada e eficiente.

2.2. *Graph Machine Learning* e aplicações em cidades

A subárea de *Machine Learning* conhecida como *Graph Machine Learning* (GML) é aquela que aplica modelos de Aprendizado de Máquina sobre grafos. Esses modelos podem utilizar de características dos nós, vértices e o próprio grafo para extrair informações relevantes. Essas informações podem ser usadas como entrada para realizar tarefas de classificação de nós, classificação de grafos e predição de novas arestas. Como já dito na seção anterior, redes de ruas podem ser modeladas naturalmente como um grafo. Dessa forma, modelos de GML podem ser aproveitados para realizar tarefas nesse contexto.

Alguns dos principais modelos de GML que vieram a influenciar fortemente a área são aqueles que aplicaram ideias da área de Processamento de Linguagem Natural, como o SkipGram usado no word2vec[Mikolov et al. 2013], para aprender representações de nós a serem utilizados em classificadores por meio de caminhadas aleatórias (*random-walks*). Exemplos desses modelos são o node2vec[Grover and Leskovec 2016] e o DeepWalk[Perozzi et al. 2014] e derivados como struc2vec[Ribeiro et al. 2017] e o NBNE[Pimentel et al. 2019].

O word2vec já foi utilizado por [Liu et al. 2017] para criar o modelo *Road2Vec*. Esse modelo usa do *skipgram* para capturar similaridade entre tráfego de ruas ao conduzir

um processo de predição de tráfego.

A capacidade do node2vec de gerar representações dos elementos de um grafo de rede de ruas já foi avaliada em um contexto grande[Jepsen et al. 2018]. Nesse trabalho mencionado, os autores realizaram experimentos que mostraram quais foram os melhores valores para os hiper-parâmetros a se utilizar para conseguir representações da rede de ruas que levavam aos melhores resultados nas tarefas de predição de velocidade máxima e classificação de tipo de ruas. Além disso, também defendem que o struc2vec pode ter baixa performance em grafos que representem grandes redes de ruas.

[Wang et al. 2020] trazem uma crítica às caminhadas aleatórias no contexto de redes de ruas para desenvolver um novo *framework* de aprendizado de representação chamado de *Representation Learning for Road Network* (RLRN). Eles criticam as caminhadas aleatórias afirmando que elas não representam as movimentações reais dos usuários da rede de ruas já que eles tendem a tentar seguir os menores caminhos até os seus destinos. Dessa forma, esse *framework* realiza caminhamentos seguindo o menor caminho entre dois nós escolhidos aleatoriamente na rede. O *framework* em si é composto por 3 módulos que tentam aprender relacionamentos entre arestas, nós e aresta-nós e é testado na tarefa de classificação de nós e arestas e estimação de tempo de viagem.

Uma outra ferramenta que é aplicada a grafos são as redes neurais convolucionais (CNN). Diferentemente de dados estruturados como tabelas, que possuem linhas ordenadas, e outros dados não estruturados como imagens, que possuem relações fixas entre os *pixels*, os nós de um grafo não possuem uma ordenação específica. Além disso, a quantidade de vizinhos de cada nó não é constante. Essas são as principais questões que dificultam a aplicação de CNNs em grafos, entretanto, é possível adaptá-las.

A aplicação das n camadas de convoluções sobre os vizinhos de um nó forma as chamadas *Graph Convolutional Networks*(GCN)[Kipf and Welling 2016]. Esses modelos se diferenciam pela função de agregação das representações de nós vizinhos para gerar a representação de um nó e também pela função que combina as representações de todos os nós de um grafo para gerar a representação do próprio grafo[Xu et al. 2018]. Exemplos de outros modelos nessa linha são GraphSAGE[Hamilton et al. 2017] e *Graph Attention Networks*(GAT)[Veličković et al. 2017]. Entretanto, devido às limitações naturais das redes de ruas, o grau médio dos nós do grafo primal é baixo. Isso faz com que a agregação de informação de nós vizinhos seja mais sensível a nós *outliers* [Jepsen et al. 2022].

O uso de GCNs para grafos de redes de ruas tem sido explorado. [Jepsen et al. 2022] desenvolveram a chamada *Relational Fusion Networks* (*RFN*), uma rede convolucional que leva em consideração a modelagem primal e dual do grafo para aprender representações para nós e arestas ao mesmo tempo. [Gharaee et al. 2021] desenvolveram a *Graph Attention Isomorphism Network* (GAIN), uma rede fortemente inspirada no GAT que também leva em consideração atributos em arestas do grafo, algo importante para redes de ruas. Junto a isso, usaram de amostragem de nós baseada na vizinhança topológica local e global para realizar tarefas de classificação de tipos de ruas. [Iddianozie and McArdle 2020] também desenvolvem um modelo baseado em GATs chamado *Structure-Aware Sampling-Graph Attention Networks* (*SAS-GAT*). Esse modelo se difere do GAIN pela sua amostragem que considera os nós mais importantes da rede baseado nas métricas de *betweenness-centrality* global e *closenness-centrality* intra-classe dos nós.

