BACHAREL EM CIÊNCIA DA COMPUTAÇÃO

ANÁLISE E VISUALIZAÇÃO DE SÉRIES TEMPORAIS DE DADOS DE MÍDIAS SOCIAIS DURANTE A PANDEMIA DE COVID-19

MATHEUS WAGNER DOS SANTOS MARTINS

Rio Verde, GO

2024

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA GOIANO -
CAMPUS RIO VERDE
BACHAREL EM CIÊNCIA DA COMPUTAÇÃO

ANÁLISE E VISUALIZAÇÃO DE SÉRIES TEMPORAIS DE DADOS DE MÍDIAS SOCIAIS DURANTE A PANDEMIA DE COVID-19

MATHEUS WAGNER DOS SANTOS MARTINS

Trabalho de Conclusão de Curso apresentado ao Instituto Federal de Educação, Ciência e Tecnologia Goiano - Campus Rio Verde, como requisito parcial para a obtenção do Grau de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. Douglas Cedrim Oliveira

Rio Verde, GO

Dezembro, 2024

Sistema desenvolvido pelo ICMC/USP
Dados Internacionais de Catalogação na Publicação (CIP)
Sistema Integrado de Bibliotecas - Instituto Federal Goiano

M386a

Martins, Matheus Wagner dos Santos
Análise e visualização de séries temporais de dados de mídias sociais durante a pandemia de COVID-19 / Matheus Wagner dos Santos Martins ; orientador Douglas Cedrim Oliveira. -- Rio Verde, 2024.
68 f.

TCC (Bacharelado em Ciência da Computação) -- Instituto Federal Goiano, Campus Rio Verde, 2024.

1. Visualização de dados. 2. Mídias sociais. 3. Distorção dinâmica do tempo. 4. COVID-19.. I. Oliveira, Douglas Cedrin, orient. II. Título.

Responsável: Johnathan Pereira Alves Diniz - Bibliotecário-Documentalista CRB-1 nº2376

SERVIÇO PÚBLICO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA GOIANO

TERMO DE AUTORIZAÇÃO PARA DISPONIBILIZAR PRODUÇÃO TÉCNICA NO REPOSITÓRIO INSTITUCIONAL DO IF GOIANO

Repositório Institucional do IF Goiano - RIIF Goiano Sistema Integrado de Bibliotecas

- Profissional de Educação do IF Goiano -

Com base no disposto na Lei Federal nº 9.610/98, e manual sobre a Produção Técnica, publicado pela DAV/CAPES/MEC*, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia Goiano, a disponibilizar gratuitamente o documento no Repositório Institucional do IF Goiano (RIIF Goiano), sem ressarcimento de direitos autorais, conforme permissão assinada eletronicamente abaixo, em formato digital para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IF Goiano.

Identificação da Produção Técnica – DAV/CAPES

[ ] Editoria [ ] Material Didático

[ ] Curso de Formação Profissional [ ] Projetos de Extensão à Comunidade

[ ] Relatório Técnico Conclusivo [ ] Atividade Técnica/Tecnológica

[ ] Disseminação do Conhecimento Técnico/Tecnológico [ ] Produto Bibliográfico

[X] Outras Produções Técnicas - Tipo: TCC (Graduação)

Nome Completo do Autor/a: Matheus Wagner dos Santos Martins

Matrícula: 2018202201940046

Título do Trabalho: Análise e visualização de séries temporais de dados de mídias sociais durante a pandemia de COVID-19

Restrições de Acesso ao Documento

Documento confidencial: [X] Não [ ] Sim

Justifique: _____

Informe a data que poderá ser disponibilizado no RIIF Goiano: 16/ 12 / 2024

O documento está sujeito a registro de patente? [ ] Sim [X] Não

O documento pode vir a ser publicado como livro e/ou artigo? [ ] Sim [X] Não

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O/A referido/a docente e/ou autor/a declara que:

1 - o documento é seu trabalho original, detém os direitos autorais da produção técnica e não infringe os direitos de qualquer outra pessoa ou entidade;

2 - obteve autorização de quaisquer materiais inclusos no documento do qual não detém os direitos de autor/a, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia Goiano os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;

3 - cumpriu quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia Goiano.

Rio Verde, 16 de dezembro de 2024.

(Assinado Eletronicamente)

Matheus Wagner dos Santos Martins (Autor)

(Assinado Eletronicamente)

Douglas Cedrim Oliveira (Orientador)

1058004

(Assinatura do Docente, Autor e/ou Detentor dos Direitos Autorais)

Documento assinado eletronicamente por:

- **Douglas Cedrim Oliveira**, **PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 16/12/2024 13:26:03.
- **Matheus Wagner dos Santos Martins**, **2018202201940046 - Discente**, em 16/12/2024 13:29:04.

Este documento foi emitido pelo SUAP em 16/12/2024. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse https://suap.ifgoiano.edu.br/autenticar-documento/ e forneça os dados abaixo:

Código Verificador: 662416
Código de Autenticação: 156deacd12

INSTITUTO FEDERAL GOIANO
Campus Rio Verde
Rodovia Sul Goiana, Km 01, Zona Rural, 01, Zona Rural, RIO VERDE / GO, CEP 75901-970
(64) 3624-1000

SERVIÇO PÚBLICO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA GOIANO

Ata nº 75/2024 - GGRAD-RV/DE-RV/CMPRV/IFGOIANO

ATA DE DEFESA DE TRABALHO DE CURSO

Aos três dias do mês de dezembro de dois mil e vinte e quatro, às dezesseis horas, reuniu-se a banca examinadora composta pelos docentes: Dr. Douglas Cedrim Oliveira (orientador), Dr. Vinícius Ruela Pereira Borges (membro externo), Me. Fábio Montanha Ramos (membro interno), para examinar o Trabalho de Conclusão de Curso intitulado: "Análise e visualização de séries temporais de dados de mídias sociais durante a pandemia de COVID-19", de Matheus Wagner dos Santos Martins, estudante do Curso de Bacharelado em Ciência da Computação do IF Goiano – Campus Rio Verde, sob Matrícula nº 2018202201940046. A palavra foi concedida ao estudante para a apresentação oral do TCC, em seguida houve arguição do candidato pelos membros da Banca Examinadora. Após tal etapa, a Banca Examinadora decidiu pela APROVAÇÃO do estudante. Ao final da sessão pública de defesa foi lavrada a presente ata, que segue assinada pelos membros da Banca Examinadora, onde a assinatura do membro externo Vinícius Ruela Pereira Borges é feita eletronicamente pelo orientador.

(Assinado Eletronicamente)

Douglas Cedrim Oliveira

Orientador(a)

(Assinado Eletronicamente pelo orientador)

Vinícius Ruela Pereira Borges

Membro

(Assinado Eletronicamente)

Fábio Montanha Ramos

Membro

Observação:

( ) O(a) estudante não compareceu à defesa do TC.

Documento assinado eletronicamente por:

- **Douglas Cedrim Oliveira**, **PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 03/12/2024 17:15:21.
- **Fabio Montanha Ramos**, **PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 13/12/2024 11:30:33.

Este documento foi emitido pelo SUAP em 03/12/2024. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse https://suap.ifgoiano.edu.br/autenticar-documento/ e forneça os dados abaixo:

Código Verificador: 657737
Código de Autenticação: 0c691a26ed

INSTITUTO FEDERAL GOIANO
Campus Rio Verde
Rodovia Sul Goiana, Km 01, Zona Rural, 01, Zona Rural, RIO VERDE / GO, CEP 75901-970
(64) 3624-1000

RESUMO

MARTINS, Matheus. **Análise e Visualização de Séries Temporais de Dados de Mídias Sociais Durante a Pandemia de COVID-19**. Dezembro, 2024. 68 f. Monografia – (Curso de Bacharel em Ciência da Computação), Instituto Federal de Educação, Ciência e Tecnologia Goiano - Campus Rio Verde. Rio Verde, GO.

Este trabalho explora as dinâmicas de comunicação durante a pandemia de COVID-19 utilizando um vasto conjunto de dados do Twitter. Com mais de 524 milhões de publicações coletadas entre fevereiro e maio de 2020, abrangendo 62 idiomas e 218 países, o estudo se concentra em analisar e visualizar esses dados para entender melhor as respostas regionais e culturais à pandemia. Utilizamos métodos de visualização matricial, como mapas de calor, para identificar padrões e tendências nas publicações. Além disso, aplicamos a técnica de Distorção Dinâmica do Tempo (DTW) para comparar séries temporais, medindo a distância entre diferentes sequências de dados que podem variar em velocidade e alinhamento. Os resultados revelaram insights significativos sobre o comportamento das publicações relacionadas à COVID-19, incluindo picos de interesse em resposta a eventos específicos e a adoção global de mensagens de saúde pública.

Palavras-chave: Visualização de Dados. Mídias Sociais. Distorção Dinâmica do Tempo. COVID-19.

ABSTRACT

MARTINS, Matheus. **Analysis and Visualization of Twitter Time Series Data During the COVID-19 Pandemic**. Dezembro, 2024. 68 f. Monografia – (Curso de Bacharel em Ciência da Computação), Instituto Federal de Educação, Ciência e Tecnologia Goiano - Campus Rio Verde. Rio Verde, GO.

This study explores the dynamics of communication during the COVID-19 pandemic using a vast dataset from Twitter. With over 524 million posts collected between February and May 2020, covering 62 languages and 218 countries, the research focuses on analyzing and visualizing these data to better understand regional and cultural responses to the pandemic. We employ matrix visualization methods, such as heatmaps, to identify patterns and trends in the tweets. Additionally, we apply the Dynamic Time Warping (DTW) technique to compare time series, measuring the distance between different data sequences that may vary in speed and alignment. The results revealed significant insights into the behavior of COVID-19-related posts, including peaks of interest in response to specific events and the global adoption of public health messages.

Keywords: Data Visualization. Social Media. Dynamic Time Warping. COVID-19.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

API	Interface de Programação de Aplicação (*Application Programming Interface*, no Inglês)
ASCII	Código Padrão Americano para Intercâmbio de Informações (*American Standard Code for Information Interchange*, no Inglês)
CEP	Código de Endereçamento Postal
CoV-2	Coronavírus 2
CSV	Valores Separados por Delimitador (*Character-separated values*, no Inglês)
COVID-19	Doença do Coronavírus 19
DTW	Distorção Dinâmica do Tempo (*Dynamic Time Warping*, no Inglês)
GPS	Sistema de Posicionamento Global (*Global Positioning System*, no Inglês)
ISO	Organização Internacional para Padronização (*International Organization for Standardization*, no Inglês)
JSON	Notação de Objeto do Javascript (*Javascript Object Notation*, no Inglês)
OMS	Organização Mundial da Saúde
SARS	Síndrome Respiratória Aguda Grave
URL	Localizador Uniforme de Recursos (*Uniform Resource Locator*, no Inglês)
XML	Linguagem de Marcação Extensível (*Extensible Markup Language*, no Inglês)

LISTA DE QUADROS

LISTA DE ALGORITMOS

SUMÁRIO

1 INTRODUÇÃO

Geralmente, grandes conjuntos de dados de alta dimensão são matrizes onde as linhas são amostras e as colunas são variáveis medidas, cada coluna pode ser definida como uma dimensão do espaço que contém os dados. Visualizar esse conjunto de dados é um desafio, uma vez que é necessário reduzir a dimensionalidade para tornar os dados visualmente interpretáveis para humanos, processo esse em que podem haver perdas, e, também, tem alto custo computacional (PROBST; REYMOND, 2020).

De acordo com Stieglitz et al. (2018), a quantidade massiva de dados gerados diariamente é comumente citado como um dos maiores desafios enfrentados por pesquisadores e cientistas de dados. Antes da ascenção da Internet, dados estatísticos consistiam em um conjunto de valores observados de uma ou mais variáveis. Os valores podem representar uma amostra em um determinado momento, uma sequência ao longo do tempo de uma ou várias séries temporais, ou uma sequência de dados espaciais em diferentes locais.

Técnicas de análise de Big Data adequadas combinadas com a visualização podem levar a uma melhor consciência situacional e decisões preditivas (BHATTARAI et al., 2019). No contexto contemporâneo os dados muitas vezes são gerados de forma automática, por sensores, redes sociais e outros dispositivos com frequência e periodicidade diferentes, e incluem, em sua maioria, dados não estruturados, como textos, vídeos, imagens, entre outros (GALEANO; PEÑA, 2019). Portanto, há uma necessidade cada vez maior da aplicação de técnicas de análise e visualização da informação mais robustas e eficientes.

Os 5 Vs de Big Data são princípios fundamentais que ajudam a entender as características e os desafios associados ao gerenciamento de grandes volumes de dados. Eles incluem o volume, que se refere à enorme quantidade de dados gerados e armazenados, variando de terabytes a petabytes, exigindo tecnologias robustas para armazenamento e processamento. A velocidade diz respeito à rapidez com que os dados são gerados, coletados e processados, especialmente em tempo real, o que exige infraestrutura ágil para lidar com o fluxo contínuo de informações. A variedade indica a diversidade dos tipos de dados disponíveis, que podem incluir dados estruturados, semiestruturados e não estruturados, como texto, áudio, vídeo e redes sociais. A veracidade trata da qualidade e confiabilidade dos dados, que precisam ser precisos e consistentes para que as análises baseadas neles sejam confiáveis e eficazes. Por fim, o valor refere-se ao potencial dos dados de gerar insights valiosos para negócios e organizações, transformando grandes volumes de dados brutos em informações úteis e acionáveis para a tomada de decisões (FURHT; VILLANUSTRE, 2016).

A pandemia de COVID-19 trouxe desafios sem precedentes para a sociedade global, afetando profundamente a maneira como as pessoas vivem, trabalham e se comunicam. Em um contexto onde a informação se tornou uma ferramenta essencial para o enfrentamento da

crise, o conceito de Big Data ganha relevância, permitindo a análise de vastas quantidades de dados gerados diariamente por diversos dispositivos e plataformas. Também segundo Furht e Villanustre (2016), a capacidade de coletar, processar e interpretar esses dados é crucial para a tomada de decisões informadas em diferentes áreas, como saúde pública, economia e políticas sociais.

As mídias sociais desempenham um papel fundamental nesse cenário, funcionando como um canal vital para a disseminação de informações e para a compreensão do comportamento coletivo. A análise de dados provenientes dessas plataformas permite identificar padrões de comunicação, medir o impacto de políticas públicas e entender as reações da população em tempo real (STIEGLITZ et al., 2018). As técnicas de visualização de dados, por sua vez, são ferramentas poderosas que transformam dados complexos em representações visuais intuitivas, facilitando a interpretação e a análise dos mesmos.

A hipótese deste trabalho é que é possível visualizar padrões de comportamento durante o período da pandemia de COVID-19 utilizando dados de redes sociais. Com o objetivo de verificar essa hipótese, foi utilizado um vasto conjunto de dados extraídos do Twitter para explorar as dinâmicas de comunicação durante a pandemia de COVID-19. A base de dados, coletada por Qazi, Imran e Ofli (2020), abrangeu 524.353.432 publicações em 62 idiomas distintos e de 218 países, oferecendo uma visão abrangente e diversificada das conversas globais sobre a pandemia. A diversidade linguística e a abrangência geográfica deste conjunto de dados permitem uma análise detalhada das respostas regionais e culturais ao longo do tempo.

Para visualizar esses dados, utilizamos métodos de visualização matricial, com ênfase na criação de mapas de calor (*heatmaps*). Esses mapas de calor são ferramentas eficazes para identificar padrões e tendências nas publicações do Twitter, destacando áreas de alta e baixa atividade. A técnica de Distorção Dinâmica do Tempo (DTW) foi empregada para comparar séries temporais, permitindo medir a distância entre diferentes sequências de dados que podem variar em velocidade e alinhamento (SALVADOR; CHAN, 2007).

Os resultados obtidos revelaram insights significativos sobre o comportamento das publicações relacionadas à COVID-19. Foi possível identificar picos de interesse e atividade em resposta a eventos específicos, como novas políticas de isolamento. Além disso, a análise de termos em diferentes idiomas mostrou como as mensagens de saúde pública foram adotadas globalmente, transcendendo barreiras linguísticas e culturais. Estes resultados destacam a importância de uma abordagem multidisciplinar que combina Big Data, análise de mídias sociais e técnicas de visualização para compreender tendências em crises globais como a pandemia de COVID-19.

O restante desse trabalho foi dividido nos seguintes capítulos:

- **Capitulo 2:** Apresenta de maneira abrangente os conceitos e fundamentos que constituem a base teórica essencial para o desenvolvimento deste trabalho;
- **Capítulo 3:** Oferece uma análise de trabalhos relacionados à proposta deste estudo. Examinando literaturas relevantes dentro da área de visualização da informação;
- **Capítulo 4:** Detalha os materiais e métodos empregados durante o desenvolvimento do trabalho;
- **Capítulo 5:** Apresenta a análise detalhada dos resultados obtidos por meio da implementação proposta;
- **Capítulo 6:** Resume os resultados alcançados e destaca possíveis melhorias e direções para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Big Data

O Big Data representa a vasta quantidade de informação gerada diariamente através dos mais diversos dispositivos eletrônicos e o tratamento analítico dessa informação através de diversas ferramentas Tecnológicas, com o intuito de se obter padrões, correlações e percepções que podem auxiliar em tomadas de decisões nas mais diversas áreas (GALDINO, 2015).

Grande parte dos cientistas e especialistas definem Big Data pelas cinco principais características a seguir (denominadas 5Vs) (FURHT; VILLANUSTRE, 2016):

- **Volume**: Refere-se à enorme quantidade de dados que são gerados, coletados e processados, no tamanho de petabytes, exabytes e zettabytes. Por exemplo, o Twitter recebe/processa milhões de *tweets* regularmente. Da mesma forma, o Facebook lida rotineiramente com milhões de postagens e imagens. De modo análogo, o Google recebe mais de um bilhão de buscas e consultas. Além disso, milhões de registros de dados são coletados de tecnologias de sensores associadas a transporte, clima, sistemas ambientais, etc (YOUNAS, 2019);

- **Velocidade**: A velocidade é considerada um atributo chave do Big Data. Em vez de os dados serem amostrados ocasionalmente (seja de forma pontual ou com um grande intervalo temporal entre as amostras), o Big Data é produzido muito mais continuamente. Existem dois tipos de velocidade em relação ao Big Data: frequência de geração; frequência de manuseio, gravação e publicação (KITCHIN; MCARDLE, 2016);

- **Variedade**: Variedade refere-se à heterogeneidade estrutural em um conjunto de dados. Os avanços tecnológicos permitem que as empresas usem vários tipos de dados estruturados, semiestruturados e não estruturados. Os dados estruturados, que constituem apenas 5% de todos os dados existentes (CUKIER, 2010), referem-se aos dados tabulares encontrados em planilhas ou bancos de dados relacionais. Texto, imagens, áudios e vídeos são exemplos de dados não estruturados, que às vezes carecem da organização estrutural exigida pelas máquinas para análise. O formato de dados semiestruturados não está em conformidade com padrões rígidos, o XML (*Extensible Markup Language*, no Inglês), uma linguagem textual para troca de dados na Web, é um exemplo típico de dados semiestruturados (GANDOMI; HAIDER, 2015);

- **Veracidade**: Refere-se à qualidade e à confiabilidade dos dados. Com grandes volumes de dados provenientes de diversas fontes, a precisão e a veracidade dos dados podem variar significativamente, tornando crucial a capacidade de validar e limpar os dados para análise eficaz (WANG; STRONG, 1996);

- **Valor**: Refere-se ao valor potencial que pode ser extraído dos dados. O foco não está apenas em coletar e armazenar dados, mas em analisá-los para obter insights significativos que possam beneficiar as organizações (SCHMARZO, 2013).

