



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

FLÁVIO ROBERTO PIRES QUIRINO FARIAS

**SMART GEO-COMPARATIVE MAPS:
FERRAMENTA PARA VISUALIZAÇÃO E COMPARAÇÃO DE
SIMILARIDADES ENTRE PONTOS DE INTERESSE E REGIÕES
A PARTIR DE VETORES DE CARACTERÍSTICAS**

CAMPINA GRANDE - PB

2023

FLÁVIO ROBERTO PIRES QUIRINO FARIAS

**SMART GEO-COMPARATIVE MAPS:
FERRAMENTA PARA VISUALIZAÇÃO E COMPARAÇÃO DE
SIMILARIDADES ENTRE PONTOS DE INTERESSE E REGIÕES
A PARTIR DE VETORES DE CARACTERÍSTICAS**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em
Ciência da Computação.**

Orientador: Professor Dr. Cláudio Elízio Calazans Campelo.

CAMPINA GRANDE - PB

2021

FLÁVIO ROBERTO PIRES QUIRINO FARIAS

**SMART GEO-COMPARATIVE MAPS:
FERRAMENTA PARA VISUALIZAÇÃO E COMPARAÇÃO DE
SIMILARIDADES ENTRE PONTOS DE INTERESSE E REGIÕES
A PARTIR DE VETORES DE CARACTERÍSTICAS**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em
Ciência da Computação.**

BANCA EXAMINADORA:

**Professor Dr. Claudio Elízio Calazans Campelo
Orientador – UASC/CEEI/UFCG**

**Professor Dr. Herman Martins Gomes
Examinador – UASC/CEEI/UFCG**

**Professor Tiago Lima Massoni
Professor da Disciplina TCC – UASC/CEEI/UFCG**

Trabalho aprovado em: 14 de Fevereiro de 2023.

CAMPINA GRANDE - PB

ABSTRACT

Digital mapping is the process of compiling a set of data in digital format. The main objective of this mapping is to allow the creation of maps with precise representations of the mapped areas. Location, route planning, search, recommendation, among others, are some of the main functionalities that services using digital maps can offer. However, the mentioned functionalities generally do not involve, whenever possible, geographical characteristics in their parameters. Thus, the tool described in this work is a digital map that promotes the integration of geographical characteristics with the basic characteristics of Points of Interest and geographic elements in carrying out activities of search, comparison, and analysis between locations and regions. To do this, points and geographic elements will be represented by feature vectors, and the similarity values of each location will be obtained by pre-trained machine learning models.

Smart Geo-comparative Maps: Ferramenta para visualização e comparação de similaridades entre Pontos de Interesse e Regiões a partir de vetores de características

Trabalho de Conclusão de Curso

Flávio Roberto Pires Quirino Farias (Aluno), Cláudio Campelo (Orientador)

Departamento de Sistemas e Computação
Universidade Federal de Campina Grande
Campina Grande, Paraíba - Brasil

RESUMO

Mapeamento digital é o processo de compilar um conjunto de dados em formato digital. O principal objetivo desse mapeamento é permitir a criação de mapas com representações precisas das áreas mapeadas. Localização, planejamento de rotas, busca, recomendação, entre outros, são algumas das principais funcionalidades que serviços que utilizam mapas digitais podem oferecer. Contudo, as funcionalidades citadas geralmente não envolvem, quando possível, características geográficas em seus parâmetros. Dessa forma, a ferramenta descrita nesse trabalho é um mapa digital que promove integração de características geográficas às características básicas de Pontos de Interesse e de elementos geográficos na realização de atividades de busca, comparação e análise entre locais e regiões. Para isso, os pontos e os elementos geográficos serão representados por vetores de características e os valores de similaridade de cada local serão obtidos por modelos de aprendizagem de máquina pré-treinados.

KEYWORDS

mapas digitais, aprendizagem de máquina, similaridade, pontos de interesse

REPOSITÓRIO

Grupo de repositórios: <https://gitlab.com/geo-web>

1 INTRODUÇÃO

Os avanços tecnológicos na área de computação, principalmente o aumento no poder de processamento computacional, contribuíram para que a cartografia pudesse evoluir aos formatos digitais. Como mapas, por muitas vezes, foram feitos manualmente [5], a carência de detalhes devido à limitação no espaço de reprodução do próprio papel, junto à impossibilidade de realizar atualizações, mesmo que mínimas, destacam e justificam essa evolução.