3. Metodologia

A seguir é apresentada a metodologia seguida durante este trabalho.

- **Pesquisa Literária**: Realizar pesquisa e estudo de trabalhos sobre *Graph Machine Learning* que possam vir a ajudar a prever novas ruas na rede urbana.
- **Implementação de modelos**: Com base nos estudos realizados na etapa de Pesquisa Literária, implementar modelos de *Graph Machine Learning* para análise estrutural e predição de arestas nos grafos.
- **Experimentação e análise de resultados**: De acordo com os dados encontrados no POC1 e os modelos implementados, realizar experimentação, medição, comparação e análise crítica dos resultados.
- **Entrega dos resultados**: Finalizar o desenvolvimento e escolha de algum modelo. Disponibilizar online o modelo e os resultados conseguidos.

4. Contribuições

Essa seção apresenta a formalização do problema e os dois experimentos realizados durante o trabalho. O primeiro experimento tinha o objetivo de prever a classe das cidades de acordo com as métricas encontradas para elas durante a POC1. O segundo experimento tinha o objetivo de encontrar os melhores parâmetros para o node2vec para classificar as cidades de acordo com a estrutura de seus grafos.

4.1. Formalização do Problema

Seja Z um conjunto de n grafos de forma: $Z = \{G_i = (V_i, E_i), \forall i \in [1, n]\}$ sendo V_i os vértices e E_i as arestas do grafo G_i.

Sejam S_0 e S_1 dois subconjuntos disjuntos não vazios de grafos de Z de forma que $S_0 \cup S_1 = Z$. Define-se $s(G_i)$ como a função que atribui a um grafo $G_i \in Z$ um dos dois subconjuntos de forma que $s(G_i) \in \{S_0, S_1\} \forall G_i \in Z$. Assim, $S_j = \{G_i | s(G_i) = S_j\} \forall j \in \{0, 1\}$.

Seja H_i um vetor de características arbitrárias que representa o estado atual do grafo $G_i \in Z$. Assim, alterações em G_i podem acarretar em mudanças em seu H_i e qualquer mudança em H_i deve acontecer devido a mudanças em G_i.

Dado um subconjunto de grafos $S_j \in \{S_0, S_1\}$, esse subconjunto possui o seu próprio vetor de características arbitrárias H'_j, na presença de um grafo G_i. O vetor de características do subconjunto é dado pela função e da forma $e(S_j, G_i) = H'_j$.

O objetivo é aproximar os vetores de características H_k de $G_k \forall G_k \in S_i$ com i previamente definido ao H'_j de S_j, com $S_j, S_i \in \{S_0, S_1\}$, de acordo com $e(S_j, G_k) \forall G_k \in S_i$ em que:

$$j = \begin{cases} 1 & \text{se } i = 0 \\ 0 & \text{se } i = 1 \end{cases}$$

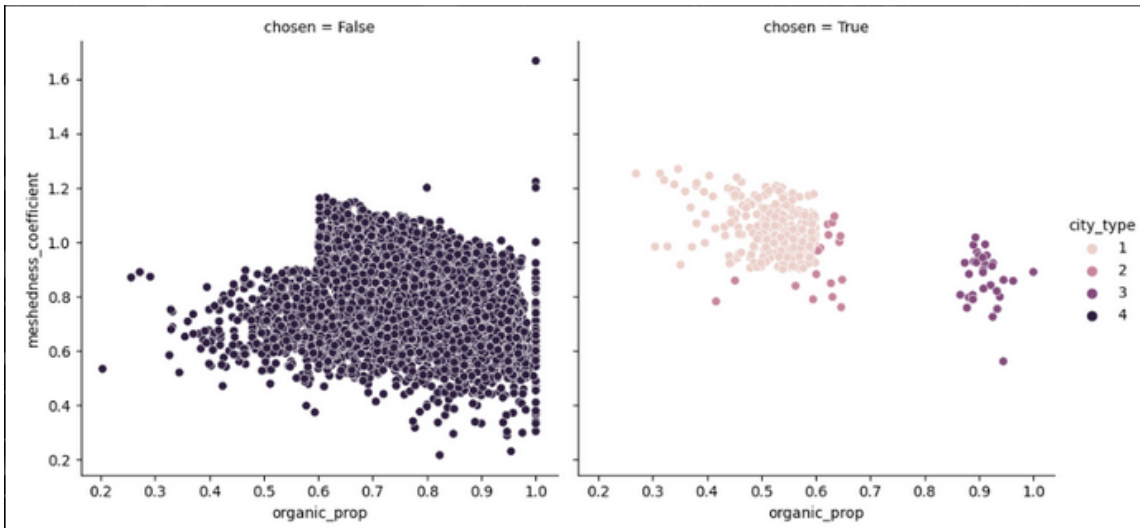
4.2. Classificação via métricas da POC1

Durante a POC1, foram computadas 108 métricas sobre os grafos das cidades do *dataset* disponível [Boeing 2020a, Boeing 2020b, Boeing 2020c]. Como já dito, foram escolhidas