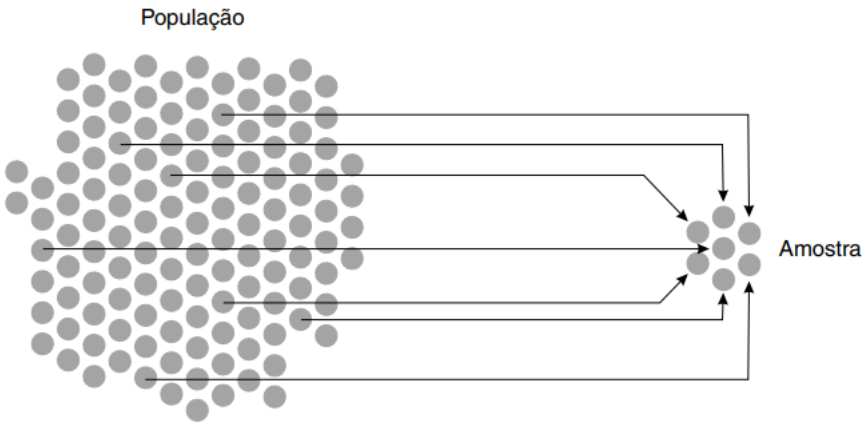
Também segundo Furht e Villanustre (2016), os desafios na implementação de soluções de Big Data encontram-se em diferentes níveis, incluindo: captura de dados, armazenamento, pesquisa, compartilhamento, análise, gerenciamento e visualização. Além disso, existem problemas de segurança e privacidade, especialmente em dados distribuídos.

2.1.1 Redução de dados

Para manipular e visualizar grandes conjuntos de dados, os sistemas modernos precisam lidar com problemas de sobrecarga de informações. Oferecer escalabilidade visual é crucial na visualização de Big Data. Nesse sentido, uma grande variedade de sistemas usa técnicas de aproximação (também conhecidas como técnicas de redução de dados), nas quais conjuntos abstratos de dados são calculados.

Dentre as abordagens existentes para redução de dados, a maioria delas é baseada em: amostragem (como ilustrado na Figura 1) e filtragem e/ou agregação (BIKAKIS, 2018).

Figura 1 – Amostragem.



Fonte: Santos (2007).

Uma “amostra” é um subconjunto da população, selecionado de modo a ser um

representante da população maior. As técnicas de amostragem são amplamente classificadas em amostragem probabilística e não probabilística (ACHARYA et al., 2013).

- **Amostragem probabilística**: permite ao investigador generalizar os resultados da amostra para a população-alvo. A amostragem probabilística inclui amostragem aleatória simples, Amostragem aleatória sistemática, Amostragem aleatória estratificada, Amostragem por agrupamento, entre outros;

- **Amostragem não probabilística**: A amostragem não probabilística inclui conveniência/amostragem intencional, amostragem por cota, amostragem por bola de neve, entre outrtras.

Acharya et al. (2013) destaca que cada método de amostragem tem suas próprias vantagens e limitações, no entanto, a amostragem probabilística é preferível, uma vez que seus resultados podem ser generalizados.

2.2 Visualização da informação

A visualização é o uso da representação visual de dados suportada por computador. Ao contrário dos dados estáticos, a visualização interativa de dados permite que os usuários especifiquem o formato usado na exibição. Algumas técnicas bastante comuns de visualização incluem o gráfico linear, que mostra a relação entre os itens e pode ser usado para comparar as mudanças ao longo de um período de tempo; o gráfico de barras, que é utilizado para comparar quantidades de diferentes categorias; e o gráfico de dispersão, que é um gráfico bidimensional que mostra a variação de dois itens (SADIKU et al., 2016).

2.2.1 Séries temporais

A visualização de séries temporais refere-se à prática de representar dados que variam ao longo do tempo em formas visuais, facilitando a identificação de tendências, padrões e anomalias. Esta capacidade é fundamental em áreas como ciências sociais, economia, medicina e ciência dos dados, onde técnicas visuais não apenas facilitam a interpretação de tendências, mas também permitem a detecção de anomalias e eventos significativos (KEIM et al., 2008).

Tal prática desempenha um papel crucial na análise exploratória e na compreensão de padrões complexos em dados que evoluem ao longo do tempo. Segundo Cleveland (1993), a representação gráfica de dados temporais é uma ferramenta poderosa para a exploração e compreensão de padrões temporais.

2.2.2 Comparação de séries temporais

Para comparar séries temporais, podemos utilizar uma gama de métricas de comparação vetorial, que podem ser divididas em dois tipos (HAN; PEI; TONG, 2022):

- **Medidas de similaridade**: As medidas de similaridade avaliam o quão semelhantes dois vetores são. Valores altos indicam alta similaridade, enquanto valores baixos indicam baixa similaridade. Um exemplo é a similaridade de cossenos;

- **Medidas de dissimilaridade**: As medidas de dissimilaridade, por outro lado, avaliam o quão diferentes dois vetores são. Valores baixos indicam alta similaridade (ou baixa dissimilaridade), enquanto valores altos indicam baixa similaridade (ou alta dissimilaridade). A distância euclidiana e o DTW são exemplos de medidas de dissimilaridade.

2.2.2.1 Similaridade de cossenos

É uma medida de semelhança entre dois vetores diferentes de zero de um espaço interno do produto que mede o cosseno do ângulo entre eles. O cosseno de 0° é 1 e é menor que 1 para qualquer ângulo no intervalo (0, π] radianos . Portanto, é um julgamento de orientação e não de magnitude: dois vetores com a mesma orientação têm uma semelhança de cosseno de 1 , dois vetores orientados a 90° um em relação ao outro têm uma semelhança de 0 e dois vetores diametralmente opostos têm uma semelhança de -1, independentemente de sua magnitude (SINGHAL, 2001).

A similaridade de cosseno é usada particularmente no espaço positivo, onde o resultado é bem delimitado em [0,1]. O nome deriva do termo "direção cosseno": nesse caso, os vetores unitários são no máximo "semelhantes" se forem paralelos e no máximo "diferentes" se forem ortogonais (perpendiculares). Isso é análogo ao cosseno, que é a unidade (valor máximo) quando os segmentos subtendem um ângulo zero e zero (não correlacionado) quando os segmentos são perpendiculares (SINGHAL, 2001).

Esses limites se aplicam a qualquer número de dimensões e a semelhança de cosseno é mais comumente usada em espaços positivos de alta dimensão. Por exemplo, na recuperação de informações e na mineração de texto , cada termo recebe uma dimensão diferente e um documento é caracterizado por um vetor em que o valor em cada dimensão corresponde ao número de vezes que o termo aparece no documento. A similaridade do cosseno fornece uma medida útil da probabilidade de dois documentos serem similares em termos de assunto (SINGHAL, 2001).

De acordo com Tan, Steinbach e Kumar (2005), a técnica também é usada para medir a coesão dentro de clusters no campo de mineração de dados.

$$cos(\mathbf{p}, \mathbf{q}) = \frac{\mathbf{p}.\mathbf{q}}{|\mathbf{p}|\,|\mathbf{q}|}, \tag{1}$$

onde $\mathbf{p}.\mathbf{q}$ é o produto interno entre os dois vetores $\mathbf{p}$ e $\mathbf{q}$.

2.2.2.2 Distância euclidiana

Considerada como uma medida de dissimilaridade, a distância euclidiana é interpretada como a distância entre dois indivíduos, cujas posições são determinadas em relação às suas coordenadas, definidas com referência a um grupo de eixos cartesianos, os quais possuem ângulos retos entre si (CLIFFORD; STEPHENSON, 1975).

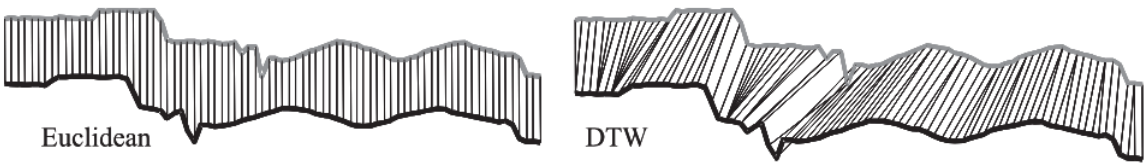
Em um espaço Euclidiano de n dimensões, onde os vetores $\mathbf{p}$ e $\mathbf{q}$ têm coordenadas $\mathbf{p} = (p_1,p_2,p_3,...,p_n)$ e $\mathbf{q} = (q_1,q_2,q_3,...,q_n)$, o cálculo pode ser feito da seguinte forma:

$$d(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2} \tag{2}$$

A métrica possui as seguintes propriedades (HAN; PEI; TONG, 2022):

- **Simetria**: A distância euclidiana é a mesma em qualquer direção. Ou seja, d($\mathbf{p}$,$\mathbf{q}$)=d($\mathbf{q}$,$\mathbf{p}$);
- **Não-negatividade**: É sempre um número não negativo e é igual a zero se, e somente se, os dois vetores forem iguais;
- **Variante no tempo**: O cálculo da distância é feito ponto a ponto, ou seja, não considera deformações que ocorrem em sequências diferentes em ambos os vetores, como é visto na Figura 2.

Figura 2 – Distância euclidiana comparada com DTW.



Fonte: Keogh e Ratanamahatana (2005).

2.2.2.3 Distorção dinâmica do tempo (DTW)

O algoritmo de Distorção dinâmica do tempo (*Dynamic Time Warping*, no Inglês) é utilizado para comparar séries temporais e mede a dissimilaridade entre duas sequências,

permitindo deformações não lineares no eixo do tempo. Isso é particularmente útil quando as séries temporais podem estar fora de fase ou ter diferentes velocidades, como em reconhecimento de fala, análise de gestos e séries temporais financeiras (SALVADOR; CHAN, 2007).

O DTW encontra o alinhamento ótimo entre duas séries temporais, minimizando a soma das distâncias acumuladas ao longo de um caminho através de uma matriz de custo. Ele permite que pontos de uma série sejam mapeados para pontos de outra série, mesmo que eles não estejam alinhados no tempo. O processo pode ser descrito pelos seguintes passos:

- **Matriz de custo**: Uma matriz de custo $n.m$ é criada, onde n e m são os comprimentos das duas séries temporais **a** e **b**. Cada elemento da matriz representa o custo de alinhar o ponto $\mathbf{a}_i$ com o ponto $\mathbf{b}_j$. O custo pode ser calculado usando uma medida de distância, geralmente, é calculado como o quadrado da distância euclidiana (KEOGH; RATANAMAHATANA, 2005);

$$\mathbf{CUSTO}(i,j) = d(\mathbf{a}_i,\mathbf{b}_j) \tag{3}$$

- **Cálculo do caminho de deformação**: O caminho de deformação é determinado através da matriz de custo. O objetivo é encontrar o caminho de menor custo desde a célula (1,1) até (n,m). O custo total de deformação é calculado somando os custos individuais ao longo do caminho, ou seja:

$$\mathbf{D}(i,j) = \mathbf{CUSTO}(i,j) + min(\mathbf{D}(i-1,j),\mathbf{D}(i,j-1),\mathbf{D}(i-1,j-1)), \tag{4}$$

  onde $\mathbf{D}(i,j)$ é o custo acumulado até a célula (i,j).

O valor retornado pelo DTW é o valor da célula $\mathbf{D}(n,m)$, que representa o custo acumulado mínimo necessário para alinhar toda a série **a** com a série **b**, como visto na Equação 4. Esse valor pode ser interpretado como uma medida de dissimilaridade entre duas séries temporais, ou seja, quanto menor, mais semelhante.

O DTW possui diversas vantagens, entre elas (SALVADOR; CHAN, 2007):

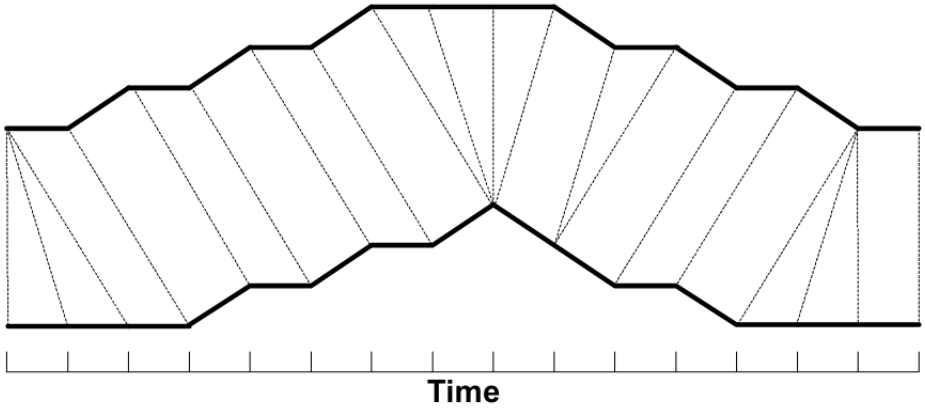
- **Robustez a Ruído**: Eficaz mesmo quando há ruído ou flutuações locais nas séries;

- **Aplicabilidade Geral**: Usado em reconhecimento de padrões, processamento de sinais, e mineração de dados;

- **Invariante no tempo**: É capaz de lidar com variações na velocidade dos eventos nas séries temporais, como é visto na Figura 3, onde as séries temporais são representadas

pelas linhas em negrito.

Figura 3 – Deformação entre duas séries temporais com DTW.



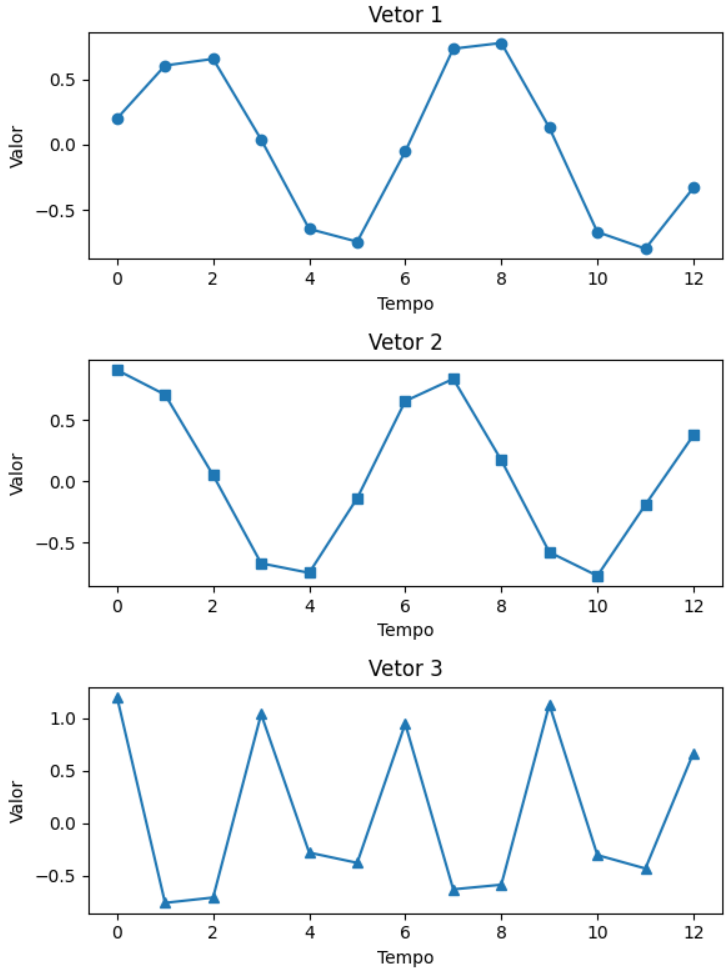
Fonte: Salvador e Chan (2007).

A complexidade da versão básica do algoritmo DTW é $O(n.m)$, já que é necessário construir uma matriz onde cada elemento é preenchido usando uma relação recursiva. Portanto, o número de operações realizadas é proporcional ao produto dos comprimentos das duas séries temporais.

Existem algumas otimizações que podem ser aplicadas para reduzir a complexidade, como o uso de janelas de restrição proposto por Sakoe e Chiba (1978), que limita o número de células a serem consideradas na matriz de custo. Dessa forma, em vez de calcular toda a matriz, apenas uma faixa diagonal de largura r é considerada.

Em síntese, o DTW é um algoritmo poderoso para a comparação de séries temporais, oferecendo vantagens significativas sobre métodos tradicionais devido à sua capacidade de lidar com variações temporais e desalinhamentos. O comportamento do DTW é mostrado na Figura 4.

Figura 4 – Séries temporais sintéticas para ilustrar o comportamento do DTW. A maior similaridade entre as duas primeiras indicará um menor valor do DTW, sendo o contrário com a última.



Fonte: Autor.

O algoritmo retorna uma medida de distância, onde valores menores indicam maior semelhança entre os vetores. As duas primeiras séries são mais semelhantes entre si, com deformações parecidas ocorrendo em momentos levemente diferentes, destacando a invariância temporal do algoritmo. Já a última série é significativamente diferente das demais. A distância calculada entre os vetores 1 e 2 é de aproximadamente 1.03, enquanto as distâncias entre o vetor 3 e o vetor 1 e entre o vetor 3 e o vetor 2 são de 2.47 e 2.13, respectivamente.

2.2.3 Mapa de calor

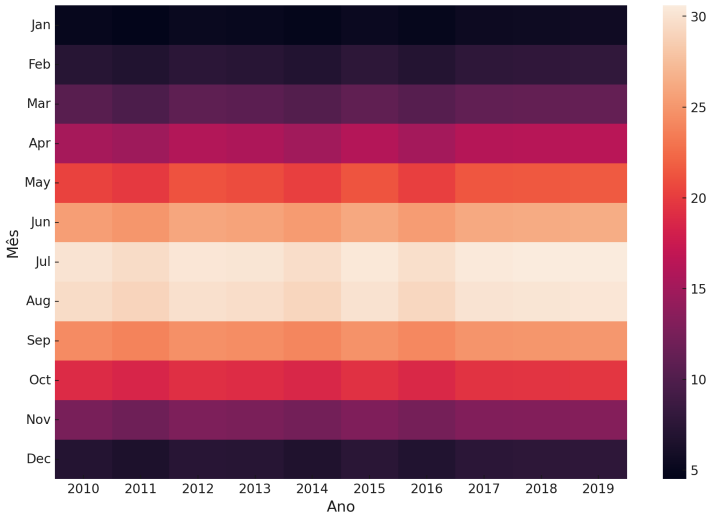
Mapas de calor (*heatmaps*) são representações gráficas que utilizam cores para ilustrar a magnitude de valores em duas dimensões, facilitando a identificação de padrões, tendências e áreas de concentração em grandes conjuntos de dados. Esse tipo de visualização

é especialmente útil em diversas áreas, como estatística, análise de dados, biologia e marketing, onde a interpretação rápida e intuitiva dos dados é crucial (TUFTE, 2001).

A construção de um mapa de calor envolve uma grade de células, onde cada célula é colorida de acordo com seu valor numérico. As cores variam em intensidade, proporcionando uma indicação visual imediata das variações de dados. Por exemplo, em um mapa de calor que mostra a frequência de visitas a uma página web ao longo do tempo, cores mais quentes (como vermelho) podem indicar áreas de alta atividade, enquanto cores mais frias (como azul) indicam baixa atividade.

Abaixo, há um exemplo de mapa de calor com dados fictícios que representam as médias de temperatura de uma cidade ao longo dos meses, de 2010 até 2019. A barra do lado direito indica a escala dos valores das células: temperaturas mais altas são exibidas com cores mais claras, enquanto temperaturas mais baixas são exibidas com cores mais escuras.

Figura 5 – Mapa de calor com médias de temperatura.



Fonte: Autor.

Percebemos que a variação de cores torna fácil detectar quando ocorreram determinadas temperaturas, o que faz do mapa de calor uma ótima forma de visualizar as informações nesse contexto.

2.3 Técnicas e estruturas de processamento de dados

2.3.1 API

Uma API, ou Interface de Programação de Aplicações, é uma forma de interação entre diferentes softwares, permitindo que eles comuniquem entre si sem intervenção direta do usuário. Essas interfaces são comuns na internet e facilitam a integração e a manipulação

de diferentes serviços e dados através de uma série de chamadas e respostas (ALHOSAINI et al., 2024).

No contexto científico, as APIs são muito utilizadas para a mineração de dados, permitindo a extração e análise de grandes volumes de dados de maneira automatizada. Por exemplo, a API do Twitter é amplamente usada para coletar dados para pesquisas em diversas áreas, como marketing digital, comportamento humano e eventos atuais, possibilitando análises profundas sobre tendências e padrões de interação dos usuários (MODI et al., 2024).

2.3.2 Estruturas de dados e armazenamento

2.3.2.1 Dicionário

Os dicionários em Python são estruturas de dados que armazenam pares de chave-valor. Cada chave é única e é utilizada para acessar seu valor associado. Os dicionários são mutáveis, permitindo a adição, remoção e modificação de itens em tempo constante. Segue um exemplo prático do uso de um dicionário para a contagem de termos de uma lista:

Algoritmo 1: Exemplo de uso de um dicionário.

```
def contar_ocorrencias(lista):
    contador = {}
    for elemento in lista:
        if elemento in contador:
            contador[elemento] += 1
        else:
            contador[elemento] = 1
    return contador

lista = ['maca', 'banana', 'maca', 'laranja', 'banana', 'maca']
ocorrencias = contar_ocorrencias(lista)
print(ocorrencias) # Saida: {'maca': 3, 'banana': 2, 'laranja': 1}
```

Fonte: Autor.