O campo dos Sistemas de Informações Geográficas (SIG) começou nos anos de 1960, com os computadores e o surgimento do conceito de geografia quantitativa e computacional [7]. Algumas décadas depois, em 1993, o *Xerox PARC Map Viewer* [12] foi um

dos pioneiros no compartilhamento online dos mapas digitais. O *PARC* funcionava através de requisições codificadas em *URL* que retornavam *gifs* em documentos *HTML* (Linguagem de Marcação de HiperTexto). Nos anos seguintes, o rápido desenvolvimento da internet trouxe grandes oportunidades para serem exploradas através do mapeamento digital. Em 2001 foi lançado o *EarthViewer*, software de imagens por satélite da empresa *Keyhole*. Em 2004 a *Google* adquiriu a *Keyhole* [1] e em 2005, o então *EarthViewer* alcançou o grande público sob o nome de *Google Earth* [10]. Ainda no mesmo período, foram lançados o *OpenStreetMap (OSM)* e o *Google Maps*, ferramentas amplamente utilizados atualmente — mais de um bilhão de usuários por mês para o *Maps* [13] e outros milhares de usuários diários para a comunidade do *OSM* [9].

Em pouco mais de 5 décadas pudemos observar uma enorme evolução no campo da cartografia, com destaque para os anos após a expansão da internet. Porém, apesar do grande número de usuários e de funcionalidades, recursos de similaridade geoespacial ainda não fazem parte das principais ferramentas de mapas digitais.

1.1 Problema

Devido ao uso em larga escala dos mapas digitais e suas diversas funcionalidades, vários nichos onde essas ferramentas têm aplicações específicas acabaram sendo criados. Trazendo para análise espacial, no **uso pessoal**, podemos citar a busca por lugares e visualização de regiões; para **uso comercial**, há o *Business Intelligence* baseado em mapas, onde é possível encontrar tendências demográficas em cidades e regiões para análise de mercado; no **planejamento urbano** podemos gerenciar a cobertura de redes elétricas, internet, linhas de gás e outros tipos de fornecimento, delimitar áreas de risco ambiental ou com carência de infraestrutura, analisar a viabilidade de construção de espaços públicos como escolas, hospitais e vias; entre outras diversas aplicações.

Ferramentas como *Google Maps* definem seus parâmetros de busca em endereços, nomes de locais, localizações geográficas ou a escolha de categorias, como estações de gás ou lojas de doces [11, 14]. Os SIG também oferecem diversas ferramentas para manipular dados espaciais como visualização, edição e análise que, no entanto, na maioria das vezes, são funcionalidades voltadas à produção de mapas. Porém, algumas atividades, como operações de busca e comparação, necessitam de outras características, como, por exemplo, a similaridade geoespacial, que é uma forma de identificar a similaridade entre áreas geográficas baseada em características obtidas a partir dos dados geoespaciais (dados geográficos que consideram o espaço e o tempo) da vizinhança daquela área.

Os autores retêm os direitos, ao abrigo de uma licença Creative Commons Atribuição CC BY, sobre todo o conteúdo deste artigo (incluindo todos os elementos que possam conter, tais como figuras, desenhos, tabelas), bem como sobre todos os materiais produzidos pelos autores que estejam relacionados ao trabalho relatado e que estejam referenciados no artigo (tais como códigos fonte e bases de dados). Essa licença permite que outros distribuam, adaptem e evoluam seu trabalho, mesmo comercialmente, desde que os autores sejam creditados pela criação original.

Existem diversos benefícios ao considerar essas características na realização de buscas. Além de expandir o próprio universo da busca, permitindo outros tipos de entradas durante as consultas, é possível também realizar comparações utilizando os dados geoespaciais e, assim, procurar por regiões e/ou locais similares. A partir disso, esse artigo apresenta uma ferramenta de mapa digital que utiliza a similaridade geoespacial unida às características básicas de Pontos de Interesse (POI, do inglês *Point of Interest*) na realização de buscas, análises e comparações entre locais.

1.2 Objetivos

O objetivo geral deste trabalho é apresentar o desenvolvimento de uma ferramenta *web* de mapa digital que utiliza características geográficas nas análises e comparações entre *POIs* e regiões. Os seguintes tópicos descrevem os objetivos específicos necessários para atingir o objetivo geral:

- Buscar *POIs* similares a partir de uma ou mais categorias, como restaurantes e/ou hotéis;
- Buscar locais similares a partir de um conjunto de características de um local, seja um conjunto de *POIs* e/ou um conjunto de Elementos Geográficos (*GEOs*) em uma determinada área;
- Buscar regiões similares a partir da seleção de uma região (considera-se região o conjunto de *POIs* e/ou *GEOs* presentes dentro de determinada área).