Tipos	Quantidade de Cidades
Planejada 1	265
Planejada 2	17
Planejada 3	31
Não planejada	8596

Tabela 1. Quantidade de cidades de cada tipo definidas na POC1

Figura 1. Distribuição dos tipos de cidades. O Tipo 4 mostra as cidades não planejadas



5 dessas métricas para classificar as cidades como planejadas ou não. Essa classificação foi baseada na pesquisa literária realizada durante a POC1.

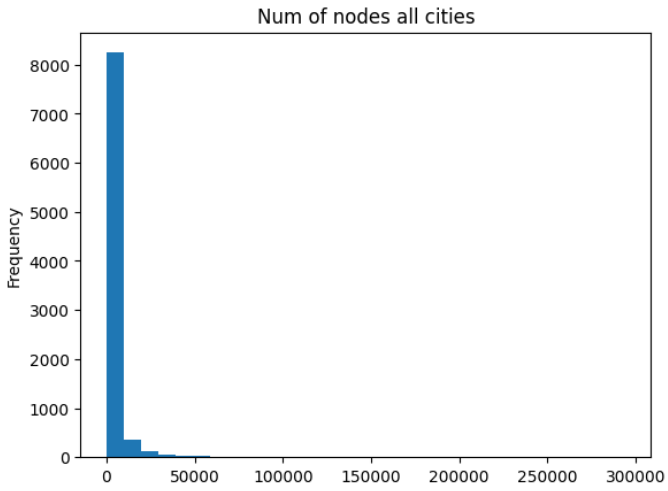
Foi considerado que as cidades planejadas possuíam 3 tipos diferentes definidos pelas seguintes regras:

1. Tipo 1: Proporção orgânica < 0.6 e Meshedness Coefficient $>= 0.9$
2. Tipo 2: Proporção orgânica < 0.65; Meshedness Coefficient $>= 0.75$; Densidade de intersecção entre [80, 120] e não fazer parte do Tipo 1
3. Tipo 3: Proporção de junção de 3 vias maior que 85% e Circuito < 1.05 e não fazer parte do Tipo 1 e do Tipo 2

Todas as outras cidades foram consideradas como não planejadas. A quantidade de cidades que caíram sobre cada tipo além das não planejadas pode ser visto na Tabela 1. A Figura 1 mostra a distribuição dos tipos de cidades quando é levado em consideração os atributos de Proporção Orgânica e Coeficiente de Malha (*Meshedness Coefficient*).

Dessa forma, por mais que as regras de classificação fossem bem definidas, era importante garantir que as classes das cidades eram realmente diferentes sob a visão das métricas computadas sob os seus grafos. Portanto, o objetivo do Experimento 1 era o de treinar um modelo de aprendizado de máquina para classificar as cidades como planejadas ou não baseado nas suas métricas. Assim, é importante ressaltar que, nesse momento, não está sendo avaliada a estrutura do grafo por meio de representações geradas por modelos de *GML* mas apenas sobre métricas relevantes computadas sobre esses grafos.

Figura 2. Distribuição da quantidade de nós dos grafos considerados



O conjunto de métricas computadas sobre os grafos forma um dado estruturado em formato de tabela. No caso, essa tabela possui 108 colunas (métricas) e 8909 linhas (cidades). Um dos melhores modelos de aprendizado de máquina voltados para dados tabulares é o LightGBM[Ke et al. 2017]. O LightGBM é um modelo baseado em Árvores de Decisão e *Boosting* altamente eficiente. Ele possui um mecanismo interno de seleção de *features* conhecido como *Exclusive Feature Bundling* (*EFB*) que, dada a alta dimensionalidade dos dados, será importante para melhorar a eficiência do modelo.