Cada termo da lista é adicionado ao dicionário como uma chave única com o valor inicial de 1, que é incrementado a cada nova ocorrência desse termo. O dicionário é a principal estrutura de dados utilizada neste trabalho, por ser compatível com o formato JSON, permitindo que as informações sejam convertidas e persistidas mais facilmente.

2.3.2.2 JSON

JSON (*JavaScript Object Notation*) é um formato leve de intercâmbio de dados, fácil para humanos lerem e escreverem, e fácil para máquinas analisarem e gerarem. Desenvolvido originalmente para trabalhar com JavaScript, JSON é independente de linguagem, tornando-se uma escolha popular para transferência de dados em aplicações web. Ele é utilizado para transmitir dados entre um servidor e um cliente como texto simples, estruturado de uma maneira que imita a notação de objetos em JavaScript (PEZOA et al., 2016).

A estrutura do JSON é composta por pares chave-valor, onde as chaves são strings e os valores podem ser strings, números, arrays, objetos, booleanos ou nulos. Esta simplicidade e flexibilidade permitem representar estruturas de dados complexas de forma concisa e legível. Por exemplo, um objeto JSON pode ser usado para representar dados de um usuário com suas informações de nome, idade e endereço.

2.3.2.3 CSV

CSV (*Comma-Separated Values*) é um formato de arquivo simples e amplamente utilizado para armazenar dados tabulares, onde cada linha do arquivo representa um registro e os valores de cada campo são separados por vírgulas. Devido à sua simplicidade e facilidade de uso, o formato CSV é comum para a transferência de dados entre sistemas diferentes, especialmente quando a interoperabilidade é necessária. Arquivos CSV podem ser gerados e lidos por uma ampla variedade de softwares, incluindo planilhas eletrônicas, bancos de dados e ferramentas de análise de dados.

Cada linha em um arquivo CSV corresponde a uma linha de uma tabela, e cada campo é separado por uma vírgula (ou outro delimitador, como ponto e vírgula, dependendo da regionalização). A primeira linha, opcionalmente, pode conter os cabeçalhos das colunas, fornecendo nomes descritivos para cada campo.

2.3.3 Normalização de dados

A normalização de dados é uma técnica amplamente utilizada em várias áreas, como aprendizado de máquina, processamento de sinais e estatística. O objetivo principal da normalização é ajustar os dados para que eles tenham uma escala comum, facilitando a comparação entre diferentes variáveis e evitando que uma variável com uma escala maior domine outras em modelos de análise ou em algoritmos de aprendizado (JAMES et al., 2013).

Existem várias formas de normalização, cada uma adequada a diferentes contextos e necessidades. As mais comuns incluem a normalização *min-max*, que ajusta os valores para uma faixa entre 0 e 1, e a normalização *z-score*, que transforma os dados para que tenham média zero e desvio padrão um. Ambas as técnicas são úteis em diferentes situações;

por exemplo, a normalização *min-max* é frequentemente usada quando a distribuição dos dados é desconhecida, enquanto o *z-score* é preferido quando se assume que os dados seguem uma distribuição normal (HASTIE; TIBSHIRANI; FRIEDMAN, 2009).

Outra abordagem importante é a normalização utilizando a norma de um vetor. A norma vetorial é uma medida que quantifica o "tamanho" ou "comprimento" de um vetor em um espaço vetorial. A vantagem de normalizar os valores de um vetor usando sua norma é que isso permite comparar vetores de diferentes magnitudes em uma escala comum. A normalização é feita dividindo cada componente do vetor pela norma do vetor, resultando em um vetor unitário, ou seja, com norma igual a 1 (STRANG, 2016).

2.4 Padrões e códigos

2.4.1 Expressões regulares

Expressões regulares, ou regex, são padrões utilizados para buscar ou manipular textos. Elas são compostas por uma série de caracteres e símbolos especiais que formam um padrão de busca. Entre os principais componentes estão os literais, que correspondem exatamente a si mesmos, e os metacaracteres, que possuem significados especiais como "." para qualquer caractere, "*" para zero ou mais ocorrências do padrão precedente, e "[]" para conjuntos de caracteres.

Podemos utilizar expressões regulares para diversas tarefas, como validação de formatos específicos (por exemplo, endereços de e-mail, números de telefone, CEPs), extração de informações (como datas, URLs e números), substituição de padrões em textos (por exemplo, trocar todas as ocorrências de uma palavra por outra), e divisão de strings em substrings com base em padrões específicos. Elas são amplamente usadas em linguagens de programação, editores de texto e ferramentas de busca, oferecendo uma maneira poderosa e flexível de lidar com grandes volumes de dados textuais de forma eficiente e precisa.

2.4.2 Unicode

Unicode é um sistema de codificação de caracteres que visa representar texto em todos os sistemas de escrita do mundo de forma consistente e uniforme. Ele atribui um número único (ponto de código) a cada caractere, independente da plataforma, programa ou idioma. O Unicode suporta mais de 143.000 caracteres, incluindo letras, números, símbolos, emojis e caracteres de scripts antigos e modernos. Esse padrão permite a interoperabilidade entre diferentes sistemas e é amplamente utilizado na informática e na comunicação digital para garantir que texto e dados sejam exibidos corretamente em qualquer dispositivo ou software.

2.4.3 ASCII

ASCII é um sistema de codificação de caracteres utilizado principalmente em computadores e dispositivos de comunicação para representar texto. Ele utiliza números de 0 a 127 para codificar caracteres, incluindo letras do alfabeto inglês (maiúsculas e minúsculas), dígitos, pontuação e alguns caracteres de controle. Por exemplo, a letra 'A' é representada pelo número 65 e o caractere de espaço pelo número 32.

Apesar de ser uma das primeiras codificações de caracteres, ainda é amplamente utilizado devido à sua simplicidade e compatibilidade com sistemas antigos. No entanto, por ser limitado a 128 caracteres, não é suficiente para representar caracteres de muitos outros idiomas e símbolos especiais, motivo pelo qual padrões mais abrangentes como Unicode são utilizados atualmente.

2.4.4 ISO 3166-1:2020

A ISO é uma organização internacional não governamental que desenvolve e publica padrões globais para uma ampla gama de indústrias e setores. Esses padrões visam garantir a qualidade, segurança, eficiência e interoperabilidade de produtos, serviços e sistemas, facilitando o comércio internacional e promovendo melhores práticas em diversas áreas.

O ISO 3166-1:2020 é parte do padrão ISO 3166. Esta parte define códigos para a representação dos nomes de países e de suas subdivisões. Especificamente, o padrão estabelece códigos de dois e três caracteres para os nomes dos países e territórios dependentes.

3 TRABALHOS RELACIONADOS

Este capítulo apresenta uma revisão dos principais trabalhos relacionados às técnicas e metodologias de visualização e análise de dados que são relevantes para a presente pesquisa. A seguir, serão discutidos estudos que exploram grafos dinâmicos, visualizações matriciais e a análise de mídias sociais, cada um com suas respectivas abordagens e contribuições para o campo.

3.1 Grafos dinâmicos

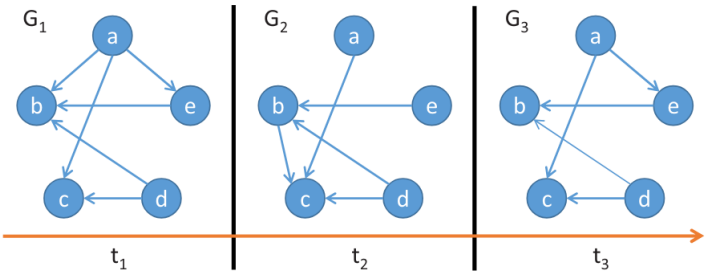
Grafos dinâmicos são representações gráficas que modelam a evolução de estruturas de redes ao longo do tempo. Eles estendem os conceitos dos grafos estáticos ao incorporar a dimensão temporal, permitindo a análise de como as conexões e interações entre os nós da rede mudam. A aplicação de grafos dinâmicos é crucial em várias disciplinas, como análise de redes sociais, bioinformática, e sistemas de comunicação, onde a estrutura da rede é sujeita a mudanças contínuas (BECK et al., 2017).

Um grafo dinâmico pode ser introduzido como um grafo estático $G=(V,E)$, onde V representa os vértices e E as arestas. Então, um grafo dinâmico é definido da seguinte forma (BECK et al., 2017):

$$\Gamma = (G_1,G_2,...,G_n)\,, \tag{5}$$

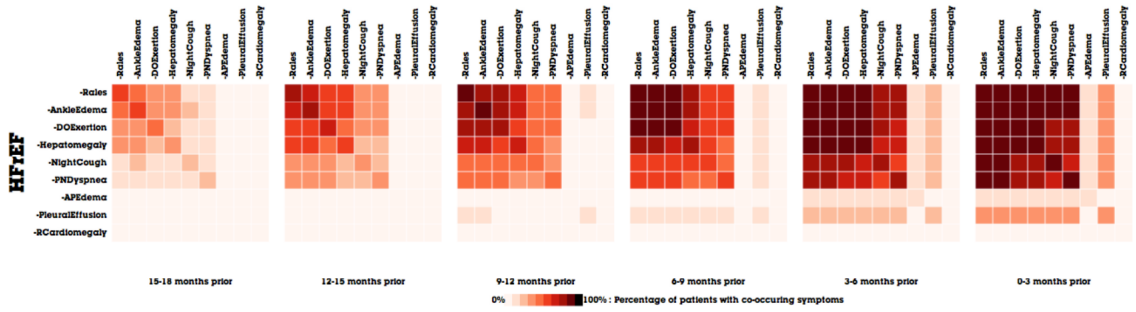
onde $G_i = (V_i,E_i)$ são grafos estáticos cujos índices se referem a sequências temporais $t = (t_1,t_2,...,t_n)$. Na literatura encontram-se diversas formas de representação de grafos dinâmicos, entre elas, podemos destacar o diagrama de nós ligados 6 e a matriz de adjacência 7.

Figura 6 – Diagrama de nós ligados.



Fonte: Beck et al. (2017).

Figura 7 – Matriz de adjacência.



Fonte: Perer e Sun (2012).

Diversos algoritmos surgiram com o intuito de melhorar o layout e a disposição dos nós dos grafos dinâmicos para facilitar a descoberta de padrões e comportamentos, dentre eles, temos o modelo direcionado por força, utilizado por Kobourov (2013), que simula forças físicas entre os nós. Também podemos destacar:

- **Layout ortogonal**: onde os vértices são plotados apenas ao longo do eixo horizontal e vertical (GÖRG et al., 2005);

- **Layout hierárquico**: que divide o grafo em camadas (AHMED et al., 2010).

O modelo de layout direcionado por força sofreu diversas melhorias ao longo do tempo, como é visto em Eades (1983) e Fruchterman e Reingold (1991). A forma mais básica da abordagem pode ser definida como a força exercida ao longo das arestas da rede, para um nó i, chamada S_i, e a repulsão entre todos os outros nós, R_i, calculada da seguinte forma:

$$S_i = \sum_{j \in N_i} \frac{\|x_j - x_i\| \, (x_j - x_i)}{k}, \tag{6}$$

$$R_i = \sum_{j=1 \ldots n, j \neq i} k^2 \frac{x_i - x_j}{\|x_i - x_j\|^2}, \tag{7}$$

onde N_i é o conjunto de nós adjacentes ao nó i, n é o número total de nós no grafo, x_i é o vetor de posições do nó i e k é o parâmetro livre que representa a distância ótima do layout, controlando o espaçamento entre os nós. O movimento dos nós é calculado por (FRISHMAN; TAL, 2008):

$$v_i = d \left(\frac{S_i + R_i}{M} \right), \tag{8}$$

onde v é a velocidade do nó i, d é o fator de deslocamento, que fixa os nós ou não (0=fixo, 1=livre), e $M=1$ é a massa do nó, que é fixa para todos os nós do grafo.

Segundo Fruchterman e Reingold (1991), esses algoritmos também enfrentam desafios, como a tendência a produzir layouts que podem ser sensíveis às condições iniciais, resultando em layouts que não preservam o mapa mental e a demanda computacional significativa para grandes grafos.

Métodos como ajustes dinâmicos e otimizações paralelas são frequentemente empregados para mitigar essas questões, como é feito por Gorochowski, Bernardo e Grierson (2012), que implementam uma melhoria no modelo padrão de layout baseado em força, que consiste em preservar a informação de nós que são mantidos e retirados do grafo em cada sequência através de um parâmetro chamado "idade", que influencia diretamente no cálculo das forças, contribuindo para um layout mais robusto e facilitando a detecção de padrões.

O mapa mental refere-se à capacidade dos usuários de manter uma representação mental estável de uma visualização gráfica enquanto ela evolui. Isso significa que a posição relativa dos nós e arestas deve ser mantida o mais constante possível para evitar confusão e facilitar o rastreamento das mudanças ao longo do tempo. No que se refere a importância do mapa mental para a visualização, podemos destacar segundo Archambault, Purchase e Pinaud (2010):

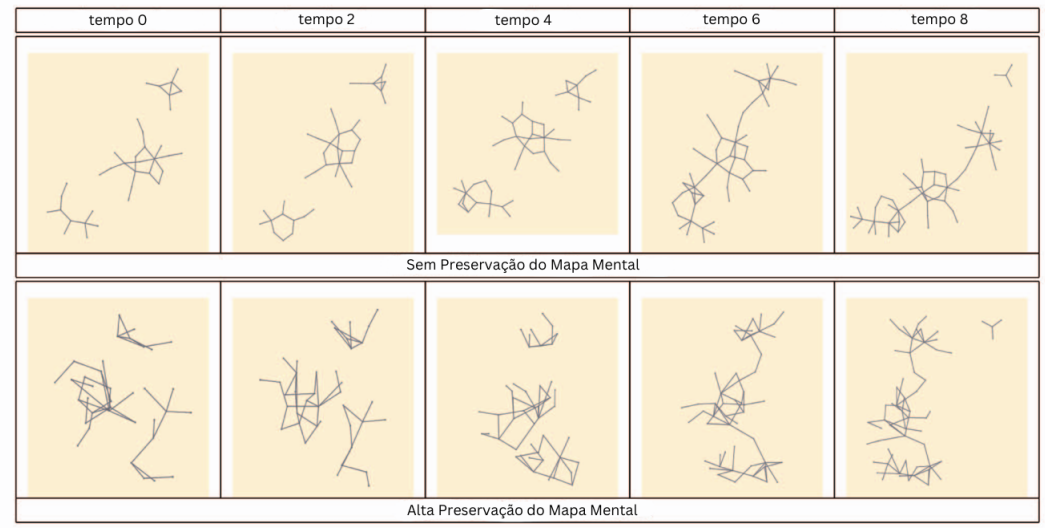
- Manter uma disposição constante dos elementos gráficos permite que os usuários identifiquem facilmente mudanças e padrões emergentes;

- Reduz a carga cognitiva, pois os usuários não precisam reaprender a estrutura do grafo a cada atualização;

- A precisão em tarefas como a identificação de novos nós ou arestas é significativamente melhorada quando o mapa mental é preservado;

- Resulta em menos erros, pois a posição dos elementos varia pouco entre os quadros.

Figura 8 – Preservação do mapa mental.



Modificado de: Archambault, Purchase e Pinaud (2010).

Vários autores destacam que, para alcançar uma boa preservação do mapa mental, a estratégia de visualização deve seguir ou cumprir as seguintes restrições:

- **Intertemporalidade**: Utilizar arestas intertemporais para ligar os mesmos nós entre diferentes momentos, ajudando a manter a posição relativa dos nós (ARCHAMBAULT; PURCHASE; PINAUD, 2010);

- **Rigidez das Arestas**: Ajustar a rigidez das arestas intertemporais para controlar o grau de preservação do mapa mental, balanceando entre estabilidade e flexibilidade (DIEHL; GÖRG, 2002);

- **Minimização de Movimento**: Utilizar algoritmos que minimizem o movimento dos nós entre atualizações, mantendo a forma geral do grafo consistente ao longo do tempo (DIEHL; GÖRG, 2002);

- **Layout Orientado por Força**: Aplicar métodos baseados em força que posicionam os nós de maneira a minimizar a sobreposição e maximizar a clareza visual (ERTEN et al., 2003);

- **Transições Suaves**: Implementar transições suaves entre os estados do grafo para facilitar o acompanhamento das mudanças pelos usuários (ARCHAMBAULT; PURCHASE; PINAUD, 2010);

- **Uso de Cores e Destaques**: Utilizar cores e destaques para indicar claramente as

modificações no grafo, como nós ou arestas novos, removidos ou alterados (PURCHASE; GÖRG; HOGGAN, 2006);

- **Flexibilidade na Adição e Remoção de Dados**: Permitir a adaptação dinâmica à medida que novos dados são adicionados ou removidos, mantendo a coerência estrutural do grafo (ARCHAMBAULT; PURCHASE; PINAUD, 2010);

- **Hierarquias e Clusters**: Manter a disposição hierárquica ou em clusters dos nós para facilitar a compreensão da estrutura do grafo ao longo do tempo (GÖRG et al., 2005).

Em suma, a visualização de grafos dinâmicos é interessante para séries temporais, por todos os motivos citados anteiormente. Todavia, em casos que o mapa mental da estrutura de grafos não é preservado corretamente entre as transições, uma estratégia de visualização matricial pode ser utilizada.

3.2 Fluxo matricial

O trabalho de Perer e Sun (2012) apresenta uma abordagem inovadora para a visualização e análise de redes temporais no contexto da evolução de sintomas durante a progressão de doenças. A ferramenta proposta utiliza uma visualização matricial para rastrear e analisar como os sintomas evoluem e interagem ao longo do tempo, oferecendo insights valiosos para profissionais da saúde e pesquisadores.

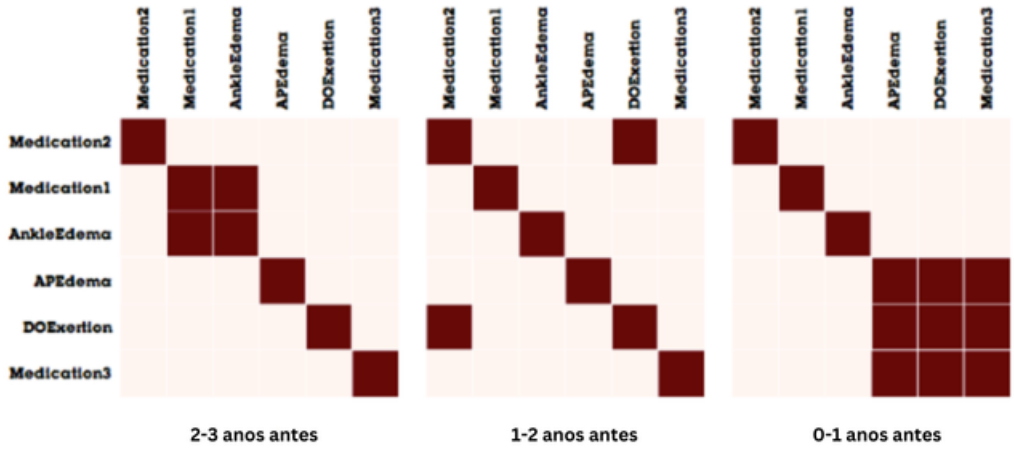
A visualização matricial foi escolhida por várias razões. Primeiramente, a visualização em forma de matriz permite uma comparação mais direta e eficiente das coocorrências de eventos ao longo do tempo, isso é especialmente útil em análises que requerem a observação de padrões e clusters temporais. Além disso, a matriz oferece um método mais intuitivo para visualizar grandes volumes de dados, facilitando a detecção de padrões emergentes e a correlação entre diferentes eventos clínicos ao longo do tempo.

3.2.1 Dados utilizados

O sistema utiliza sequências de eventos clínicos que evoluem até o diagnóstico de insuficiência cardíaca. Esses dados são extraídos de registros médicos eletrônicos e incluem informações detalhadas sobre sintomas, diagnósticos, medicações e outros eventos clínicos relevantes. A abordagem de coleta de dados é essencial para modelar redes de eventos que evoluem com o tempo e para capturar as nuances das trajetórias de doenças nos pacientes.