1.3 Justificativa

Como apresentado na seção 1.1, o campo de mapas digitais demonstrou uma grande evolução nas últimas décadas. Partindo de uma abordagem manual de construção de mapas, com problemas de precisão e um baixo nível de detalhes, a uma abordagem totalmente digital e automatizada, superando os problemas da geração anterior e adicionando inúmeros novos recursos, evidenciando, assim, que esse campo e as soluções que o envolvem ainda têm um longo caminho pela frente. Contudo, mesmo atualmente, algumas limitações podem ser observadas, como a ausência de dados geoespaciais nos parâmetros de busca e na comparação entre locais e regiões.

A identificação de regiões geograficamente similares pode ser aplicada em diversas atividades. Para demonstrar a importância desse recurso, a seguir são apresentados três exemplos envolvendo as áreas mencionadas acima (seção 1.1):

- **Uso Pessoal:** Pessoas que estão de mudança para outra cidade, estado ou até mesmo outro país, podem buscar por vizinhanças semelhantes à sua vizinhança atual e assim decidir onde morar. Mais especificamente, casais com filhos pequenos podem encontrar regiões com escolas, parques e farmácias, mesmo sem pesquisar diretamente por esses ambientes. Uma vez que eles morem em uma região que possua esses *POIs*, a busca por similaridade de regiões deve retornar outras regiões com características similares.
- **Comércio:** Empresas de todos os tamanhos podem se beneficiar com o uso de dados geoespaciais. De pequenos estabelecimentos à grandes corporações, dados geoespaciais podem ajudar o negócio a tomar melhores decisões sobre onde abrir suas lojas. Observando empresários, por exemplo. Ao analisar dados espaciais, os empresários podem visualizar

os melhores lugares para abrir lojas de acordo com informações demográficas, ajudando a identificar regiões com alta concentração de consumidores. Podem também observar a renda média da região e o custo de vida para determinar se aquela é uma boa área para o tipo de loja que estão abrindo. Além disso, podem identificar áreas com alta concorrência e determinar onde é menos concorrido e tem mais potencial de sucesso.

- **Planejamento Urbano:** As etapas de planejamento, segundo o Programa Cidades Sustentáveis[15], envolvem, entre outros, a definição e implementação de estratégias que promovam o uso eficiente da terra e seus recursos em benefício da cidade e de seus habitantes. Assim, dados geoespaciais e a similaridade geoespacial podem ajudar o planejamento urbano de diversas formas. Os dados geoespaciais podem fornecer imagens detalhadas do ambiente físico e social de uma cidade. Tais dados podem ser utilizados para análise das redes de transporte, medição do nível de segurança pública e no monitoramento das condições ambientais. Além de possibilitar acompanhar de forma temporal o progresso das atividades de planejamento urbano. Enquanto a similaridade geoespacial pode ser utilizada para comparar cidades e identificar áreas com potencial similaridade. Por exemplo, a similaridade geoespacial pode identificar cidades com densidade populacional similar e com estruturas urbanas similares. Essa combinação de dados e similaridade permite que o planejamento seja feito conforme as necessidades específicas de cada cidade.

Como apresentado, a similaridade geoespacial pode ser encontrada em diversos campos de atividade. Em vista disso, considerar as características geoespaciais, quando possível, é extremamente importante uma vez que permite identificar diferentes necessidades e traz funcionalidades que se ajustam de acordo com os dados de cada região.

2 SOLUÇÃO

O *Smart Geo-comparative Maps*, ou *SG Maps*, é um sistema *web* de busca e recomendação em mapas digitais, onde é promovida a integração de dados geoespaciais, por meio de vetores de características, representando as categorias de **POIs**, como restaurantes e bares, e **GEOs**, como rios, lagos e ruas. Esses vetores são obtidos por meio de modelos de aprendizagem de máquina (ML) pré-treinados desenvolvidos em uma pesquisa em andamento no Laboratório de Computação Inteligente Aplicada (LACINA) da Universidade Federal de Campina Grande (UFCG). Seguindo uma arquitetura baseada em microsserviços, o *SG Maps* foi implementado visando aproveitar ao máximo os benefícios desse estilo arquitetural.

2.1 Fundamentação Teórica

2.1.1 Modelos de Similaridade. Os dois principais elementos utilizados para comparação de similaridade são os *POIs* e *GEOs*. A definição dos vetores de características e suas comparações a fim de encontrar valores de similaridade geoespacial são obtidas a partir de modelos de ML pré-treinados (LACINA). Os modelos são baseados na técnica *word2vec* [8], que consiste, de forma geral, em mapear os *POIs* e *GEOs*, nesse caso, a partir de seus nomes e categorias, para vetores que representam esses elementos num espaço vetorial.