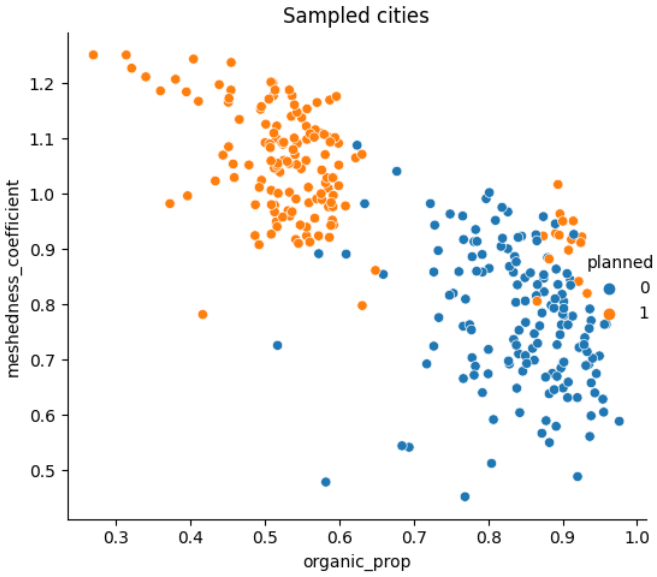
Visando o Experimento 2, foi realizada uma amostragem sobre os dois grupos de cidades. A Figura 2 mostra a distribuição da quantidade de nós para os grafos no *dataset*. É possível ver que a grande maioria dos grafos possuíam menos do que 50 mil nós, mas existiam alguns que chegavam a 300 mil nós. Dado que o grau médio de todos os grafos disponíveis era de 3,23, o tempo de execução do Experimento 2 seria fortemente afetado pela quantidade de arestas resultante nos grafos. Dessa forma, foi decidido escolher um subconjunto do *dataset* original de 150 cidades de cada classe. Essa amostragem levou em consideração apenas cidades com mais de 300 nós e menos de 12 mil e 500 nós. A Figura 3 mostra a distribuição das cidades amostradas para cada grupo.

A amostragem, para ambos experimentos, tem o efeito de aumentar o balanceamento das classes de treino para que o classificador não seja influenciado negativamente.

Como mais um pré-processamento dos dados, foram desconsideradas as métricas que possuíam menos de 100 dados não nulos dentre as 300 instâncias. As métricas desconsideradas foram aquelas que mediam a proporção de cruzamento entre um número muito grande de ruas na rede, como, por exemplo, cruzamentos superiores a 8 ruas. As métricas restantes com dados nulos foram completadas com o valor 0. Também foram retiradas colunas sem utilidade para a classificação, como id da cidade, ou que poderiam gerar algum viés, como região do mundo e país da cidade. Ao final, sobraram 54 colunas.

A versão utilizada do LightGBM foi a disponível no site oficial[lgb]. O treinamento utilizou de Validação Cruzada em 5 *folds* com 20% dos dados separados para teste. Além disso, foi realizado um *Grid Search* para *tunning* de hiperparâmetros. Os valores do *Grid Search* e os escolhidos pelo melhor modelo podem ser vistos na Tabela 2.

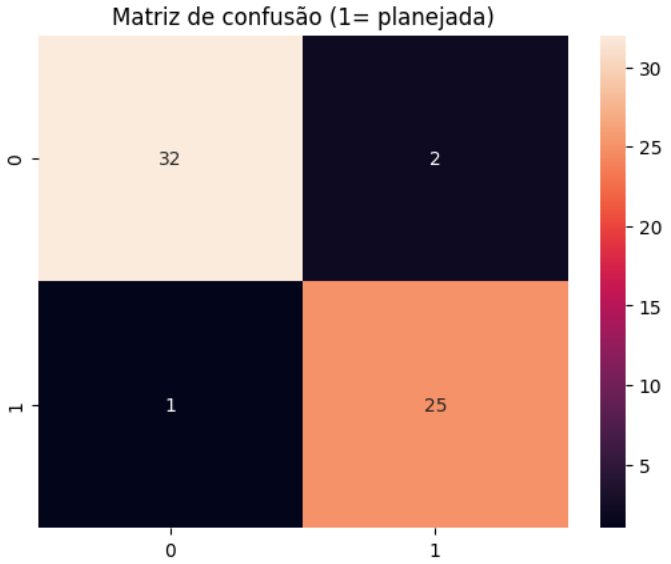
Figura 3. Distribuição das cidades amostradas



Parâmetro	Valores testados	Escolhido
Learning Rate	(0.1, 0.05, 0.03, 0.01, 0.001)	0.1
Boosting Type	('gbdt', 'dart', 'goss')	'gbdt'
Num Leaves	(100, 200, 300)	100
Max Depth	(50, 100, 200, 300, -1)	50
Subsample For Bin	(10, 20, 30, 50)	50
N Estimators	(50, 100, 200, 300)	200

Tabela 2. Parâmetros do Grid Search realizado no Experimento 1. Também foram definidos o *objective* como *binary* e a métrica como *auc*