As matrizes são construídas através de um processo detalhado. Primeiro, uma rede de eventos é modelada a partir dos dados clínicos, onde cada nó representa um evento clínico. As conexões entre os nós são determinadas pela coocorrência desses eventos dentro de intervalos de tempo especificados, que são ajustáveis conforme a necessidade da análise.

Figura 9 – Eventos clínicos.

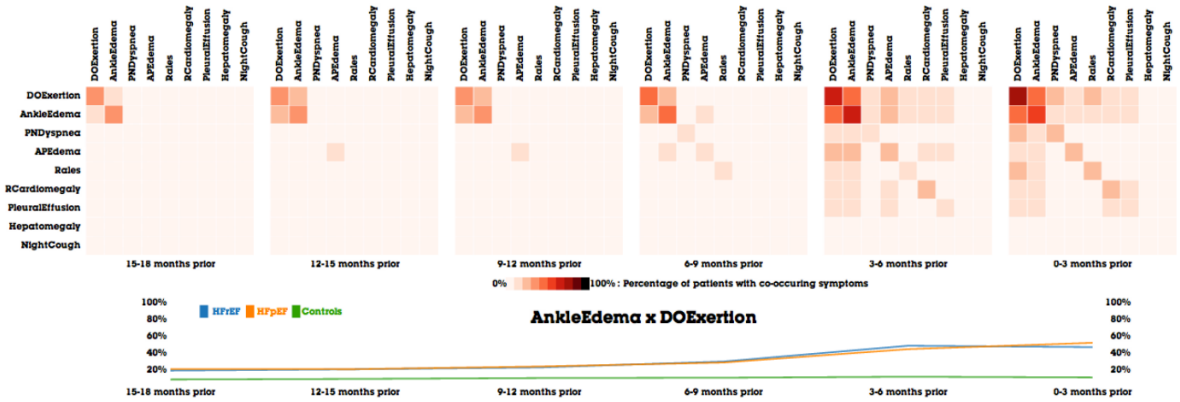


Modificado de: Perer e Sun (2012).

Para organizar eficientemente os dados na matriz, é empregado um algoritmo de agrupamento hierárquico que otimiza uma métrica conhecida como modularidade de Newman (NEWMAN, 2004). Esse método agrupa eventos que frequentemente coocorrem, minimizando a distância entre nós conectados na visualização matricial. Isso resulta em uma representação que destaca claramente os clusters de eventos, facilitando a interpretação visual dos padrões de dados.

A matriz é então visualizada na forma de um mapa de calor (*heatmap*, em inglês), cujas cores representam o peso das conexões, indicando o número de coocorrências dos eventos em relação ao total de pacientes, ou seja, uma porcentagem.

Figura 10 – Visualização com pesos.



Fonte: Perer e Sun (2012).

A matriz resultante permite uma análise detalhada das interações temporais entre sintomas e medicamentos em um grupo de pacientes. Profissionais da saúde podem usar

essa ferramenta para observar como os sintomas se desenvolvem e interagem ao longo do tempo, identificar períodos críticos e ajustar tratamentos.

É mostrado que a técnica funciona bem para dados temporais cuja evolução pode ser rastreada até um evento final, que foi, no caso, o diagnóstico de insuficiência cardíaca. Esse foco permitiu aos profissionais de saúde visualizar como os eventos anteriores, como sintomas e tratamentos, se interconectam e contribuem para o resultado do diagnóstico.

3.3 Mídias sociais

Usamos o termo mídia social para nos referir a "aplicações baseadas na Internet que se baseiam nos fundamentos ideológicos e tecnológicos da Web 2.0", onde "Web 2.0" significa que "o conteúdo e as aplicações não são mais criados e publicados por indivíduos, mas são continuamente modificados por todos os usuários de forma participativa e colaborativa" (KAPLAN; HAENLEIN, 2010).

A quantidade massiva de dados gerados diariamente em redes sociais é comumente citada como um dos maiores desafios enfrentados por pesquisadores e cientistas de dados que buscam extrair algum conhecimento útil acerca de um determinado tópico ou entender o comportamento de uma comunidade (STIEGLITZ et al., 2018).

Análise de mídias sociais é um eixo de pesquisa focado na extração de conhecimentos úteis de dados de mídias sociais, com o objetivo de ajudar indivíduos e organizações a tomar as melhores decisões relativas a problemas de negócios, marketing, política, saúde, entre outros (SEBEI; TAIEB; AOUICHA, 2018). Um crescente corpo de pesquisas e aplicações práticas emprega dados de mídias sociais como proxy para desvendar o comportamento complexo de uma sociedade (BUKOVINA, 2016).

Segundo Stieglitz et al. (2018), o processo de análise de mídias sociais pode ser descrito em quatro passos distintos:

- **Descoberta de dados**: A descoberta de dados envolve a coleta e avaliação de dados de várias fontes e é frequentemente usada para entender tendências e padrões nos dados. Requer uma progressão de etapas que as organizações podem usar como estrutura para entender seus dados;

- **Coleta**: A coleta de dados é um processo utilizado para captar informações geradas pelas pessoas (ou por processos) e que servirão de insumos para planejar estratégias para o negócio. Esses dados podem ser coletados em plataformas específicas para coletas, formulários, sites e outras metodologias;

- **Preparação**: A preparação de dados consiste na limpeza e transformação de dados brutos para que sejam processados e qualificados. Geralmente, envolve a reformatação, a correção e a combinação de conjuntos de informações para que sejam enriquecidas

e validadas;

- **Análise**: A análise de dados é um processo de inspeção, limpeza, transformação e modelagem de dados com o objetivo de descobrir informações úteis, informar conclusões e apoiar a tomada de decisões.

3.3.1 Coleta de dados

Os dados são muitas vezes coletados com ferramentas que se comunicam com a respectiva API da plataforma de mídia social, caso exista. O Twitter é uma rede social muito utilizada em pesquisas pelo fato de possuir uma API muito bem documentada e de fácil acesso, permitindo buscar publicações por hashtags, palavras chave, localização, idioma, entre outros.

Embora o Twitter seja frequentemente utilizado para esse tipo de análise, outras redes sociais, como o Facebook, também oferecem dados valiosos que podem ser explorados para fins semelhantes. O uso de dados do Facebook pode ser um exemplo adicional de como redes sociais podem ser analisadas para entender tendências e comportamentos, como demonstrado em estudos de previsão de surtos, onde o Facebook foi utilizado para modelar e prever a propagação de doenças, como o COVID-19 (KHAYYAT et al., 2021).

As políticas de privacidade do Twitter não permitem que as publicações recuperadas via API sejam disponibilizadas publicamente na íntegra, uma vez que tal prática vai contra os termos de privacidade de usuário da plataforma. Portanto, o único dado relacionado diretamente a publicação disponível em cojuntos de dados públicos é a identificação única do *tweet* (ID). Dessa forma, é necessário acessar a API da plataforma para obter todos os dados da publicação através do ID, que é um processo conhecido como *hidratação* (LAMSAL, 2021).

3.3.2 Conjuntos de dados

Podemos encontrar diversos conjuntos de dados já coletados de mídias sociais construídos para os mais diversos fins, como o de Lamsal (2021), que montou um conjunto de dados massivo contendo mais de trezendos e dez milhões de registros do Twitter no idioma Inglês, com assuntos referentes ao surto da doença causada pelo vírus SARS-CoV-2 (sigla do inglês que significa Coronavírus 2 da Síndrome Respiratória Aguda Grave) cuja doença recebeu a denominação pela Organização Mundial da Saúde (OMS) de COVID-19.

Outro exemplo é o conjunto de dados criado por Qazi, Imran e Ofli (2020), que possui dados de publicações em diversos idiomas. O local associado a publicação pode ser inferido de quatro formas:

- **Coordenadas geográficas**: As coordenadas atuais (latitude e longitude) são recebidas por meio do dispositivo do usuário. Embora o Twitter ofereça a opção de ativar o serviço de geolocalização exata em dispositivos móveis, apenas 1% dos *tweets* coletados contém coordenadas GPS;

- **Campo "Place"**: Um campo opcional que o usuário preenche selecionando uma das localizações sugeridas pelo Twitter. Esse campo associa um *tweet* a uma localização que se refere a uma cidade, uma área ou um ponto de interesse famoso, como um prédio ou um restaurante. Quando presente, o campo "Place" indica que o *tweet* está associado a um lugar, mas pode não ter se originado necessariamente desse lugar;

- **Localização do usuário**: É um texto livre inserido pelo usuário e, portanto, muitas vezes é ruidoso, pois os usuários podem digitar qualquer coisa, por exemplo, "Califórnia, EUA", "Terra", "Marte" e "casa";

- **Conteúdo do *tweet***: O texto da publicação pode ter uma ou mais menções de localização, que podem ser extraídas para a tarefa de inferência de localização, todavia, sofre o mesmo problema do campo "Place", tendo em vista que a publicação pode não ter se originado do local informado.

A tarefa de inferir a localização relacionada a dados do Twitter é conhecida na literatura como "problema da interface de geolocalização", e pode ser dividida em dois subproblemas principais: (i) geolocalização do usuário (localização residencial e atual) e (ii) geolocalização do *tweet* (origem do *tweet* e menções de localizações).

A escolha de publicações com coordenadas geográficas reduz muito a quantidade de dados recuperada, no entanto, permite que análises baseadas em localização sejam mais confiáveis. Quanto ao idioma, Qazi, Imran e Ofli (2020) destacam algumas vantagens em utilizar publicações com múltiplos idiomas:

- **Maior Abrangência e Representatividade**: Captura uma visão mais completa das discussões globais sobre a COVID-19, incluindo comunidades que falam idiomas menos comuns;

- **Análises Culturais e Regionais**: Facilita a análise de como diferentes culturas e regiões respondem à pandemia, permitindo insights mais detalhados e contextualizados;

- **Inclusão e Diversidade**: Garante que diversas vozes e perspectivas sejam incluídas nas análises, promovendo uma abordagem mais inclusiva e equitativa na pesquisa;

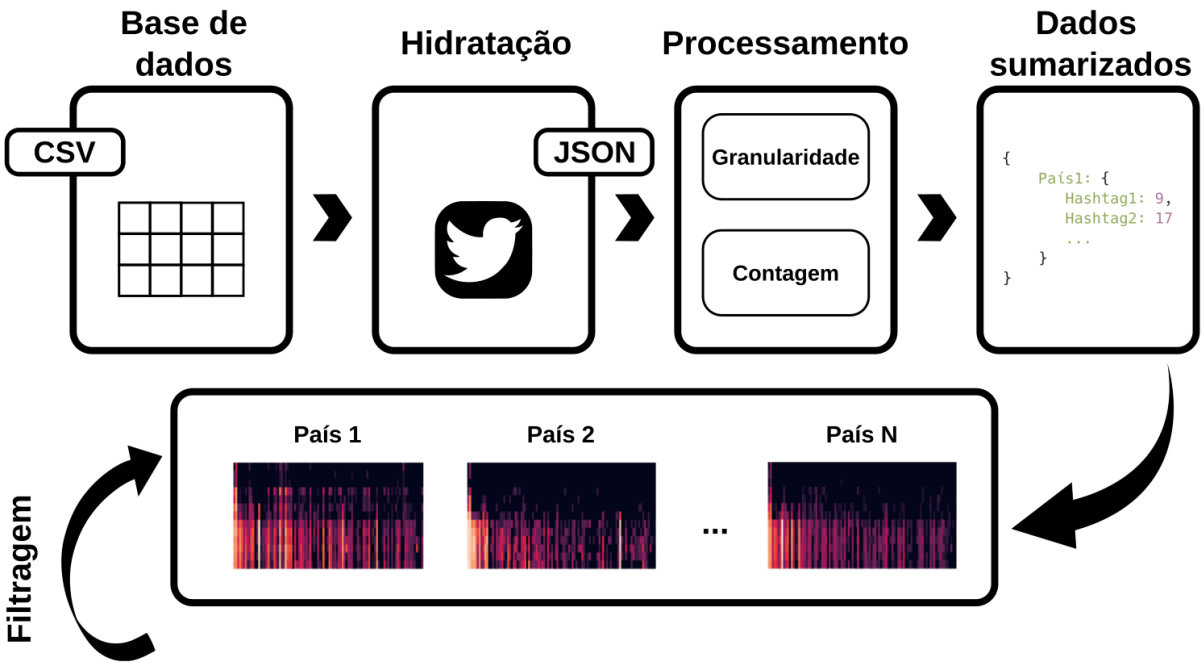
- **Comparação e Contraste**: Permite comparações entre diferentes grupos linguísticos e culturais, ajudando a identificar padrões e variações significativas na resposta à pandemia.

Em resumo, a presença de múltiplos idiomas no conjunto de dados oferece uma visão mais rica e diversificada das conversas sobre a COVID-19, facilitando uma análise mais completa e inclusiva das respostas globais à pandemia.

4 MATERIAIS E MÉTODOS

A Figura 11 resume as etapas de obtenção, processamento e visualização dos dados do Twitter. Cada uma das etapas é detalhada nas próximas seções.

Figura 11 – Fluxograma de processamento e visualização de dados do Twitter.



Fonte: Autor.

O processo inicia com a obtenção de dados brutos de tweets armazenados em uma base de dados no formato CSV, contendo identificadores únicos. Em seguida, ocorre a hidratação desses dados via API do Twitter, que transforma os IDs em informações completas no formato JSON, incluindo texto, hashtags e localização. Na etapa de processamento, os dados são organizados, a granularidade temporal é definida e as contagens de hashtags são realizadas. Os resultados são, então, agregados em dados sumarizados, estruturados por país com suas respectivas contagens. Para a visualização, são gerados *heatmaps* para cada país, oferecendo uma visão geral dos padrões temporais. A partir disso, selecionamos um país e uma hashtag específica e geramos outra matriz ordenada por semelhança com o termo escolhido.

4.1 Conjunto de dados utilizado

Utilizamos o dataset criado por Qazi, Imran e Ofli (2020), baseado no Twitter, cujas características são detalhadas nas próximas seções.

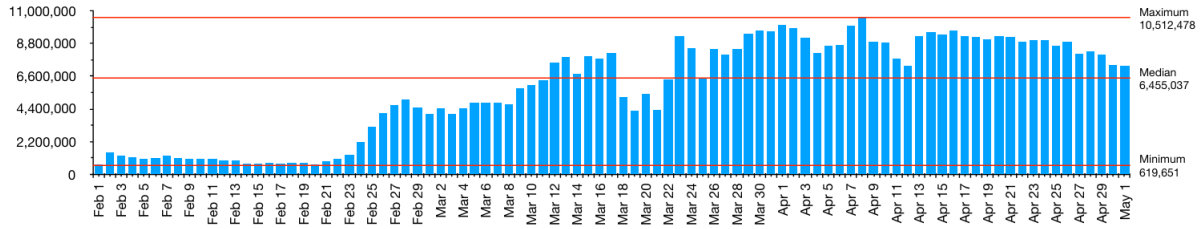
4.1.1 Coleta

A coleta de dados foi realizada através de centenas de hashtags e palavras-chave multilíngues relacionadas à pandemia de COVID-19. A coleta começou em 1º de fevereiro de 2020, usando hashtags em alta, como #covid19, #coronavirus, #covid 19, e novas hashtags foram adicionadas à medida que surgiam nos dias seguintes. Em 1º de maio, o número total de hashtags/palavras-chave era de 803.

4.1.2 Quantidade de publicações

O conjunto possui, no total, 524.353.432 (quinhentos e vinte e quatro milhões, trezentos e cinquenta e três mil, quatrocentos e trinta e dois) *tweets*, realizados no período de 01/02/2020 (primeiro de fevereiro de dois mil e vinte) até 01/05/2020 (primeiro de maio de dois mil e vinte). Cada *tweet* possui, no máximo, 280 caracteres. Na Figura 12, podemos visualizar a distribuição diária das publicações, junto com a média, valor mínimo e máximo.

Figura 12 – Distribuição diária das publicações.

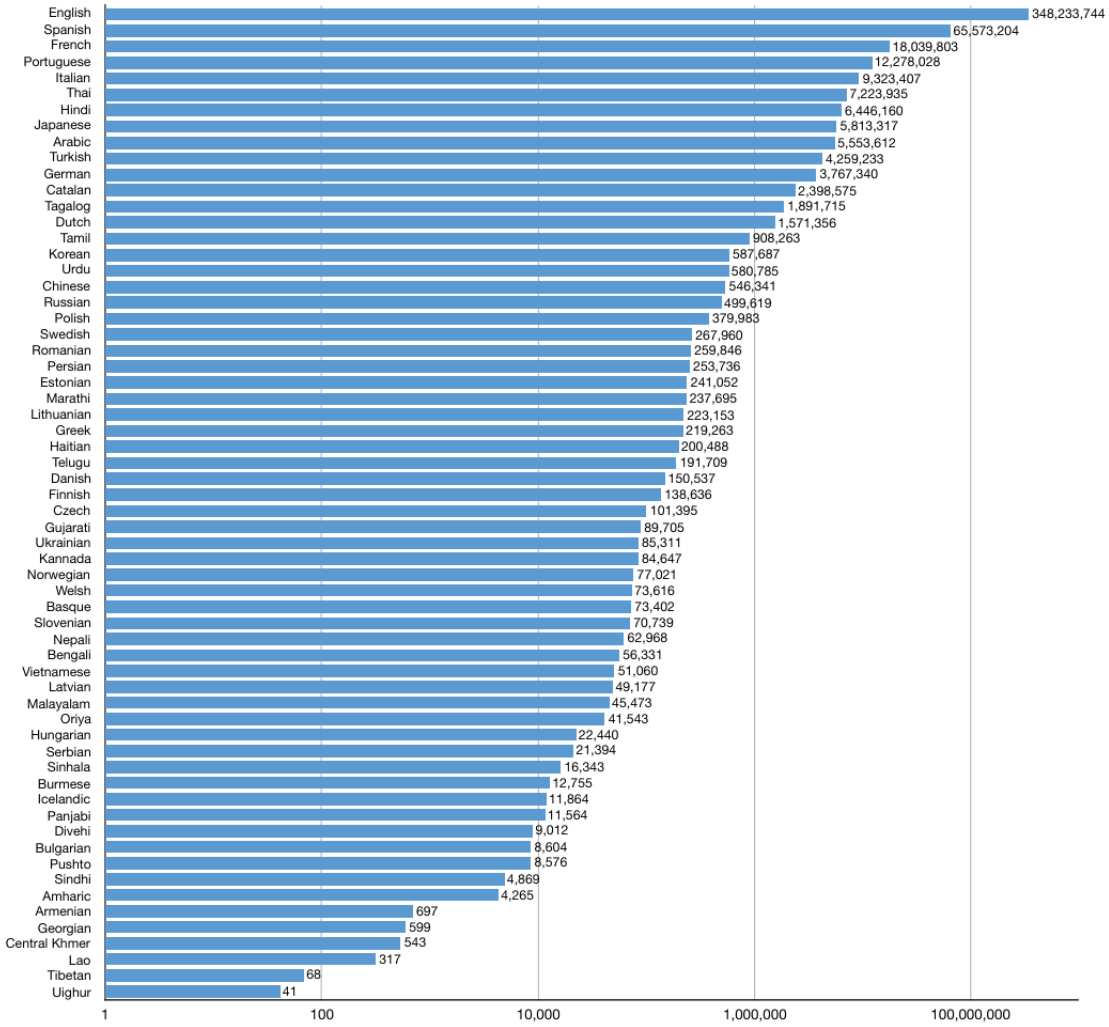


Fonte: Qazi, Imran e Ofli (2020).

4.1.3 Variedade linguística

No total, o conjunto inclui 62 idiomas distintos e 218 países. O Inglês domina, aparecendo em 348 milhões de publicações, o que corresponde a aproximadamente 66,41%. Em seguida, vem o Espanhol e o Francês, correspondendo a 12,5% e 3,44%, respectivamente. A distribuição pode ser observada na Figura 13.

Figura 13 – Distribuição dos idiomas.



Fonte: Qazi, Imran e Ofli (2020).

4.1.4 Recorte

Cada publicação possui até quatro tipos de informação de localização, como foi discutido na Seção 3.3.2. A Tabela 1 mostra os tipos de informação junto com a quantidade de publicações que possuem.

Tabela 1 – Campo de localização e quantidade das publicações que o possuem.