2.1.2 *Microserviços*. Segundo Martin Fowler [6], o estilo arquitetural de microserviços pode ser definido como uma abordagem para desenvolver uma única aplicação como um conjunto de pequenos serviços, onde cada serviço executa seu próprio processo. Esse estilo traz consigo benefícios e pontos negativos que precisam ser considerados durante o planejamento. Podemos descrever as principais características dos microserviços [4] como:

- **Flexibilidade:** O sistema pode acompanhar as constantes mudanças do ambiente de negócios e consegue suportar todas as modificações necessárias;
- **Modularidade:** O sistema é composto por componentes isolados onde cada componente contribui para o comportamento geral do sistema, em vez de ter um único componente que oferece todas as funcionalidades;
- **Evolução:** O sistema deve permanecer disponível enquanto constantemente evolui e adiciona novos recursos.

2.2 Funcionalidades

As principais funcionalidades fornecidas pelo sistema estão divididas entre a aplicação *frontend*, ou seja, a interface gráfica que fornece ao usuário a interação com o mapa, e a *API*, onde são encontradas as funções mais específicas dos modelos de similaridade.

2.2.1 Levantamento de Requisitos.

A Tabela 1 mostra o resultado do levantamento de requisitos feito a partir de um documento de necessidades e funcionalidades:

Tabela 1: Levantamento de Requisitos

ID	Necessidade	Prioridade
1	Busca Por Categoria na Base de Dados	Alta
2	Busca por Complemento da Categoria na API Word2Vec	Alta
3	Selecionar Região	Alta
4	Selecionar Conjunto de POIs	Alta
5	Inferir POI a partir do Contexto	Alta
6	Inferir regiões semelhantes a partir de uma região pré-selecionada	Alta
7	API para comportar os modelos Word2Vec	Alta

2.2.2 Funcionalidades da Interface Gráfica.

O *frontend* do sistema apresenta um conjunto de funcionalidades que permitem a busca e comparação entre *POIs* e Regiões.

- **Busca:** O sistema permite que o usuário procure por *POIs* a partir de seus conjuntos de categorias;
- **Visualização em grupos:** Devido ao alto número de pontos em uma cidade, o usuário visualizará, por padrão, apenas os principais pontos de cada área, onde a popularidade é definida através da contagem de *check-in* e a área é calculada automaticamente, sob demanda, de acordo com as coordenadas que definem o posicionamento atual da tela do usuário e o nível de aproximação do mapa;
- **Similaridade entre Regiões:** É possível também calcular a similaridade entre regiões. O usuário deve criar uma região

base no mapa, que será considerada referência entre as demais regiões para encontrar a mais similar. Não é possível existir mais de uma região base simultaneamente. Além disso, deve também adicionar um conjunto de regiões candidatas. Estas podem ser adicionadas um número ilimitado de vezes e todas as regiões serão consideradas candidatas a mais similar em relação à região base. Ao realizar a comparação, serão exibidos os valores de similaridade de cada região candidata e a região mais similar será destacada de verde no mapa;

- **Similaridade entre POIs:** Ao ativar o botão de comparar *POIs*, o usuário deve selecionar, de acordo com as informações mostradas, dois *POIs*. O sistema mostrará então o valor de similaridade entre os pontos de interesse;

2.2.3 Funcionalidades da API do Modelo.

As funções de similaridade foram disponibilizadas em uma *REST API* que funciona como uma interface para os modelos de *ML* que estão sendo produzidos no LACINA. A *API* implementa apenas uma interface de comunicação que permite a aplicação utilizar as funções dos modelos. Dentre as funções listadas na Figura 1, as principais são:

- **Vetor (de Representação) de Categoria** (do inglês *Type Vector*): Retorna, a partir de uma categoria, seja de um *POI* ou de um *GEO*, o vetor de características que representa aquele objeto.
- **(Categorias) Mais Similares** (do inglês *Most Similars*): Retorna, a partir de uma categoria, seja de um *POI* ou de *GEO*, um conjunto de categorias similares, ordenadas a partir do valor de similaridade, definido pela porcentagem de confiança.
- **Vetor (de Representação) de Conjunto** (do inglês *Avg Vector*): Retorna, a partir de um conjunto de categorias, sejam de um *POI* ou de *GEOs*, o vetor de características que representa aquele conjunto.
- **Similaridade de Cosseno** (do inglês *Cos Similarity*): Retorna, a partir de duas categorias de *POIs*, o valor de similaridade definido a partir do cálculo da distância de cosseno entre os vetores de representação das categorias.
- **Similaridade de Região** (do inglês *Similar Regions*): Retorna, a partir das coordenadas de um retângulo, o número de linhas e colunas para formar uma grade de retângulos, e do identificador do retângulo interno a ser comparado, os valores de similaridade região, ou seja, calcula para cada *POI* e *GEO* presente na área determinada, o valor de similaridade de cada retângulo da grade com o retângulo base.