Figura 4. Matriz de confusão para os dados de teste do Experimento 1



Ao final do processo de validação, a melhor configuração atingiu 0.9625 de *score*. Já no teste, o modelo conseguiu 95% de acurácia e 0.9513 de ROCAUC. A Figura 4 mostra a matriz de confusão para os dados de teste. Como forma de entender melhor quais foram as métricas utilizadas para realizar a classificação das cidades, foram visualizados o valores de shapley via SHAP para modelos baseados em árvore[Lundberg et al. 2020]. A Figura 5 mostra os valores e a influência de cada métrica no resultado. As 3 métricas mais importantes foram utilizadas para classificar manualmente as cidades. Interessante perceber que prop_3way parece ser mais importante. *Circuity* foi outra variável utilizada para classificar mas, como *straightness* está muito ligada a primeira, essa variável, provavelmente, foi utilizada como proxy para *circuity*.

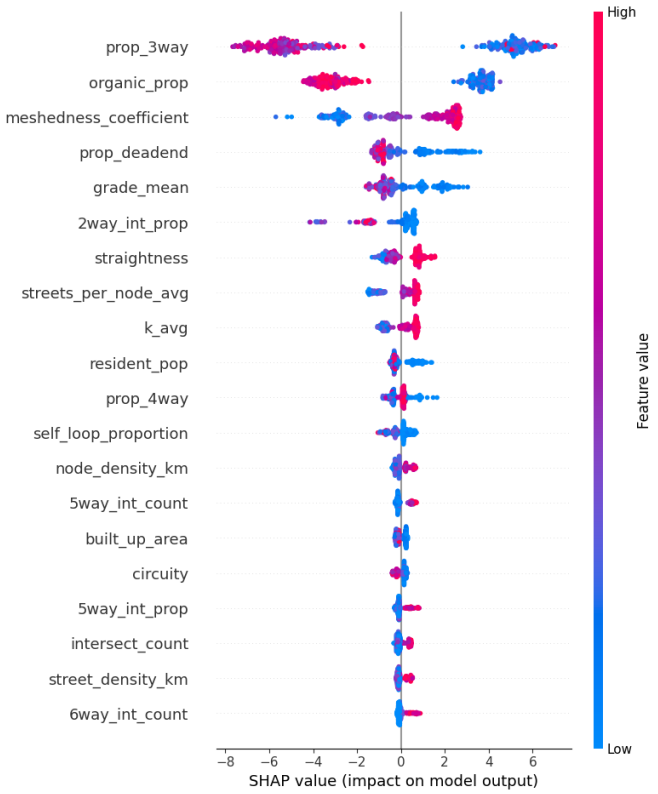
Com o Experimento 1, dada a performance do LightGBM na tarefa, conclui-se que a divisão entre cidades planejadas e não planejadas é realmente separável dadas as métricas propostas e a abordagem tomada para a definição dos grupos.

4.3. Classificação usando node2vec

Uma vez confirmada a divisão das cidades baseadas nas métricas computadas sobre a rede, passa-se a tentar identificar a divisão estrutural dos grafos desses grupos. Como dito anteriormente, [Jepsen et al. 2018] avaliaram as configurações dos parâmetros do *node2vec* para aprender representações dos nós da rede de ruas dinamarquesa. Essa representações foram utilizadas em tarefas de predição de velocidade máxima nas vias e classificação de tipos de ruas. De acordo com os bons resultados obtidos pelos experimentos deles, foi decidido tentar utilizar do *node2vec* para gerar representações dos grafos das redes de ruas das cidades amostradas como descrito na seção anterior. Além disso, o objetivo era treinar e avaliar um modelo classificador que usasse as representações geradas pelo *node2vec* para aprender quais eram as cidades planejadas ou não.

A Figura 6 mostra a organização do experimento. Primeiramente, executa-se o node2vec nos grafos dos dois grupos gerando representações para os seus nós. Depois, agrega-se as representações n-dimensionais de cada nó de cada grafo por meio da média

Figura 5. SHAP para o modelo do Experimento 1 classificar uma cidade como planejada