Campo de localização	**Quantidade de publicações**
Coordenadas exatas (latitude e longitude)	378.772
Campo “Place”	5.495.431
Localização do usuário	297.148.292
Texto da publicação	452.933.900

Modificado de: Qazi, Imran e Ofli (2020).

Para o estudo, utilizamos apenas o subconjunto com coordenadas exatas, visto

que os demais podem ser ruidosos e não representar a origem real da publicação.

4.1.5 Formato e estrutura

O dataset está estruturado em formato tabular, contendo duas colunas, que representam, respectivamente:

- **tweet_id**: Identificação única da publicação (número inteiro);
- **user_id**: Identificação única do usuário (número inteiro).

Segue uma amostra do conjunto de dados original:

Tabela 2 – Pequena amostra do conjunto de dados público.

tweet_id	user_id
1256223753220034566	916540973190078465
1256223748904161280	697426379583983616
1256223744122593287	1277481013
1256223753463365632	596005899
1256223753115238406	139159502
1256223748161757190	1655021437
1256223749214437380	3244990814

Fonte: Autor.

As demais informações devem ser recuperadas através da API, passando por suas regras de negócio. Ou seja, o número de publicações recuperadas pode ser menor do que a quantidade referenciada pelo conjunto de dados original. Para as análises, utilizamos a lista de hashtags retornada pela API, em vez de buscar as palavras-chave diretamente no texto do *tweet*. Isso se deve ao trabalho adicional que seria necessário para implementar um processamento mais complexo para filtrar as palavras-chave do texto.

4.2 Tecnologias

Python

Python é uma linguagem de programação de alto nível, funcional e orientada a objetos, além de ser modular, interpretada e multiplataforma, tendo se tornado uma linguagem de programação famosa pela sua sintaxe fácil e acessível. Umas das vantagens é que ele conta com uma quantidade enorme de bibliotecas, tanto nativas quanto de terceiros, oferecendo ferramentas simples para se trabalhar com processamento de texto, análise de dados e agrupamento. O Python também pode ser usado para criar interfaces gráficas e *web services* (RASCHKA, 2015).

A linguagem foi escolhida para o processo de recuperação e análise dos dados presentes no conjunto, por conter um vasto número de ferramentas para trabalhar com dados e visualização da informação. Utilizamos a versão 3.10.12.

Numpy

A biblioteca Numpy é utilizada para a computação numérica em Python. Ela implementa uma série de operações matemáticas e aritméticas que podem ser aplicadas aos valores e *arrays*, como adição, subtração, multiplicação, divisão, exponenciação e operações de raiz quadrada, todas realizadas de forma vetorizada. Neste trabalho, utilizamos a função "*log10*" para a normalização de valores, "*transpose*" para cálculos de matriz transposta e "*norm*" para o cálculo da norma vetorial. Utilizamos a versão 1.25.1.

MatplotLib

Matplotlib é uma biblioteca fundamental para visualização de dados em Python, amplamente utilizada para criar gráficos estáticos, animados e interativos. Desenvolvida inicialmente por Hunter (2007), Matplotlib fornece uma interface robusta e flexível para gerar uma vasta gama de gráficos, desde simples gráficos de linha e dispersão até gráficos de barras, histogramas e gráficos tridimensionais.

Um dos principais pontos fortes do Matplotlib é sua capacidade de personalização. Usuários podem ajustar praticamente todos os aspectos dos gráficos, incluindo cores, estilos de linha, rótulos, títulos e legendas, garantindo que as visualizações atendam a requisitos específicos e sejam esteticamente agradáveis. Além disso, Matplotlib suporta a criação de gráficos complexos através de suas interfaces orientadas a objetos, permitindo a composição de figuras detalhadas e a inclusão de múltiplas subplots em uma única visualização. Esta é a ferramenta gráfica utilizada para criar todas as visualizações. Utilizamos a versão 3.7.2.

Seaborn

Seaborn é uma biblioteca de visualização de dados baseada na Matplotlib, projetada para simplificar a criação de gráficos estatísticos atraentes e informativos. Integrada com Pandas, ela facilita a manipulação e visualização de dados diretamente de *DataFrames*, oferecendo diversas funções para gráficos de dispersão, gráficos de barras, gráficos de violino e muito mais. Seus temas e estilos embutidos garantem que os gráficos sejam visualmente atraentes com mínima necessidade de personalização adicional (WASKOM, 2021).

Além de gráficos estatísticos básicos, Seaborn é especialmente útil para criar gráficos multivariados, como gráficos de pares e gráficos de facetas, que permitem visualizar relações complexas entre múltiplas variáveis. Essa capacidade, combinada com sua interface de alto nível e facilidade de uso, faz de Seaborn uma ferramenta essencial para cientistas de dados e analistas que buscam transformar dados em insights visuais claros e compreensíveis.

Esta biblioteca é utilizada em conjunto com a matplotlib, para a defnição e configuração dos mapas de calor (ou *heatmaps*). Utilizamos a versão 0.13.2.

Tslearn

Tslearn é uma biblioteca projetada para facilitar o aprendizado de máquina em séries temporais. Ela oferece uma gama de ferramentas para processamento, análise e modelagem de dados de séries temporais, sendo ideal para pesquisadores e profissionais que trabalham com dados sequenciais em diversas áreas, como finanças, saúde, climatologia, entre outros (TAVENARD et al., 2020).

Uma das principais funcionalidades da tslearn é a capacidade de pré-processar dados de séries temporais, incluindo normalização, padronização e interpolação de valores ausentes. Além disso, a biblioteca oferece uma variedade de métodos de clustering, classificação e análise de séries temporais, permitindo a aplicação de técnicas avançadas de aprendizado de máquina nesses dados sequenciais. A integração com Scikit-learn permite que os usuários aproveitem pipelines e validadores cruzados, facilitando a construção e avaliação de modelos.

Um dos algoritmos da Tslearn utilizados nesse trabalho é o de Distorção Dinâmica do Tempo. O DTW é uma técnica de comparação de séries temporais que permite medir a distância entre duas sequências que podem variar em velocidade. Ao contrário de métodos tradicionais que comparam pontos correspondentes de cada série, o DTW encontra o alinhamento ideal que minimiza a distância entre elas, permitindo deformações não lineares no tempo. Utilizamos a versão 0.6.3 da biblioteca.

Unicodedata

A biblioteca unicodedata foi utilizada para realizar a limpeza e pré-processamento das hashtags. Ela fornece uma interface para a base de dados Unicode, permitindo acesso a várias propriedades de caracteres, como nome, categoria, decomposição e valores numéricos. Através dessa biblioteca, é possível normalizar strings para garantir que diferentes representações de caracteres equivalentes sejam tratadas da mesma maneira, verificar a categoria de um caractere (como letra, número, símbolo), obter o nome de um caractere Unicode e vice-versa, entre outras funcionalidades. Isso é especialmente útil ao lidar com textos em múltiplos idiomas e scripts, garantindo que o processamento e a análise sejam feitos de forma consistente e correta. Utilizamos a biblioteca embutida presente na versão 3.10.12 do Python.

4.3 Hidratação

O termo *hidratação* no contexto de análise e recuperação de informações refere-se ao processo de enriquecimento de dados, onde informações incompletas ou identificadores

são transformados em dados completos e utilizáveis (LAMSAL, 2021). Isso é especialmente relevante quando trabalhamos com dados do Twitter, cujos dados disponíveis publicamente apresentam apenas identificadores da publicação, como visto na Seção 4.1.5.

A API do Twitter (https://developer.twitter.com/en/docs/twitter-api) permite que desenvolvedores acessem e interajam com os dados do Twitter de forma automatizada, isso inclui a capacidade de hidratar *tweets*. A API oferece diferentes *endpoints* para recuperar informações detalhadas sobre *tweets*, usuários e outras entidades do Twitter.

Para a hidratação, utilizaremos a biblioteca Tweepy (ROESSLEIN, 2009), na versão 4.14.0, que é uma biblioteca escrita em Python que abstrai o processo de autenticação da API do Twitter, entre outras configurações. No Algoritmo 2, há um exemplo de código Python usando tweepy para hidratar uma lista de IDs de *tweets*.

No código, o *bearer_token* é uma chave usada para autenticação de alto nível, permitindo que você faça consultas públicas sem precisar de um usuário autenticado. A *consumer_key* identifica sua aplicação registrada no Twitter, enquanto a *consumer_secret* é uma chave secreta associada a essa aplicação. O *access_token* é utilizado para acessar os dados de um usuário autenticado, permitindo que a aplicação realize ações em nome desse usuário, como ler ou postar *tweets*. Já o *access_token_secret* funciona em conjunto com o *access_token*, garantindo que a comunicação entre a aplicação e a API do Twitter seja segura e autorizada.

Para recuperar as publicações através de seus IDs, algumas limitações se aplicam. A principal restrição é o limite no número de requisições, que permite 900 requisições por intervalo de tempo (15 minutos). Cada requisição pode recuperar até 100 *tweets* de uma vez.

Algoritmo 2: Hidratação.

```python
import tweepy

# Suas chaves de autenticacao da API do Twitter
bearer_token = 'seu_bearer_token'
consumer_key = 'sua_consumer_key'
consumer_secret = 'seu_consumer_secret'
access_token = 'seu_access_token'
access_token_secret = 'seu_access_token_secret'

# Autenticando na API do Twitter
client = tweepy.Client(
    bearer_token=bearer_token,
    consumer_key=consumer_key,
    consumer_secret=consumer_secret,
    access_token=access_token,
    access_token_secret=access_token_secret
)

# Lista de IDs dos tweets que queremos hidratar
tweet_ids = ['1234567890', '0987654321']

# Hidratar os tweets
tweets = client.get_tweets(ids=tweet_ids)

# Exibir os tweets hidratados
for tweet in tweets.data:
    print(f'ID: {tweet.id} - Texto: {tweet.text}')
```

Fonte: Autor.

Após a hidratação, a quantidade de publicações é reduzida em aproximadamente 15,6%, publicações com nenhuma hashtag ou cujos caracteres não fazem parte do alfabeto latino também foram removidas. A checagem dos caracteres é mostrada no Algoritmo 3, qualquer publicação que não passe nos critérios é desconsiderada.

Algoritmo 3: Verificação de idiomas não latinos.

```python
def is_latin(text):
    # Verifica se todos os caracteres no texto estao no alfabeto latino
    for char in text:
        # Ignora espacos e caracteres de controle
        if not char.isalpha():
            continue

        # Obtem o nome Unicode do caractere
        try:
            char_name = unicodedata.name(char)
        except ValueError:
            return False

        # Verifica se o nome contem 'LATIN'
        if 'LATIN' not in char_name:
            return False
    return True
```

Fonte: Autor.

Abaixo, mostramos a quantidade de publicações restantes após cada etapa de recuperação. No total, houve uma redução de aproximadamente 34,68% em relação ao original.

Tabela 3 – Quantidade de dados após os recortes.

Conjunto	**Quantidade de publicações**	**%**
Original	378.772	100
Hidratados	319.557	84,37
Apenas latinos	311.542	82,25
Com hashtags	247.427	65,32
Diferença do original	**131.345**	**34,68**

Fonte: Autor.

Por fim, os dados hidratados são persistidos em listas no formato JSON, com o título do arquivo sendo o ano, mês e dia das publicações incluídas na lista junto com o primeiro identificador, exemplo: “geo_2020-02-01_1223484435896643584”. Dessa forma, cada arquivo possui informações de apenas um dia específico. Cada objeto possui o formato descrito na Tabela 4.

Tabela 4 – Estrutura e Tipos de Dados dos *Tweets*.

Campo	**Tipo**	**Descrição**
tweet_id	String	Identificador único para o *tweet*.
created_at	String	Data e hora de criação do *tweet* no formato ISO 8601.
geo_source	String	Fonte das informações geográficas do *tweet*.
geo	Objeto	Contém informações geográficas detalhadas.
place	Objeto	Informações adicionais sobre o local.
lang	String	Código do idioma do *tweet*.
text	String	Texto completo do *tweet*.
hashtags	Lista	Hashtags presentes no *tweet*. Pode ser um dicionário vazio ou uma lista com dicionários detalhando cada hashtag.

Fonte: Autor.

As informações persistidas nessa etapa são utilizadas como entrada na fase de pré-processamento.

4.4 Pré-Processamento

Nesta seção, são descritas as etapas essenciais para transformar os dados antes de realizar a análise. As etapas incluem a seleção da granularidade, que divide os arquivos de acordo com intervalos de tempo, como semanas ou meses. Em seguida, ocorre a contagem de hashtags, onde as hashtags das publicações são normalizadas e contabilizadas, resultando em arquivos JSON organizados por períodos de tempo.

4.4.1 Granularidade

Os arquivos gerados na fase de hidratação são processados separadamente, com sua divisão baseada em um parâmetro ajustável que define a granularidade em dias. Por exemplo, ao escolher o valor 7, estaremos realizando uma análise semanal. No Algoritmo 4, podemos visualizar a divisão dos arquivos por intervalo de data.

Algoritmo 4: Divisão de arquivos por intervalo de data.

```
def split_files_per_time(files, interval_in_days):
    # Lista com os grupos
    dir_date_split = []

    # Lista de arquivos de um mesmo grupo
    files_split = []
    next_date = None

    for file in files:
        file_date = get_file_date(file.name)

        if file_date == next_date:
            dir_date_split.append(files_split)
            files_split = []
            next_date = None

        if next_date == None:
            next_date = file_date + timedelta(interval_in_days)

        files_split.append(file)

    # Adiciona os arquivos restantes apos o loop
    else:
        dir_date_split.append(files_split)

    return dir_date_split
```

Fonte: Autor.

O algoritmo recebe uma lista com os nomes dos arquivos gerados na etapa anterior, percorre cada item extraindo a data a partir do nome e agrupa os arquivos em sublistas conforme o intervalo de dias especificado. Após processar todos os arquivos, é retornada uma lista contendo todos os grupos organizados por intervalos de tempo.

Dessa forma, para "*interval_in_days*=7", a primeira sublista conterá os arquivos do dia "01-02-2020" até "07-02-2020", a segunda conterá os arquivos do dia "08-02-2020" até "14-02-2020", e assim por diante.

4.4.2 Contagem de hashtags

Cada grupo de arquivos gerado na etapa anterior é processado separadamente, resultando na contagem de hashtags por país no período definido por cada grupo, cuja estrutura pode ser vista no Quadro 1. O processo inicia com a criação de uma estrutura de dicionário para armazenar essas contagens. Em seguida, os arquivos do grupo são lidos e suas publicações são unidas em uma única lista, contendo todas as publicações do período, cada uma no formato mostrado na Tabela 4.

Para cada publicação na lista, extraímos o nome do país com base nas coordenadas geográficas. Em seguida, adicionamos uma entrada no dicionário com o nome do país. Para cada hashtag presente, criamos uma entrada associada ao país com o valor inicial 1, que é incrementado a cada ocorrência da mesma hashtag.

Cada hashtag passa por uma função de normalização, definida no Algoritmo 5, que realiza o processamento do texto em três etapas:

- **Remoção de caracteres não alfanuméricos**: A função utiliza expressão regular para remover todos os caracteres que não são letras ou números, incluindo os sublinhados (*underscore*). Dessa forma, todos os caracteres especiais e espaços são eliminados;

- **Conversão para minúsculas**: Após a remoção, a função retira quaisquer espaços em branco do início e do fim do texto e converte todos os caracteres restantes para minúsculas. Isso garante que as comparações não sejam sensíveis a diferenças de capitalização;

- **Normalização Unicode e remoção de acentos**: Retorna os caracteres para sua forma mais básica (por exemplo, "é" se torna "e"). Em seguida, o texto é transformado em uma codificação ASCII, ignorando caracteres que não podem ser convertidos para ASCII. Isso remove os acentos e outros sinais diacríticos dos caracteres.

Algoritmo 5: Normalização de texto.

```
from unicodedata import normalize

def normalize_text(source):
    # Removendo caracteres nao alfanumericos
    source = re.sub(r'[\W_]+', '', source).strip().lower()
    # Aplicando normalizacao unicode
    return normalize('NFKD', source).encode('ASCII', 'ignore').decode('ASCII')
```

Fonte: Autor.

Dessa forma, evitamos que palavras como “covid19” e “COVID_19” sejam tratadas como termos diferentes e contabilizadas de forma separada. Cada grupo pré-processado gera um único arquivo em formato JSON, nomeado de acordo com seu intervalo de tempo em dias. Por exemplo, o primeiro grupo pré-processado terá o nome “2020-02-01_to_2020-02-07”.

No Quadro 1, podemos visualizar uma parte do tipo de estrutura gerada nesta etapa. Como o conjunto de dados abrange três meses e estamos trabalhando com os dados semanalmente, ao final teremos 13 arquivos pré-processados, cada um correspondente a uma semana dentro desse período.

Quadro 1 – Estrutura do arquivo pré-processado.

```
{
  "it": {
    "ansa": 3,
    "arte": 1,
    "bologna": 1,
    "debelliamoquestovirus": 1
  },
  "us": {
    "2020problems": 1,
    "aboveaverageliving": 1,
    "acn": 2,
  }
}
```

Fonte: Autor.

A estrutura corresponde ao código de identificação do país, definido pela ISO 3166-1:2020, listando todas as hashtags provenientes daquele país durante a semana e seu número de ocorrências.

4.5 Visualização

Mostramos, a seguir, a criação das matrizes dos países visualizadas por meio de *heatmaps*, que resumem quantitativamente o uso de hashtags ao longo do tempo. A análise inclui a seleção das 100 hashtags mais frequentes. Além disso, a seção abordará a comparação das séries temporais de diferentes países utilizando o algoritmo *Dynamic Time Warping* (DTW).

4.5.1 Matriz do país

O processo de visualização começa criando uma matriz para cada país, que é visualizada por meio de um *heatmap*, resumindo os quantitativos temporais criados na etapa anterior. Cada arquivo gerado na etapa anterior representa uma linha na matriz.

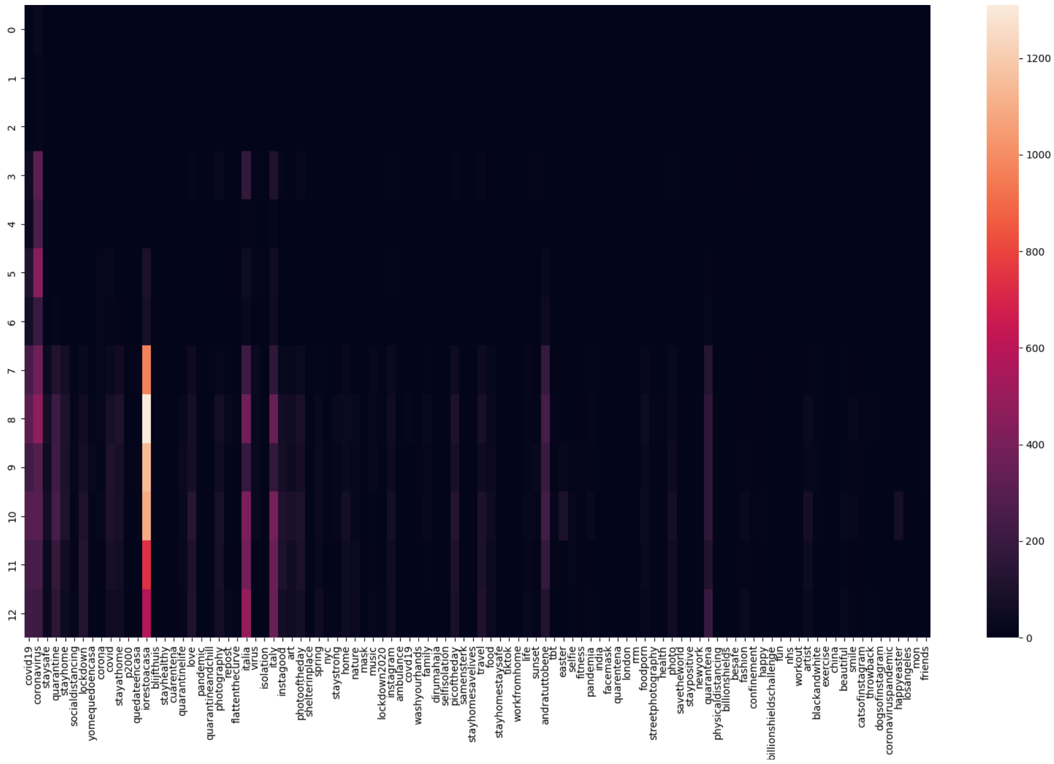
Com a granularidade semanal e o *dataset* abrangendo de 01/02/2020 a 01/05/2020, geramos 13 arquivos, correspondentes a 13 semanas, resultando em uma matriz com 13

linhas (Figura 14). Já com granularidade mensal, seriam 4 arquivos para 4 meses, e a matriz teria 4 linhas (Figura 15).