2.2.4 Funcionalidades da Interface de Banco de Dados.

Por fim, a *API* que acessa o Banco de Dados (BD) é responsável por realizar as consultas necessárias para o funcionamento da interface gráfica e da *API* do modelo. Ela permite que o sistema realize consultas diretamente no BD, obtendo os *POIs* e *GEOs*, permitindo a aplicação de filtros e as demais variedades de uma consulta ao banco.

p2v

GET	/p2v/vocabulary	Vocabulary
GET	/p2v/type_vector	Type Vector
GET	/p2v/most_similars	Most Similars
GET	/p2v/avg_vector	Avg Vector
GET	/p2v/cos_similarity	Cos Similarity
GET	/p2v/region_vector	Region Vector
GET	/p2v/rectangles	Rectangles
GET	/p2v/similar_regions	Similar Regions

Figura 1: Documentação da API dos modelos

2.3 Arquitetura

O sistema foi construído baseado na arquitetura de microsserviços, onde os principais módulos foram divididos em serviços independentes que unidos compõem o sistema. Os principais módulos são: O *frontend*, também chamado aqui de interface gráfica, e o *backend*. Este, por sua vez, foi modularizado em microsserviços: a API dos modelos *Word2Vec*, a API do Banco de Dados e uma API geral chamada de *Gateway*, responsável por realizar a comunicação entre os microsserviços. O Banco de Dados não faz parte da solução desenvolvida, então foi considerado um sistema externo.

Na realização da documentação arquitetural, foi utilizado o modelo de descrição arquitetural C4[2], que consistem em uma forma simples de documentar arquitetura de programas. O C4 define uma separação em 4 níveis: *Contexto*, *Container*, *Componente* e *Código*. Contudo, a fim de praticidade e de uma melhor apresentação da informação, apenas os dois primeiros níveis foram considerados neste documento.

2.3.1 Visão de Contexto.

O diagrama de contexto representa um ponto de partida na produção dos diagramas arquiteturais, mostrando a posição do sistema principal e seu relacionamento com os elementos externos.

O *SG Maps* é uma ferramenta de mapas digitais que utiliza modelos de aprendizagem de máquina para realizar buscas e comparações entre elementos de um mapa. Para obter o mapa, o sistema utiliza o *OpenStreetMap* [3], um sistema aberto que fornece dados de mapa, desenvolvido por uma comunidade voluntária de mapeadores que contribuem e mantêm os dados atualizados sobre estradas, trilhos, cafés, estações, ferroviárias e muito mais por todo o mundo. Os dados de *POIs* e *GEOs* são obtidos a partir de um banco de dados relacional (*PostgreSQL*) já existente.

Mais detalhadamente, na Figura 2, vemos o *SG Maps*, o sistema abordado nesse documento, e suas principais dependências externas, além de um usuário comum (*client*) que interage com o sistema.

2.3.2 Visão de Container.

O diagrama de Container representa os *containers* que compõem o sistema principal. *Containers*, neste documento, são elementos que precisam ser executados para que o sistema funcione de forma geral.

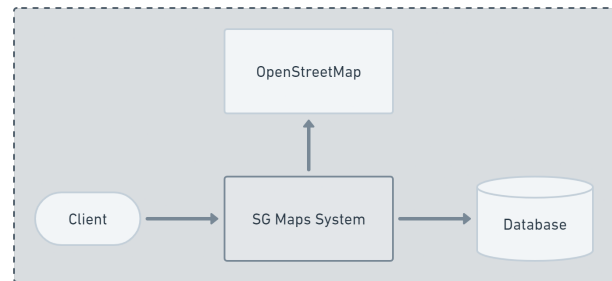


Figura 2: Visão de Contexto do Sistema SG Maps

Quando observamos os *containers* que formam o *SG Maps*, podemos dividi-los em 2 partes principais:

- **Frontend:** Desenvolvido em ambiente *Node.js*, utiliza *leaflet* como a biblioteca de construção da interface gráfica de mapa. Permite que o usuário tenha acesso direto ao mapa e as principais funcionalidades presentes na ferramenta.
- **Backend:** Constituído pelas APIs do BD, dos modelos *Word2Vec* e o *Gateway*.
 - **API Gateway:** Funciona como interface entre os demais microsserviços. Assim, é responsável por realizar a comunicação entre esses serviços para atender as requisições recebidas. Além disso, qualquer *client* do *backend* pode ter acesso aos microsserviços através de um único lugar.
 - **API do Banco de Dados:** Responsável pela comunicação dos serviços com o banco de dados, permitindo a realização de consultas e fornecendo os principais recursos de um banco de dados.
 - **API dos Modelos Word2Vec:** Interface *REST API* para disponibilizar as funcionalidades de similaridade dos modelos de aprendizagem de máquina através da representação vetorial dos dados.