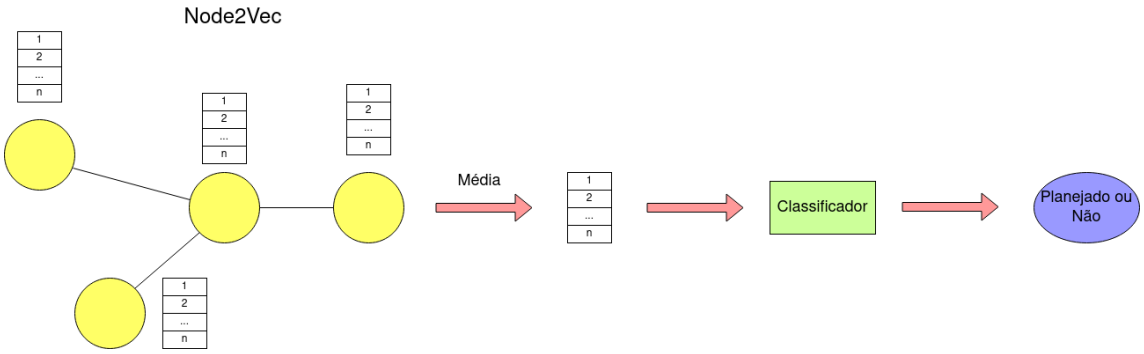


gerando um vetor n-dimensional diferente para cada grafo. Depois, com esses vetores e um label, no caso, cidade planejada ou não, treina-se um classificador para prever, dado o vetor de entrada, se uma cidade é planejada ou não.

Em conformidade com os parâmetros utilizados no artigo mencionado para o node2vec, foram testadas 6 configurações para gerar as representações n-dimensionais dos grafos. A Tabela 3 mostra as combinações de hiperparâmetros testados no Node2Vec. Essa tabela mostra apenas os hiperparâmetros que variaram. O seguintes parâmetros permaneceram fixos: P(2), Q(0.25), Tamanho do Contexto(15), Caminhadas por nó (10) e Número de amostras negativas (2). O tamanho de um caminhamento ser dinâmico refere ao seu ajuste de acordo com a quantidade de nós no grafo sob análise. Dado que o tamanho do caminhamento fixo era de 80 sobre o grafo inteiro de redes de ruas da Dinamarca no artigo base, é justo pensar que esse parâmetro deveria ser alterado ao levar em consideração apenas a rede de ruas contida em cidades. A Tabela 4 mostra as condições e os tamanhos dos caminhamentos realizados no caso dinâmico. A implementação do Node2Vec foi feita utilizando da biblioteca Python PyGeometric[Fey and Lenssen 2019].

Como classificador, também foi escolhido o LightGBM. Para cada configuração testada do Node2Vec, foi realizado um GridSearch para *tunning* de hiperparâmetros do LightGBM. A variação dos hiperparâmetros junto com o valor escolhido pelo melhor modelo para cada configuração do Node2Vec pode ser vista na Tabela 5. Além disso, também foi utilizado de Validação Cruzada com 5 *folds* e 20% dos dados foram separados para teste. Os resultados podem ser vistos na Tabela 6.

Figura 6. Diagrama representando o Experimento 2



ID	Dimensões	Caminhamento	Épocas
(1)	64	Fixo	30
(2)	64	Fixo	50
(3)	64	Dinâmico	30
(4)	64	Dinâmico	50
(5)	128	Fixo	30
(6)	128	Dinâmico	30

Tabela 3. Configurações testadas do Node2Vec. Caminhamento Fixo significa que ele tem o tamanho de 80.

No geral, a combinação do node2vec com o LightGBM não teve uma performance satisfatória. A maioria dos modelos de classificação obtiveram performance próxima do aleatório de acordo com o ROCAUC. Ambas as métricas mostram que o vetor de características dos grafos não eram facilmente separáveis independentemente da configuração utilizada pelo node2vec ainda mais levando em consideração a capacidade do LightGBM.

Existem alguns motivos possíveis para a baixa performance dos modelos. O primeiro é o de que a divisão das classes entre cidades planejadas e não planejadas realizada na POC1 foi insatisfatória. Seja pelos valores das métricas definidas para cada regra de classificação de tipos ou pelo uso de métricas que não conseguiram refletir a estrutura do grafo, a aparente confiabilidade dessa divisão mostrada pelo Experimento 1 é questionada.

Um segundo motivo poderia ser o de que o número de épocas utilizado pelas configurações do node2vec foi muito baixo para convergir as representações dos nós e consequentemente do grafo. Um terceiro motivo poderia advir de erros na implementação do experimento proposto. Um quarto motivo poderia ser o de que o node2vec e os hiperparâmetros utilizados simplesmente não geraram bons modelos para conseguir capturar as representações latentes dos nós dos grafos.

5. Conclusão

O trabalho tinha como objetivo final conseguir sugerir novas ruas baseada na estrutura inerente de cidades consideradas planejadas. Esse objetivo não foi alcançado.

Primeiramente, tentou-se validar a divisão entre cidades planejadas e não planejadas realizada na POC1. Para tal, foi usado o LightGBM para aprender, a partir das métricas computadas sobre os grafos das cidades, como dividir os dois grupos de cidades

Número de nós	Tamanho do caminhamento
< 500	30
< 1000	40
< 2000	50
< 5000	60
< 8000	70
≥ 8000	80

Tabela 4. Tamanho do caminhamento dinâmico do Node2Vec.