A estrutura da matriz é período × hashtag, onde cada linha representa o intervalo de tempo selecionado, cada coluna é uma hashtag e os valores são a quantidade de vezes que aquela hashtag associada ao país apareceu naquele intervalo de tempo específico.

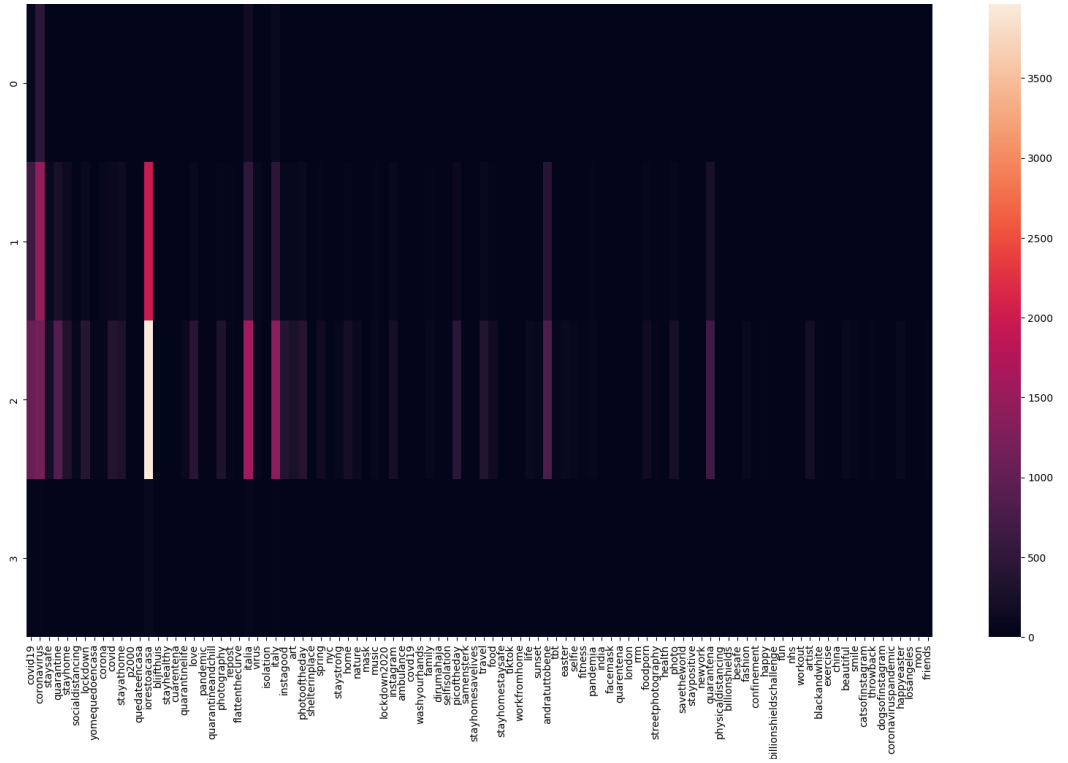
Para construir as matrizes de forma consistente e facilitar a visualização, selecionamos as 100 hashtags mais frequentes em todas as publicações, ordenadas de forma decrescente. Assim, garantimos que as colunas permaneçam as mesmas, independentemente da granularidade escolhida. Nas Figuras 14 e 15, vemos os *heatmaps* gerados para a Itália, em granularidade semanal e mensal, respectivamente.

Figura 14 – Matriz da Itália, resumindo os quantitativos semanais das hashtags.



Fonte: Autor.

Figura 15 – Matriz da Itália, resumindo os quantitativos mensais das hashtags.

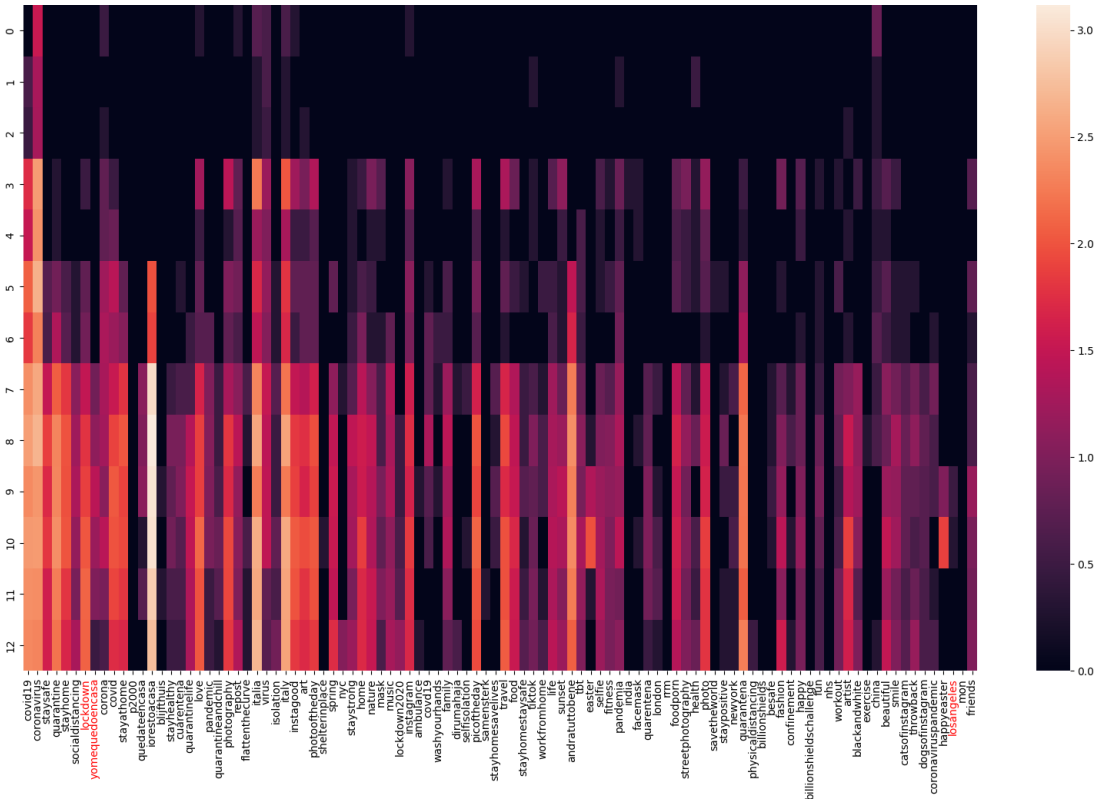


Fonte: Autor.

Cada linha do *heatmap* é marcada com um número inteiro, representando o número da semana ou mês, e as colunas correspondem às 100 hashtags mais frequentes em todas as publicações.

Analisando a escala da Figura 14, podemos ver que os valores variam de 0 a 1200. Isso significa que valores mais altos dominam a visualização, obscurecendo padrões importantes entre os valores menores. Para melhor visualizar pequenas diferenças, mudamos a escala do dado para logarítmica, nesse caso de base 10. O resultado pode ser visto na Figura 16, as hashtags que estão em vermelho tem suas respectivas séries temporais ilustradas na Figura 17.

Figura 16 – Matriz da Itália, resumindo os quantitativos semanais das hashtags com os valores normalizados utilizando *log10*.

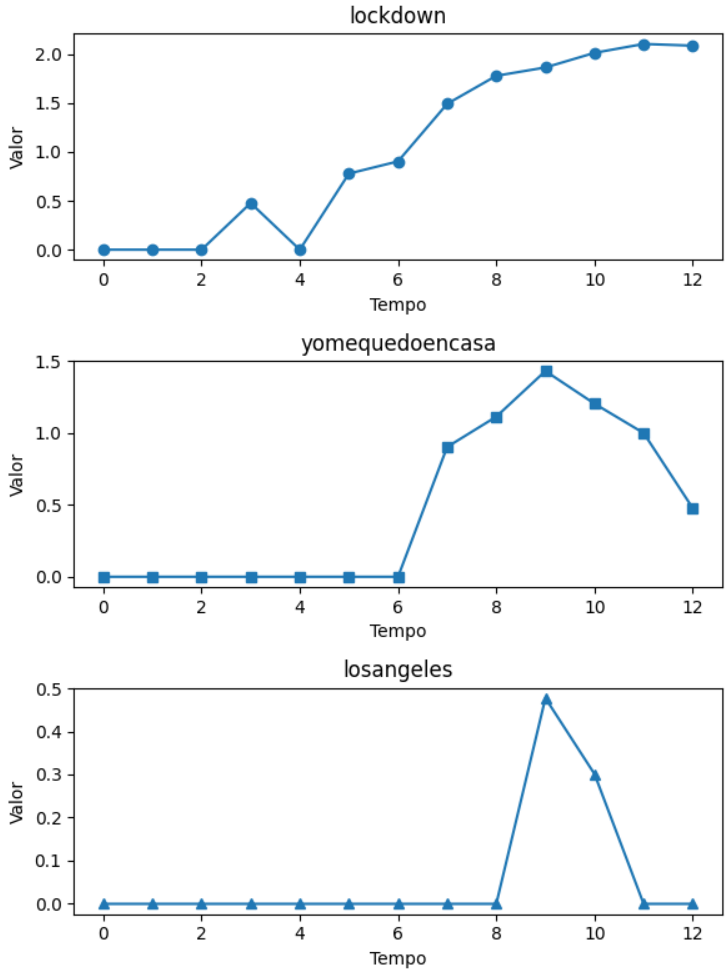


Fonte: Autor.

Após a normalização, a escala foi reduzida, variando de 0 a 3, permitindo a visualização de muito mais variações. Valores altos são mapeados para cores mais claras, enquanto valores baixos são mapeados para cores mais escuras. Dessa forma, podemos observar o comportamento de cada hashtag ao longo da semana (ou qualquer outro intervalo de tempo definido). O processo de construção da matriz pode ser visto no Algoritmo 6.

Na Figura 17, visualizamos as séries temporais das colunas "*lockdown*", "*yomequedoencasa*" e "*losangeles*". Cada ponto do eixo X representa a semana e o eixo Y é composto pelos valores das frequências dos termos. A ordem dos termos na matriz do país não implica necessariamente que dois termos adjacentes sejam semelhantes. Os termos, nesse caso, são ordenados com base no número total de ocorrências em todo o conjunto de dados. Essa abordagem garante que a ordem das colunas seja mantida de forma consistente, mesmo quando utilizamos diferentes granularidades.

Figura 17 – Séries temporais dos termos em vermelho da matriz da Itália em gráfico de linha.



Fonte: Autor.

Algoritmo 6: Construção da matriz de um país.

```
1  def build_country_matrix(steps, country, tags):
2      matrix = []
3
4      for step in steps:
5          # Verifica se o pais tem dados naquela semana
6          if step.get(country):
7              row = [log10(1 + (step[country].get(tag) or 0)) for tag in tags]
8          else:
9              row = [0 for _ in tags]
10
11         matrix.append(row)
12
13     return matrix
```

Fonte: Autor.

O algoritmo recebe três parâmetros: "*steps*", "*country*" e "*tags*". "*steps*" é uma lista onde cada item é um dicionário seguindo a estrutura "país × hashtag × valor", mostrada no Quadro 1, representando cada uma das semanas. "*country*" é o país que desejamos visualizar, e "*tags*" é a lista de hashtags que serão usadas como colunas da matriz. Na linha 6, verificamos se o país possui informações naquela semana; caso contrário, preenchemos a linha com zero. Na linha 7, obtemos o valor da hashtag e substituímos por zero se ela não existir no dicionário, ou seja, se não aparecer naquela semana. Por fim, incrementamos 1 ao valor para evitar erros ao utilizar a função logarítmica.

Cada coluna da matriz representa uma série temporal, que pode ser comparada com séries temporais de outros países para identificar termos que apresentaram comportamentos semelhantes em algum momento.

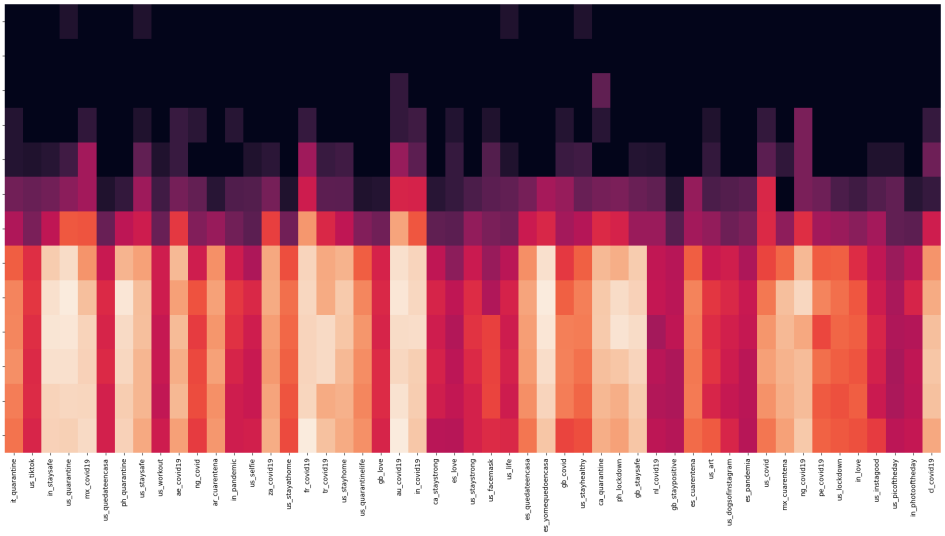
4.5.2 Comparando as séries temporais

A comparação funciona escolhendo uma série temporal, ou seja, um país e uma hashtag específicos. Esta série é então comparada com as séries temporais de todos os outros países para obter uma classificação das mais semelhantes à escolhida inicialmente. Os resultados são exibidos em um *heatmap*, onde a primeira coluna representa a série temporal escolhida como pivô, e as demais colunas mostram as séries temporais mais semelhantes, ordenadas da mais semelhante à menos semelhante.

Comparar séries temporais pode ser visto como comparar vetores. Para isso, utilizamos o algoritmo DTW, que é invariante no tempo e permite uma comparação otimizada. Como exemplo, utilizaremos "Itália" como país, juntamente com a hashtag

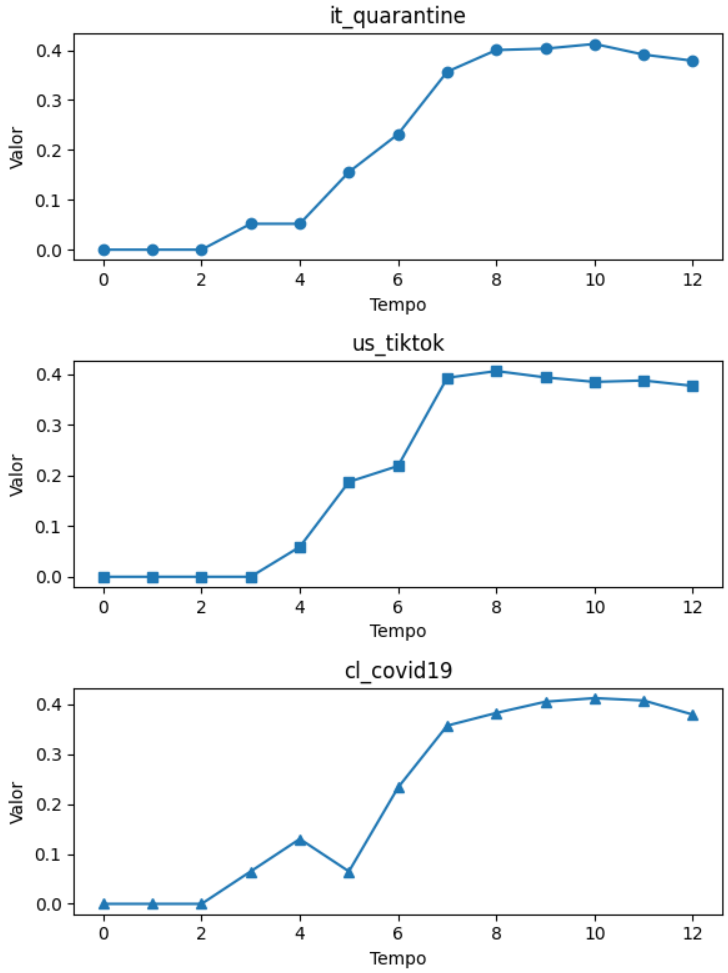
"*quarantine*". As séries temporais dos termos mais similares e do termo menos similar da Figura 18 podem ser vistas na Figura 19, que exibe-as em gráfico de linha.

Figura 18 – *Heatmap* ilustrando as semelhanças com a hashtag "*quarantine*" da Itália. Da mais similar "*us_tiktok*" à menos similar "*cl_covid19*".



Fonte: Autor.

Figura 19 – Séries temporais dos termos mais similares, "*it_quarantine*" e "*us_tiktok*", seguido pelo termo menos similar "*cl_covid19*".



Fonte: Autor.

Observamos que as séries "*it_quarantine*" e "*us_tiktok*" mostram deformações semelhantes, mas ocorrendo em momentos relativamente diferentes. Isso demonstra a característica de invariância temporal do algoritmo DTW, que consegue identificar padrões de similaridade entre as séries, mesmo quando essas variações acontecem em tempos distintos.

A matriz de resultados é um recorte das séries temporais mais semelhantes de cada país. Sua estrutura é semelhante à matriz original: as linhas representam as 13 semanas, numeradas de 0 a 12, e as colunas são nomeadas no padrão "país_hashtag", indicando o país e a hashtag representados pela respectiva série temporal. O Algoritmo 7 é responsável pela comparação e classificação das séries temporais.

Algoritmo 7: Comparação das séries temporais.

```
def time_warping(serie_pivot, country_pivot, tag_pivot, other_series):
    norm_i = np.linalg.norm(serie_pivot)
    n_ti = serie_pivot / norm_i

    rank = []
    for other_country, weeks in other_series.items():
        if other_country == country_pivot:
            continue

        # Convertendo a matriz de "semana X tag" para "tag X semana"
        weeks_t = np.transpose(weeks)
        for other_tag, termJ in enumerate(weeks_t):
            norm_j = np.linalg.norm(termJ)

            # Verificando se o vetor possui dados relevantes
            if norm_j <= 1e-2:
                continue

            n_tj = termJ / norm_j

            score = dtw(n_ti, n_tj)
            rank.append((country_pivot,tag_pivot,other_country,other_tag,score))

    # Retornando as classificacoes ordenadas
    return sorted(rank, key=lambda x: x[-1])
```

Fonte: Autor.

O algoritmo recebe quatro parâmetros: "*serie_pivot*", que é a série temporal escolhida para a comparação; "*country_pivot*", que é o país ao qual a série temporal pertence; "*tag_pivot*", que é a hashtag à qual a série temporal pertence; e "*other_series*", que são as matrizes dos outros países no formato de dicionário, ou seja, podemos percorrê-las com "*other_series.items*()", que permite capturar o nome do país na variável "*other_country*" e sua respectiva matriz na variável "*weeks*".

Começamos normalizando o vetor pivô pela sua norma. Em seguida, percorremos as demais séries temporais uma a uma, armazenando a distância calculada pelo DTW com o vetor pivô. Antes de cada comparação, verificamos se a norma vetorial da série é menor ou igual a "1e-2" (0,01). Isso ocorre porque vetores com uma norma tão baixa possuem poucas informações, ou seja, muitos zeros. Por fim, retornamos as classificações ordenadas

de forma crescente. Como discutido na seção 2.2.2.3, o DTW utiliza a distância euclidiana, portanto, valores mais baixos indicam maior similaridade.

5 RESULTADOS E DISCUSSÃO

A ferramenta recebe quatro parâmetros: a granularidade em dias, o número de hashtags (ou colunas) desejadas na saída, o país e a hashtag usada como pivô para a comparação. A seguir, analisaremos os resultados em duas granularidades distintas: semanal e mensal.