Na Figura 3 é possível observar melhor a distribuição dos *containers* e seus relacionamentos na composição do sistema *SG Maps*.

3 SISTEMA EM USO

O menu lateral exibe as principais funcionalidades da aplicação. A primeira funcionalidade básica do *SG Maps* é a busca e pode ser observada na Figura 4. Ao selecionar algum resultado de busca, o usuário é levado até a coordenada do *POI*. Além disso, as marcas apresentadas no mapa são, a princípio, grupos de *POIs*. Esses grupos são formados por *POIs* dentro e uma área delimitada por um raio definido pela posição da tela no mapa e da distância da aproximação. O *POI* mais popular, ou seja, aquele que tem um maior valor de *check-in*, será escolhido como representante do grupo, sendo, assim, exibido na tela. Mover a tela, realizar a busca, aumentar ou diminuir o *zoom* afeta diretamente os grupos exibidos.

No bloco de controles de similaridade é possível realizar a comparações entre *POIxPOI* e comparações entre RegiãoxRegião. A Figura 5 demonstra a função de cálculo de similaridade entre *POIs*. O resultado da Similaridade entre Regiões pode ser visualizado na Figura 6.

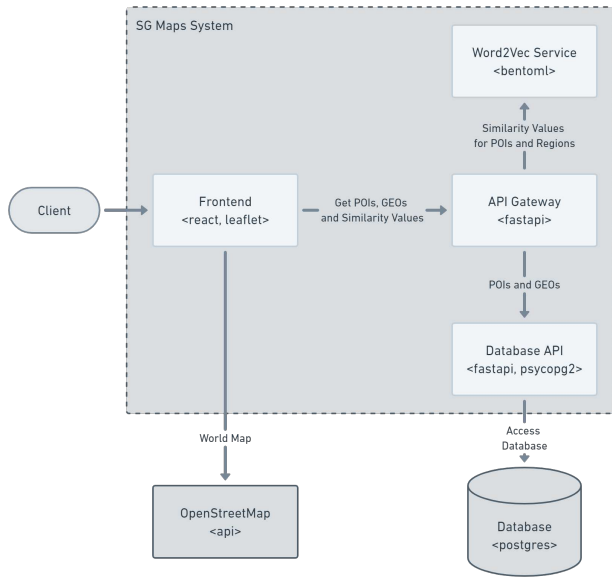


Figura 3: Visão de Container do Sistema SG Maps

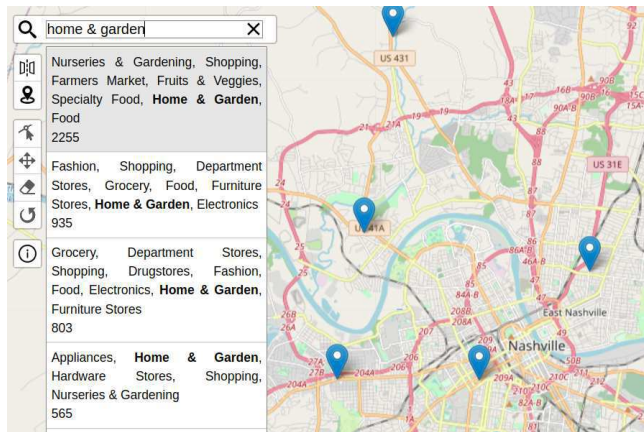


Figura 4: Página Inicial do SG Maps

Além da interface gráfica, é possível utilizar os recursos da API dos modelos diretamente. A Figura 1 mostra os endpoints disponíveis na API.