Hiperparâmetro	Valores testados	Escolhidos por config do Node2Vec					
		1	2	3	4	5	6
Learning Rate	(0.1, 0.05, 0.03, 0.01, 0.001)	0.1			0.05	0.001	0.01
Boosting Type	('gbdt', 'dart', 'goss')	goss			dart		
Num Leaves	(100, 150, 200)	100					
Max Depth	(50, 100, 200, 300, inf)	50					
N Estimators	(50, 100, 200, 300)	100	50	100	50	300	200

Tabela 5. Hiperparâmetros do Grid Search para o LightGBM no Experimento 2 além dos valores escolhidos pelos melhores modelos. As configurações do Node2Vec são aquelas presentes na Tabela 3.

Melhor modelo para a Config do N2V	ROCAUC	F1-score
(1)	0.534	0.313
(2)	0.472	0.367
(3)	0.533	0.516
(4)	0.547	0.5
(5)	**0.575**	**0.537**
(6)	0.416	0.508

Tabela 6. Resultados para os melhores modelos do LightGBM para cada configuração do Node2Vec

sob uma amostra de 300 instâncias. Nesse experimento, o LightGBM obteve uma performance muito alta, o que indicou, nesse momento, que a divisão entre os dois grupos de cidades estava correto.

O segundo experimento, baseado na pesquisa literária, tentou usar a combinação de Node2Vec com o LightGBM para gerar representações n-dimensionais dos grafos e classificá-los como planejados ou não. Os resultados mostraram que a abordagem seguida produziu modelos que possuíram dificuldade em diferenciar as representações geradas pelo node2vec entre as duas classes com performance ligeiramente melhor do que o aleatório.

Dado que não foi possível confirmar a diferença estrutural dos grafos dos dois grupos, não faria sentido continuar com a tarefa de aprender as estruturas inerentes de um grupo para aplicá-las no outro. Para tal, seria necessário corrigir as representações dos grafos utilizando de outros modelos mais complexos do que o node2vec e que, possivelmente, levassem em consideração características associadas aos elementos do grafo.

Todos os dados e código desenvolvido no trabalho podem ser vistos no repositório do projeto em `https://github.com/Pendulun/POC2`.

Referências

Lightgbm. `https://lightgbm.readthedocs.io/en/stable/`.

(2016). The fundamentals of urbanization evidence base for policy making. White paper, United Nations Human Settlements Programme (UN-Habitat), P. O. Box 30030, 00100 Nairobi GPO KENYA.

Agryzkov, T., Oliver, J. L., Tortosa, L., and Vicent, J. F. (2017). Different typer of graphs to model a city. In *Computational Methods and Experimental Measurements XVIII*. WIT Press.

Angel, S. (2012). *Planet of Cities*. Lincoln Institute of Land Policy, Cambridge, MA.

Angel, S., Lamson-Hall, P., and Blanco, Z. G. (2021). Anatomy of density: measurable factors that constitute urban density. *Buildings and Cities*, 2(1):264–282.

Barrington-Leigh, C. and Millard-Ball, A. (2019). A global assessment of street-network sprawl. *PLOS ONE*, 14(11):e0223078.

Barrington-Leigh, C. and Millard-Ball, A. (2020). Global trends toward urban street-network sprawl. *PROCEEDINGS OF THE NATIONAL ACADEMY OF SCIENCES*, 117(4).

Barthelemy, M., Bordin, P., Berestycki, H., and Gribaudi, M. (2013). Self-organization versus top-down planning in the evolution of a city. *Scientific Reports*, 3(1).

Boeing, G. (2018). A multi-scale analysis of 27, 000 urban street networks: Every US city, town, urbanized area, and zillow neighborhood. *Environment and Planning B: Urban Analytics and City Science*, 47(4):590–608.

Boeing, G. (2020a). Global Urban Street Networks GraphML.

Boeing, G. (2020b). Global Urban Street Networks Indicators.

Boeing, G. (2020c). Global Urban Street Networks Metadata.

Boeing, G. (2021). Street network models and indicators for every urban area in the world. *Geographical Analysis*, 54(3):519–535.

Burghardt, K., Uhl, J. H., Lerman, K., and Leyk, S. (2022). Road network evolution in the urban and rural united states since 1900. *Computers, Environment and Urban Systems*, 95.

Caoa, H., Liu, H., Zhao, F., Li, Y., and Du, M. (2016). The evaluation of node importance in urban road network based on complex network theory. In *MATEC Web of Conferences*, volume 61.

Courtat, T., Gloaguen, C., and Douady, S. (2011). Mathematics and morphogenesis of cities: A geometrical approach. *Phys. Rev. E*, 83:036106.