5.1 Granularidade semanal

Na Figura 20, visualizamos as imagens geradas para seis países selecionados de forma arbitrária—Estados Unidos, Itália, França, Espanha, Brasil e Filipinas—com granularidade semanal. O período de cada semana é mostrado na Tabela 5. Devido ao período abrangido pelo conjunto de dados, o número máximo de semanas será de treze, independentemente da granularidade escolhida.

Tabela 5 – Período de cada semana.

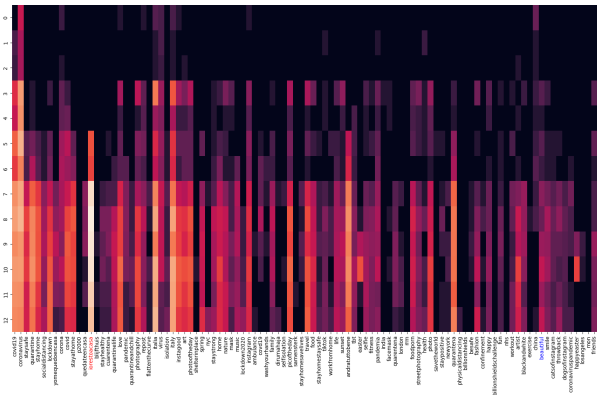
Semana	**Período**
1	2020-02-01 até 2020-02-07
2	2020-02-08 até 2020-02-14
3	2020-02-15 até 2020-02-21
4	2020-02-22 até 2020-02-28
5	2020-02-29 até 2020-03-06
6	2020-03-07 até 2020-03-13
7	2020-03-14 até 2020-03-20
8	2020-03-21 até 2020-03-27
9	2020-03-28 até 2020-04-03
10	2020-04-04 até 2020-04-10
11	2020-04-11 até 2020-04-17
12	2020-04-18 até 2020-04-24
13	2020-04-25 até 2020-05-01

Fonte: Autor.

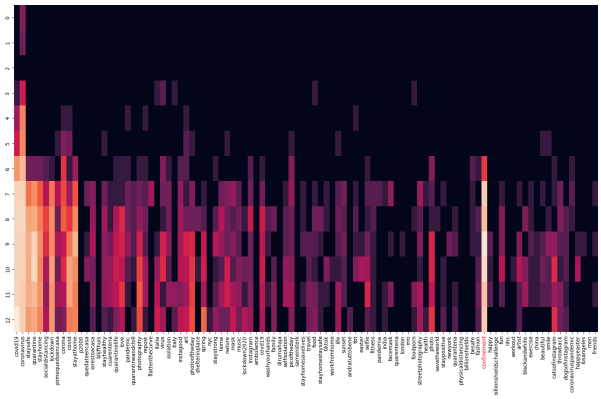
Segundo Mathieu et al. (2020), o número de casos confirmados da doença aumentou significativamente a partir de março de 2020. Isso é evidente nas matrizes da Figura 20, onde valores elevados começam a aparecer a partir da sexta semana (linha 5), enquanto valores mais baixos predominam anteriormente, representados pelas cores escuras do *heatmap*. Na Figura 21, vemos os termos mais semelhantes a hashtag "*iorestoacasa*" da Itália.

Figura 20 – Matriz de cada país, resumindo os quantitativos semanais das hashtags com os valores normalizados utilizando *log10*. Os termos coloridos (vermelho e azul) serão analisados em seguida.

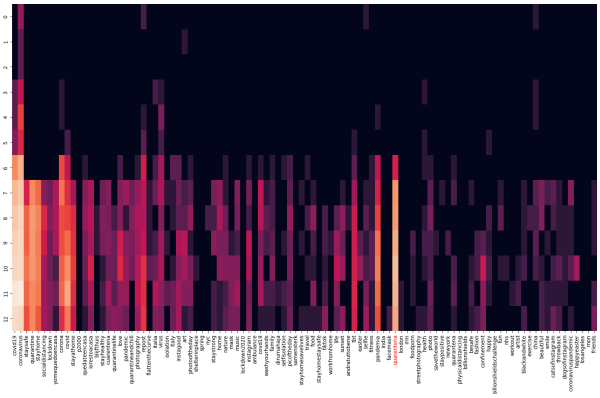
(a) Itália.



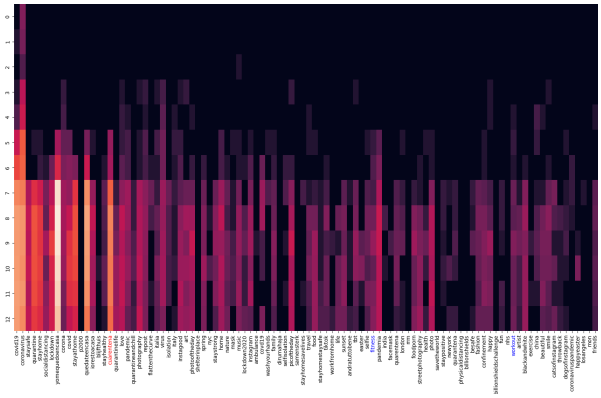
(b) França.



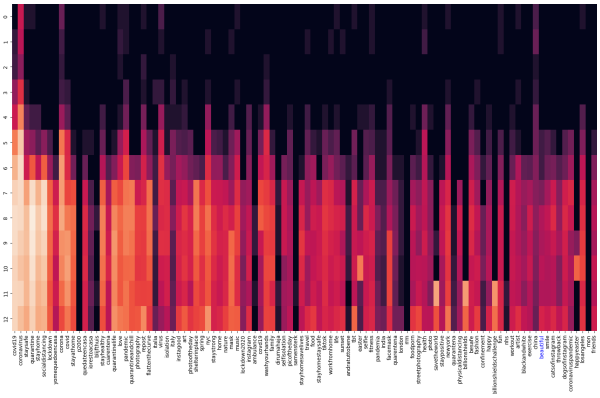
(c) Brasil.



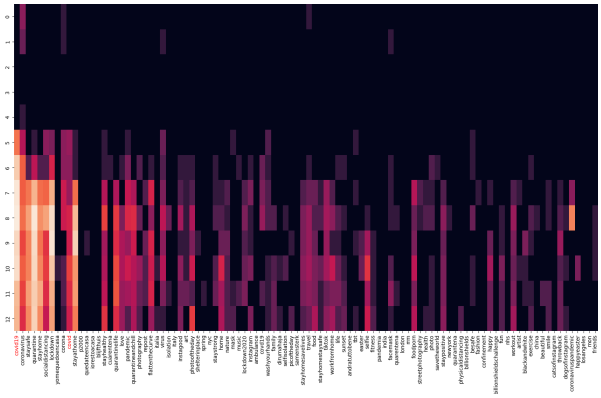
(d) Espanha.



(e) Estados Unidos.

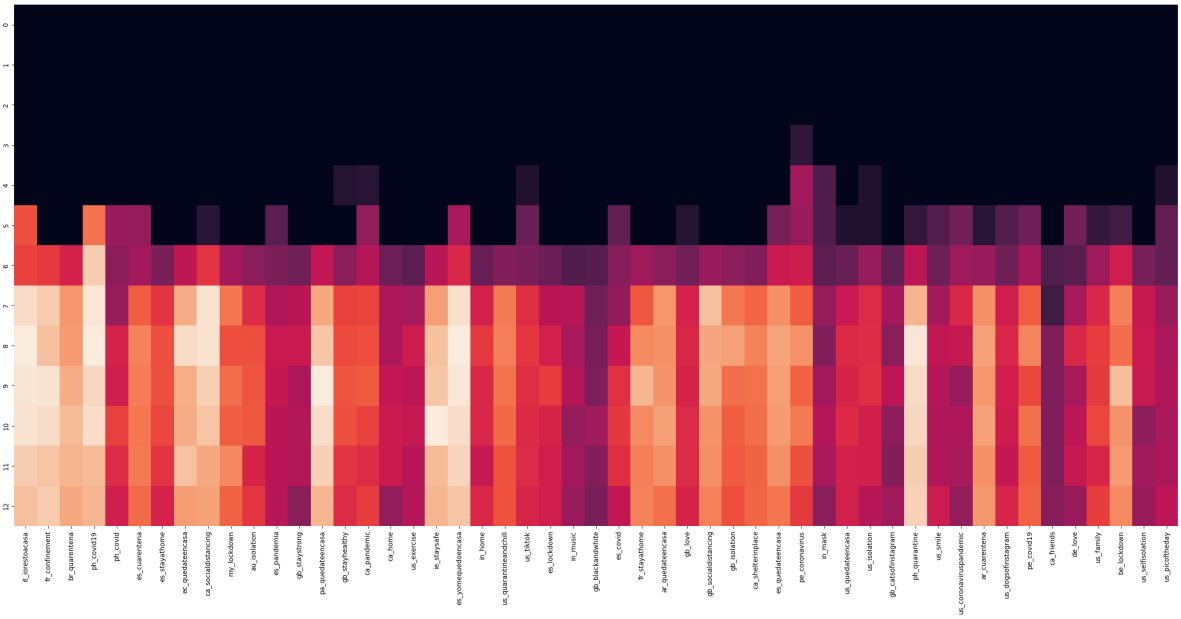


(f) Filipinas.



Fonte: Autor.

Figura 21 – *Heatmap* ilustrando as semelhanças com a hashtag "*iorestoacasa*" da Itália. Da mais similar "*it_iorestoacasa*" à menos similar "*us_picoftheday*".



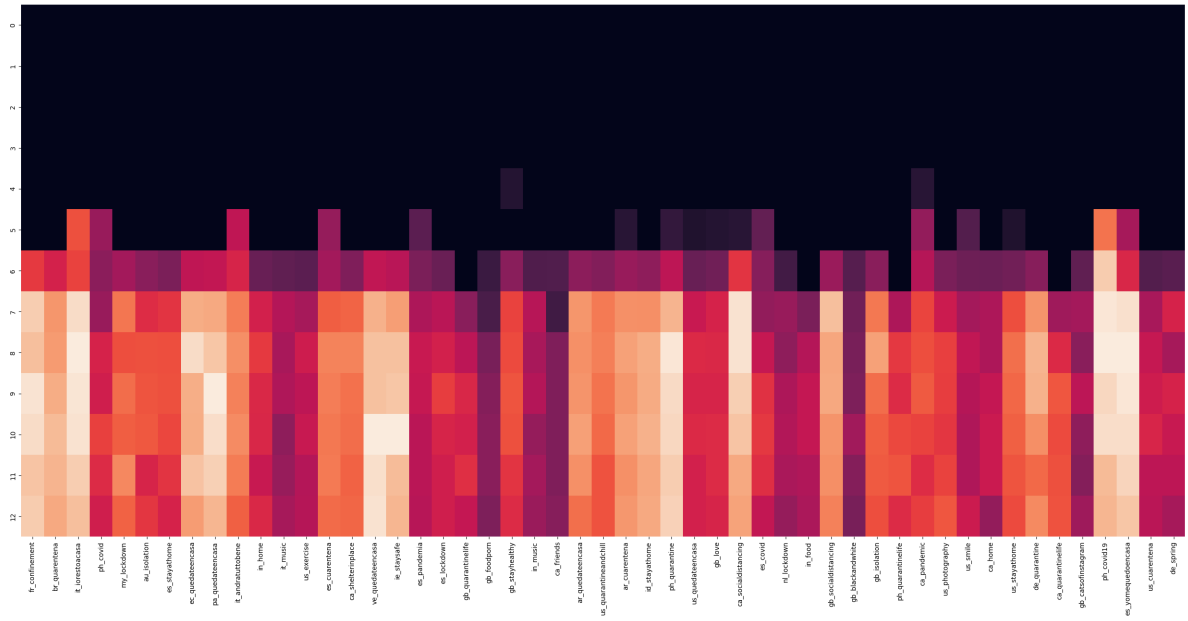
Fonte: Autor.

Na Figura 21, observamos que alguns dos termos mais semelhantes em comportamento são a hashtag "*confinement*" da França, "quarentena" do Brasil, seguidos por "covid19" e "covid" das Filipinas e, depois, "*cuarentena*" da Espanha. Os termos citados foram destacados com a cor vermelha nas matrizes originais, exibidas pela Figura 20.

O isolamento na Itália começou em 9 de março de 2020 (BBCNEWS, 2020). Em seguida, o isolamento começou na França em 17 de março de 2020 (FRANCE24, 2021). Esses eventos podem ter influenciado o comportamento observado na Figura 21, já que correspondem à sexta e sétima semanas a partir do dia 1 de fevereiro de 2020, que é o início do período abrangido pelo conjunto de dados.

Os resultados não satisfazem a propriedade de simetria. Isso significa que o termo equivalente a "*fr_confinement*" não será necessariamente "*it_iorestoacasa*". Na Figura 22, "*br_quarentena*" aparece antes de "*it_iorestoacasa*". Segundo Villela et al. (2021), as medidas de confinamento e distanciamento social foram implementadas no Brasil a partir de 16 de março de 2020, data próxima ao início do isolamento na França, o que pode ter influenciado o comportamento observado.

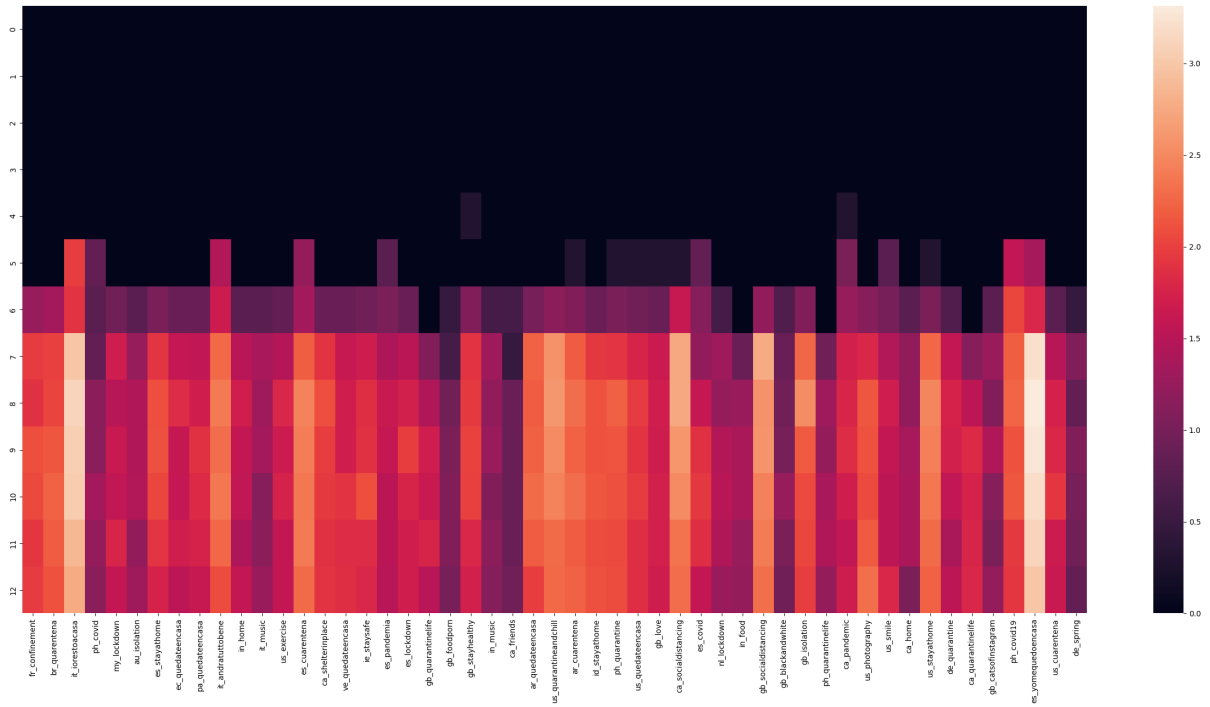
Figura 22 – *Heatmap* ilustrando as semelhanças com a hashtag "*confinement*" da França. Da mais similar "*fr_confinement*" à menos similar "*de_spring*".



Fonte: Autor.

O *heatmap* gerado mantém a escala original do país para cada uma das colunas, o que resulta em um mapeamento de cores individual para cada uma delas. Podemos visualizar a plotagem em escala única através de um quinto parâmetro chamado "*keep_original_scale*", o resultado da Figura 22 em escala única pode ser visto na Figura 23.

Figura 23 – *Heatmap* ilustrando as semelhanças com a hashtag "*confinement*" da França. Da mais similar "*fr_confinement*" à menos similar "*de_spring*", em escala única.



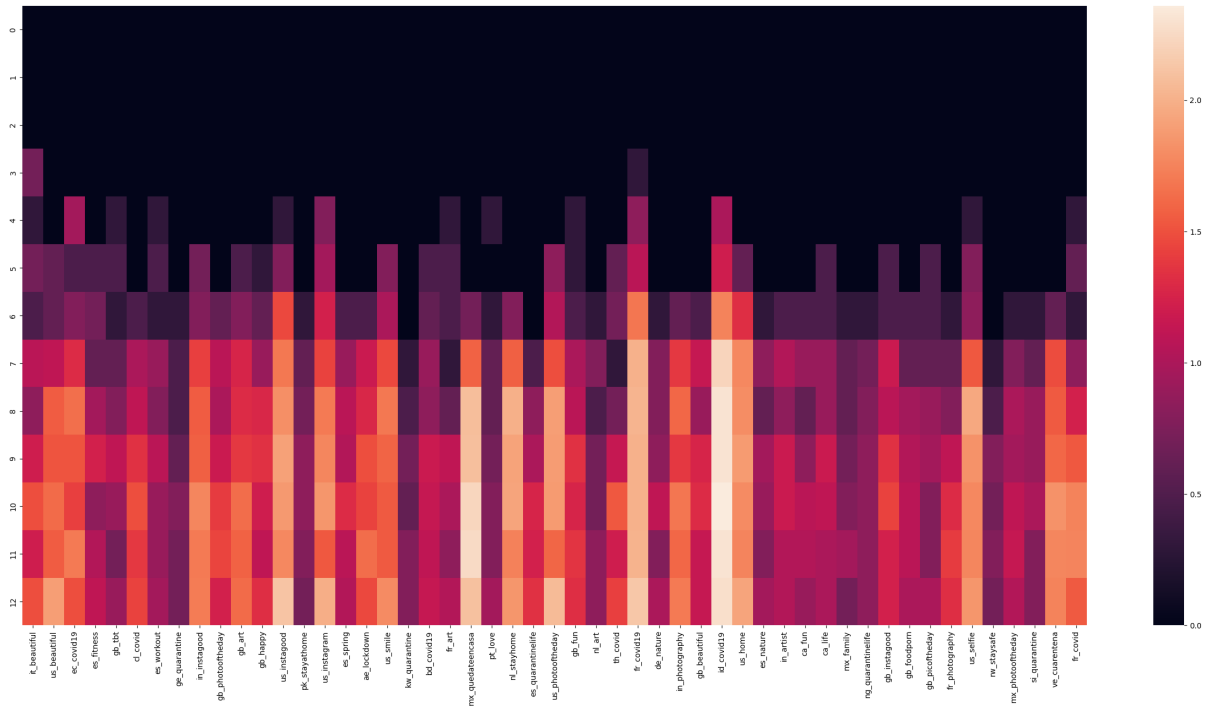
Fonte: Autor.

As distâncias calculadas não são afetadas pelo parâmetro "*keep_original_scale*", tendo em vista que os vetores são normalizados pela sua respectiva norma vetorial, como visto no Algoritmo 7. Portanto, mantemos a mesma ordem do resultado anterior.

A técnica utilizada revela a semelhança no comportamento dos termos, independentemente da língua, permitindo uma análise abrangente da diversidade linguística. Podemos notar isso ao observar termos em idiomas diferentes do inglês, que predomina no conjunto de dados. Essa capacidade de transcender barreiras linguísticas facilita a compreensão das tendências globais e das respostas coletivas aos eventos da pandemia.

Nosso método é invariante ao idioma e pode identificar semelhanças entre termos em diferentes línguas, inclusive aquelas que utilizam alfabetos fora do latino. No entanto, as ferramentas de processamento de texto utilizadas apresentam limitações ao lidar com alfabetos não latinos, que acaba por ignorar esses caracteres. Além disso, o número de publicações com caracteres fora do alfabeto latino é relativamente pequeno no *dataset* e ainda menor após a hidratação. Por esse motivo, recuperamos apenas idiomas cujos caracteres fazem parte do alfabeto latino, como descrito na seção 4.3.

Figura 24 – *Heatmap* ilustrando as semelhanças com a hashtag "*beautiful*" da Itália. Da mais similar "*it_beautiful*" à menos similar "*fr_covid*", em escala única.