4 AVALIAÇÕES

A avaliação da ferramenta foi realizada por 9 usuários, avaliando o nível de satisfação com a ferramenta de forma geral e, em seguida, de forma específica com as funcionalidades busca, similaridade entre POIs e similaridade entre Regiões. Para obter o nível de satisfação, os usuários precisaram responder um formulário que contendo uma matriz de satisfação para cada funcionalidade. Cada item representava uma afirmação sobre a utilização de alguma funcionalidade e poderia ser avaliado numa escala de 5 pontos, onde 1 indica Discordo Totalmente e 5 indica Concordo Totalmente. Além disso, os

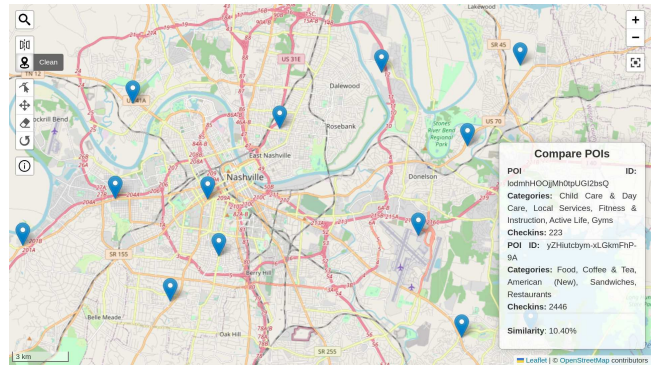


Figura 5: Exemplo de Similaridade entre POIs

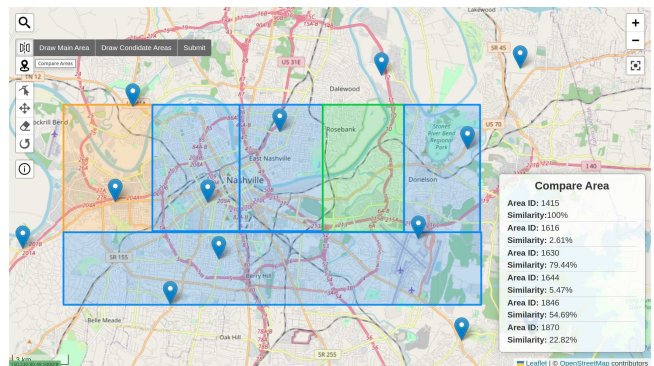


Figura 6: Exemplo de Similaridade entre Regiões

usuários também tinham a opção de escrever *feedbacks* para cada parte testada.

4.1 Avaliação das Funcionalidades

- Avaliação da Ferramenta:
 - 8 de 9 pessoas identificaram e conseguiram utilizar as funcionalidades com facilidade;
 - 6 de 9 pessoas consideram que as informações disponíveis são suficientes para utilizar a ferramenta;
- Avaliação da Busca:
 - 9 de 9 pessoas conseguiram realizar busca com facilidade e concordaram que os resultados retornados foram coerentes;
- Avaliação da Comparação entre Regiões:
 - 8 de 9 pessoas conseguiram realizar a comparação entre regiões com facilidade;
 - 7 de 9 pessoas conseguiram identificar o resultado com facilidade;
 - 8 de 9 pessoas concordaram que o resultado retornado é coerente;
 - 7 de 9 pessoas concordaram que as informações de comparação foram bem dispostas na tela;
- Avaliação da Comparação entre POIs:
 - 7 de 9 pessoas conseguiram realizar a comparação entre POIs com facilidade;

- 9 de 9 pessoas conseguiram identificar os resultados de comparação com facilidade;

4.2 Feedback das Funcionalidades

- Avaliação da Ferramenta: A maior sugestão para a ferramenta é a presença de uma opção de tutorial, como um *tour* no menu na primeira utilização, um vídeo explicativo ou textos curtos que destaquem as funcionalidades de cada botão;
- Avaliação da Busca: Não houve sugestões de melhoria;
- Avaliação da Comparação entre Regiões: Novamente, foi sugerido que houvesse um cartão de informações sobre o que é a similaridade de regiões e como ela é feita. Também sugeriram que os retângulos contivessem seus respectivos identificadores;
- Avaliação da Comparação entre POIs: Assim como a função anterior, um cartão de informações com um passo a passo sobre como realizar a comparação foi a sugestão para essa funcionalidade.

5 EXPERIÊNCIAS E LIÇÕES APRENDIDAS

5.1 Processo de Desenvolvimento

No início do projeto, foi realizada uma avaliação dos requisitos para entender as necessidades e as funcionalidades esperadas. Para isso, foi desenvolvido um documento detalhado que incluía uma descrição completa dos requisitos.

Com base nas informações obtidas no documento de requisitos, foi estabelecido um cronograma de desenvolvimento para garantir o alinhamento das atividades que seriam desenvolvidas. Inicialmente o progresso do projeto foi acompanhado através de reuniões semanais para discutir o andamento das atividades e solucionar dúvidas.