Fey, M. and Lenssen, J. E. (2019). Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*.

Floater, G., Rode, P., Robert, A., Kennedy, C., Hoornweg, D., Slavcheva, R., and Godfrey, N. (2014). Cities and the new climate economy: the transformative role of global urban growth. *Scientific Reports*.

Gharaee, Z., Kowshik, S., Stromann, O., and Felsberg, M. (2021). Graph representation learning for road type classification. *Pattern Recognition*, 120:108174.

Grover, A. and Leskovec, J. (2016). Node2vec: Scalable feature learning for networks.

Hamilton, W. L., Ying, R., and Leskovec, J. (2017). Inductive representation learning on large graphs.

Hu, M.-B., Jiang, R., Wu, Y.-H., Wang, W.-X., and Wu, Q.-S. (2008). Urban traffic from the perspective of dual graph. *The European Physical Journal B*, 63.

Iddianozie, C. and McArdle, G. (2020). Improved graph neural networks for spatial networks using structure-aware sampling. *ISPRS International Journal of Geo-Information*, 9(11):674.

Jenelius, E. and Mattsson, L.-G. (2015). Road network vulnerability analysis: Conceptualization, implementation and application. *Computers, Environment and Urban Systems*, 49.

Jepsen, T. S., Jensen, C. S., and Nielsen, T. D. (2022). Relational fusion networks: Graph convolutional networks for road networks. *IEEE Transactions on Intelligent Transportation Systems*, 23(1):418–429.

Jepsen, T. S., Jensen, C. S., Nielsen, T. D., and Torp, K. (2018). On network embedding for machine learning on road networks: A case study on the danish road network. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 3422–3431.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Kipf, T. N. and Welling, M. (2016). Semi-supervised classification with graph convolutional networks.

Li, X. and Liu, X. (2020). Research on identification method of key road sections in the road network under disaster situation. In *2020 5th International Conference on Electromechanical Control Technology and Transportation (ICECTT)*, pages 575–578.

Liu, K., Gao, S., Qiu, P., Liu, X., Yan, B., and Lu, F. (2017). Road2vec: Measuring traffic interactions in urban road system from massive travel routes. *ISPRS International Journal of Geo-Information*, 6(11):321.

Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., and Lee, S.-I. (2020). From local explanations to global understanding with explainable ai for trees. *Nature Machine Intelligence*, 2(1):2522–5839.

Masucci, A. P. and Molinero, C. (2016). Robustness and closeness centrality for self-organized and planned cities. *The European Physical Journal B*, 89.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space.

Perozzi, B., Al-Rfou, R., and Skiena, S. (2014). DeepWalk: Online learning of social representations.

Pimentel, T., Castro, R., Veloso, A., and Ziviani, N. (2019). Efficient estimation of node representations in large graphs using linear contexts. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8.

Porta, S., Crucitti, P., and Latora, V. (2004). The network analysis of urban streets: A dual approach.

Porta, S., Crucitti, P., and Latora, V. (2005). The network analysis of urban streets: A primal approach.

Ribeiro, L. F. R., Savarese, P. H. P., and Figueiredo, D. R. (2017). Struc2vec: Learning node representations from structural identity.

Rui, Y., Ban, Y., Wang, J., and Haas, J. (2013). Exploring the patterns and evolution of self-organized urban street networks through modeling. *The European Physical Journal B*, 86.

Scellato, S., Cardillo, A., Latora, V., and Porta, S. (2006). The backbone of a city. *The European Physical Journal B*, 50.

Strano, E., Nicosia, V., Latora, V., Porta, S., and Barthélemy, M. (2012). Elementary processes governing the evolution of road networks. *Scientific Reports*, 2(1).

Uhl, J. H., Leyk, S., Chiang, Y.-Y., and Knoblock, C. A. (2022). Towards the automated large-scale reconstruction of past road networks from historical maps. *Computers, Environment and Urban Systems*, 94.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. (2017). Graph attention networks.

Wang, M.-X., Lee, W.-C., Fu, T.-Y., and Yu, G. (2020). On representation learning for road networks. *ACM Transactions on Intelligent Systems and Technology*, 12(1):1–27.

Xie, F. and Levinson, D. (2007). Measuring the structure of road networks. *Geographical Analysis*, 39(3):336–356.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. (2018). How powerful are graph neural networks?

Xue, J., Jiang, N., Liang, S., Pang, Q., Yabe, T., Ukkusuri, S. V., and Ma, J. (2022). Quantifying the spatial homogeneity of urban road networks via graph neural networks. *Nature Machine Intelligence*, 4(3):246–257.

Zhang, R., Rong, Y., Wu, Z., and Zhuo, Y. (2020). Trajectory similarity assessment on road networks via embedding learning. In *2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM)*. IEEE.