A escolha das tecnologias utilizadas foi feita, a princípio, sem nenhum planejamento, gerando atraso e um considerável aumento do nível de dificuldade no desenvolvimento das atividades planejadas. Graças a arquitetura baseada em microsserviços, foi possível alterar os pontos de gargalo do projeto, selecionando as tecnologias que melhor atendiam às necessidades.

5.2 Principais Desafios

Desenvolver a interface do mapa com Leaflet, React e Hooks foi um desafio significativo para frontend. Embora o React tenha sido inicialmente considerado uma opção viável, acabou por não ser possível continuar com sua utilização devido às limitações encontradas.

Um dos maiores obstáculos enfrentados foi a exibição de uma quantidade excessiva de POIs no mapa, o que poderia causar congestionamento, comprometendo o desempenho da ferramenta. No entanto, o problema foi resolvido ao agrupar os POIs próximos em grupos, exibindo apenas o melhor (POI com maior *check-in*) de cada grupo para evitar o congestionamento no *frontend*.

Além disso, a implementação da *REST API* para funcionar como interface na utilização dos modelos *Word2Vec* também foi um desafio, já que muitas das tecnologias envolvidas eram relativamente novas e desconhecidas. De forma geral, o desenvolvimento da interface do mapa com Leaflet, React e Hooks trouxe desafios significativos, mas foi possível superar e encontrar soluções para melhorar o desempenho, utilização e desenvolvimento da ferramenta.

5.3 Trabalhos Futuros

Esta ferramenta aborda apenas a similaridade entre regiões e POIs e não utiliza todas as funções dos modelos *Word2Vec* apresentados. No entanto, as funções não utilizadas podem ser exploradas em projetos futuros. A forma de agrupar POIs é uma parte importante da ferramenta. Atualmente, os grupos são montados de forma simples, mas há potencial para melhorar a diversidade na experiência do usuário, considerando outros valores, como a similaridade entre as categorias dos POIs. Em resumo, a ferramenta de mapas digitais tem muito potencial para ser aprimorada. Além disso, existe espaço para melhorar como os POIs são agrupados para oferecer uma melhor experiência de visualização para o usuário.

REFERÊNCIAS

- [1] Loren Baker. [n. d.]. Google Acquires Satellite Mapping Firm Keyhole. <https://www.searchenginejournal.com/google-acquires-satellite-mapping-firm-keyhole/993/>
- [2] Simon Brown. [n. d.]. The C4 model for visualising software architecture. <https://c4model.com/>
- [3] contribuidores do OpenStreetMap. [n. d.]. O OpenStreetMap fornece dados de mapa para milhares de sites, aplicativos móveis e dispositivos de hardware. <https://www.openstreetmap.org/about>
- [4] Nicola Dragoni, Saverio Giallorenzo, Alberto Lluch Lafuente, Manuel Mazzara, Fabrizio Montesi, Ruslan Mustafin, and Larisa Safina. 2017. Microservices: yesterday, today, and tomorrow. *Present and ulterior software engineering* (2017), 195–216.
- [5] Institute for Digital Research and Education. [n. d.]. History of Digital Mapping and Beyond. <https://sandbox.idre.ucla.edu/sandbox/history-of-digital-mapping>
- [6] Martin Fowler. [n. d.]. Microservices. <https://martinfowler.com/articles/microservices.html>
- [7] Environmental Systems Research Institute. [n. d.]. History of GIS. <https://www.esri.com/en-us/what-is-gis/history-of-gis>
- [8] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).
- [9] Pascal Neis. [n. d.]. OSMstats about members. <https://osmstats.neis-one.org/?item=members>
- [10] Jeremy Norman. [n. d.]. Origins of Google Earth. <https://www.historyofinformation.com/detail.php?entryid=3145>
- [11] Google Maps Platform. [n. d.]. Place Search. <https://developers.google.com/maps/documentation/places/web-service/search>
- [12] Steve Putz. [n. d.]. Experimental Map Browser WWW server. <http://web.mit.edu/eichin/www/xerox-maps.html>
- [13] Ethan Russell. [n. d.]. 9 things to know about Google's maps data: Beyond the Map. <https://cloud.google.com/blog/products/maps-platform/9-things-know-about-googles-maps-data-beyond-map>
- [14] Google Support. [n. d.]. Search locations on Google Maps. <https://support.google.com/maps/answer/3092445>
- [15] Programa Cidades Sustentáveis. [n. d.]. ETAPAS PARA ELABORAÇÃO DO PLANO INTEGRADO. https://www.cidadessustentaveis.org.br/institucional/planejamento-integrado_etapas