Fonte: Autor.

Podemos analisar o comportamento de termos que não fazem parte do idioma local do país. A adoção de termos em inglês em publicações de países de línguas diferentes pode mostrar como a pandemia influenciou a linguagem e introduziu novas expressões no vocabulário cotidiano de várias culturas. Isso exemplifica a interconectividade e a influência mútua das línguas em tempos de crise.

O uso do termo "*beautiful*" pode indicar um esforço para encontrar e compartilhar momentos de beleza e positividade em um período desafiador. Isso pode refletir uma tendência global de busca por esperança e bem-estar emocional durante a pandemia, com pessoas se concentrando em aspectos positivos da vida para contrabalançar o estresse e a ansiedade.

Além disso, a associação entre "*beautiful*" e termos como "*fitness*" e "*workout*" sugere que houve uma ênfase significativa em manter a saúde física e mental. Durante a pandemia, muitas pessoas podem ter voltado sua atenção para exercícios e cuidados pessoais, vendo isso como uma maneira de manter-se saudáveis e mentalmente equilibradas. Os termos citados encontram-se destacados com a cor azul na Figura 20.

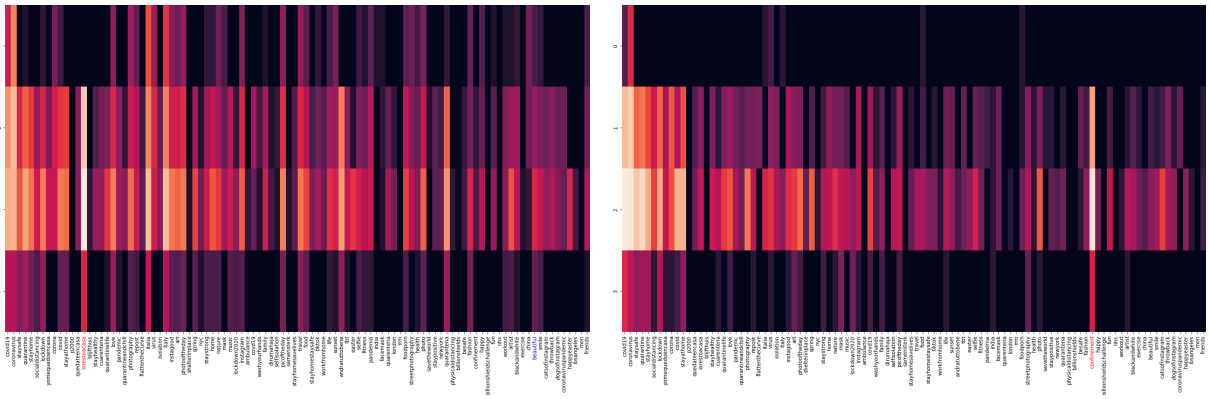
5.2 Granularidade mensal

Utilizar a granularidade mensal resulta em matrizes de menor escala, considerando que temos apenas quatro meses de dados. Na Figura 25, apresentamos os seis países vistos no início da seção anterior, em granularidade mensal (ou seja, "*interval_days*=30").

Figura 25 – Matriz de cada país, resumindo os quantitativos mensais das hashtags com os valores normalizados utilizando *log10*. Os termos coloridos (vermelho e azul) serão analisados em seguida.

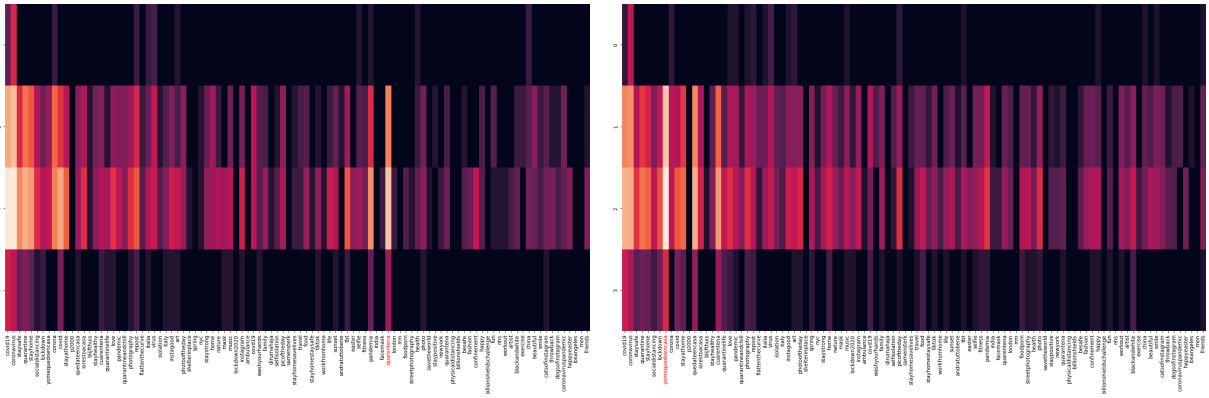
(a) Itália.

(b) França.



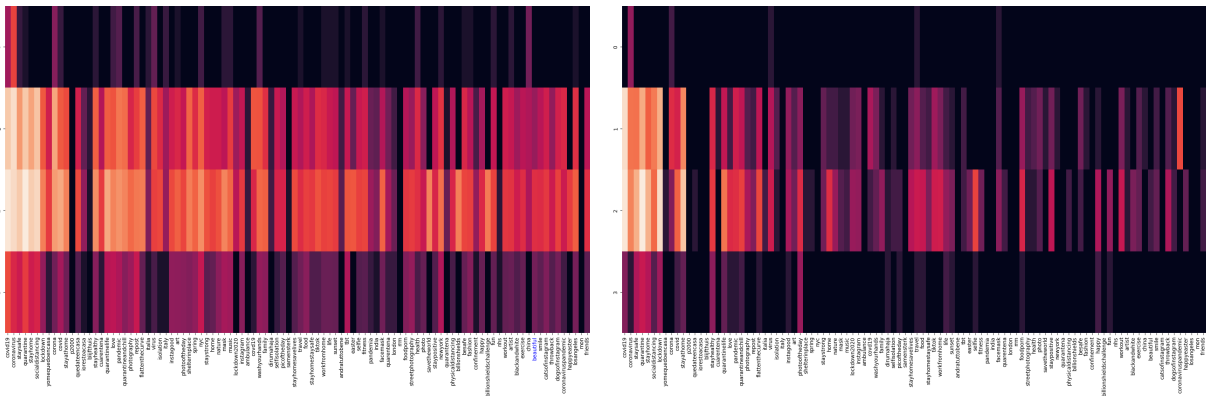
(c) Brasil.

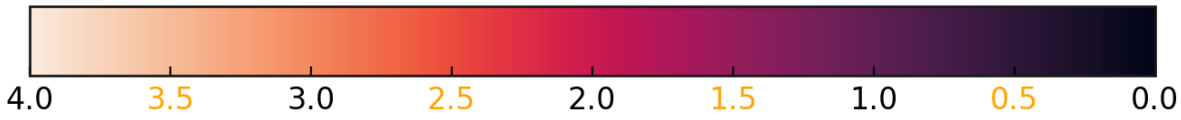
(d) Espanha.



(e) Estados Unidos.

(f) Filipinas.





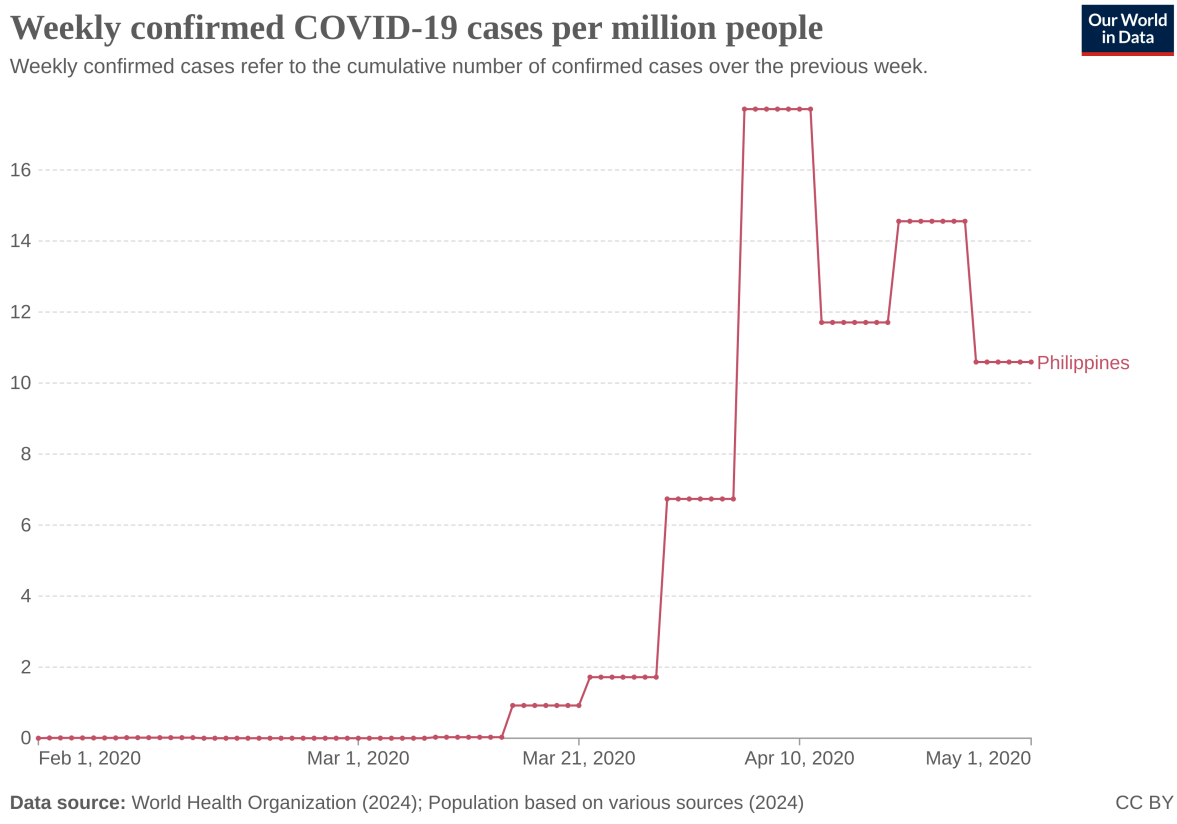
Fonte: Autor.

As colunas permanecem organizadas na mesma ordem, independente da granularidade. É possível observar um padrão consistente ao comparar as granularidades semanal e mensal, visto que os maiores valores são mais frequentes no segundo e terceiro mês (linhas

1 e 2), uma tendência que reflete o comportamento observado na granularidade semanal, onde a contagem de casos começa a aumentar significativamente a partir da sexta semana.

Na Figura 26, observamos a evolução de casos semanais de COVID-19 das Filipinas. Ao visualizar a matriz das Filipinas, é possível perceber uma consistência em relação ao gráfico de casos confirmados de COVID-19. No início, em fevereiro de 2020, o número de casos era baixo, o que é refletido pela baixa intensidade na matriz. Em março e abril, o número de casos começou a aumentar significativamente, coincidindo com uma maior atividade e intensidade de termos relacionados à pandemia na visualização. Já em 1º de maio, a situação começa a se estabilizar, o que também se reflete na matriz, com uma redução na evolução dos termos.

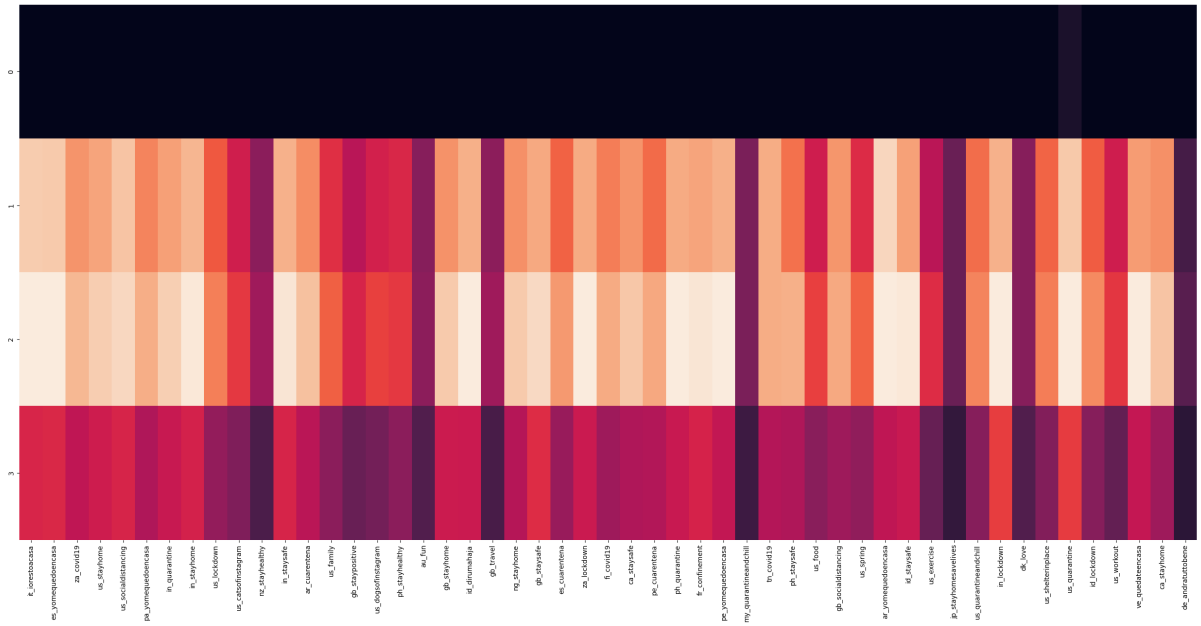
Figura 26 – Casos semanais confirmados de COVID-19 por milhão de habitantes nas Filipinas, dentro do período abrangido pelo conjunto de dados (1º de fevereiro de 2020 até 1º de maio de 2020).



Fonte: Mathieu et al. (2020).

Apresentamos a seguir as matrizes de comparação de séries temporais em granularidade mensal, iniciando pela plotagem da Itália com a hashtag "*iorestoacasa*".

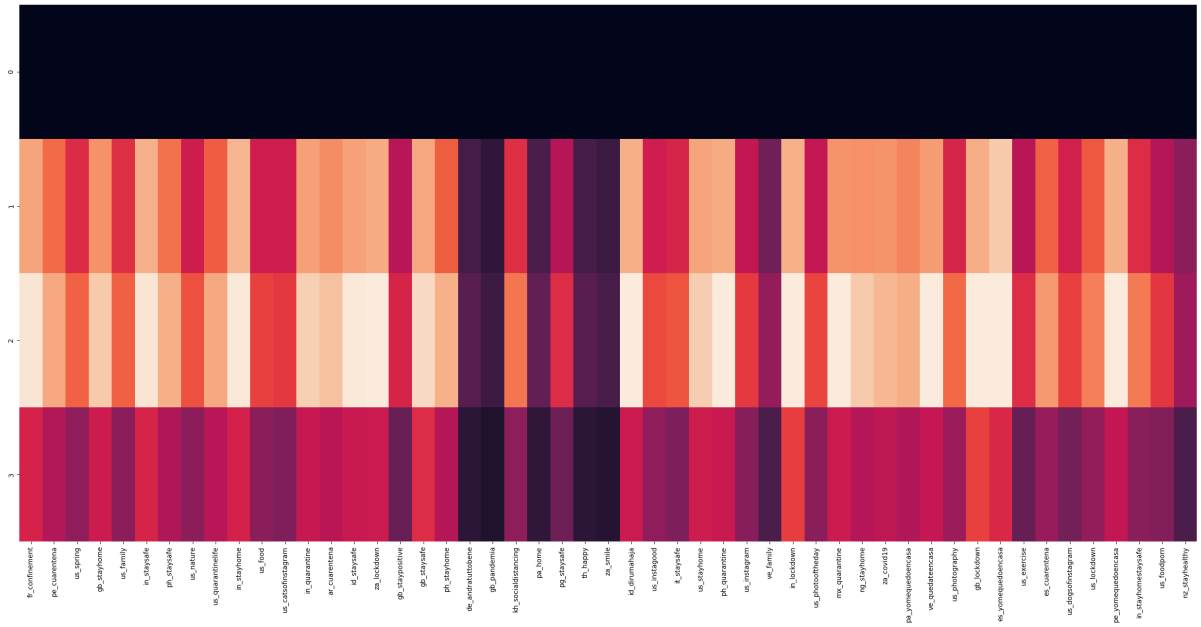
Figura 27 – *Heatmap* ilustrando as semelhanças com a hashtag "*iorestoacasa*" da Itália. Da mais similar "*it_iorestoacasa*" à menos similar "*de_andratuttobene*".



Fonte: Autor.

Podemos identificar uma relação interessante entre os termos "*iorestoacasa*", da Itália, e "*yomequedoencasa*", da Espanha. Essa correlação não foi observada na análise semanal, onde o termo "*confinement*", da França, aparecia em vez de "*yomequedoencasa*". Essa mudança de comportamento pode ser explicada pelo maior nível de agregação na análise mensal, que captura uma visão mais ampla dos dados, possibilitando a identificação de conexões que emergem ao longo de um período mais estendido. Os termos citados foram destacados com a cor vermelha nas matrizes originais.

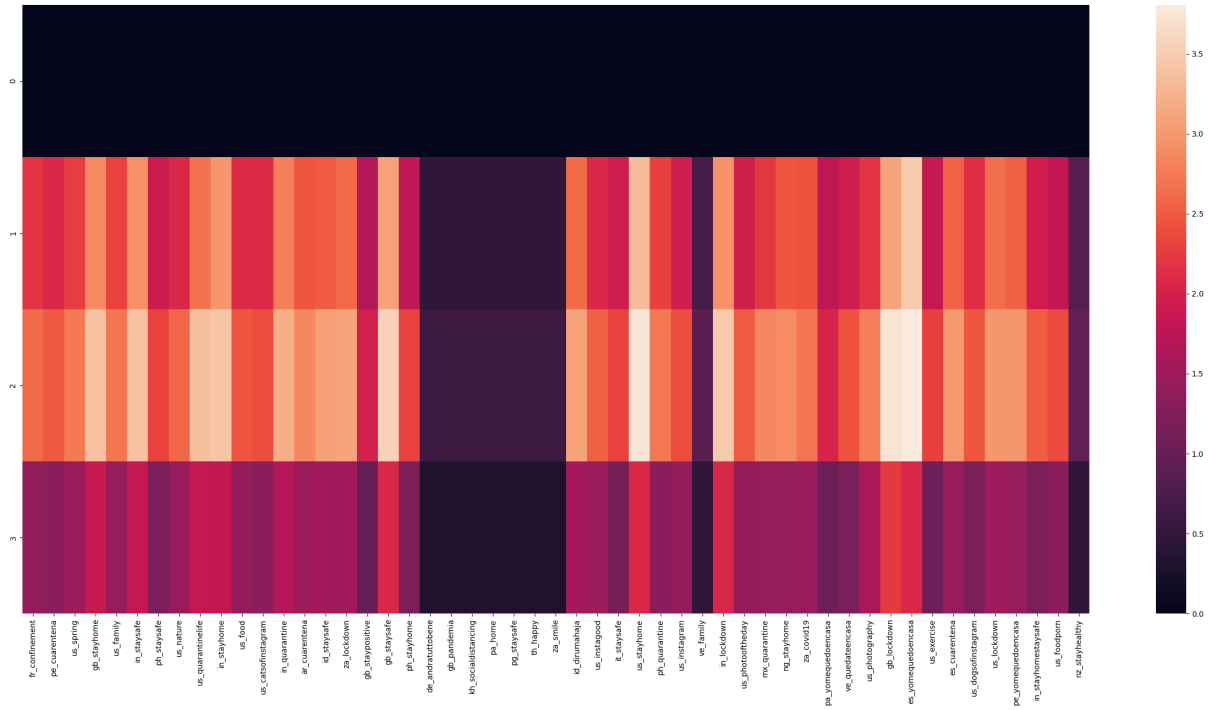
Figura 28 – *Heatmap* ilustrando as semelhanças com a hashtag "*confinement*" da França. Da mais similar "*fr_confinement*" à menos similar "*nz_stayhealthy*".



Fonte: Autor.

A hashtag "*confinement*" da França apareceu relacionada com "*cuarentena*" do Peru, uma conexão que não foi observada na análise semanal, onde a relação predominante era com "quarentena" do Brasil.

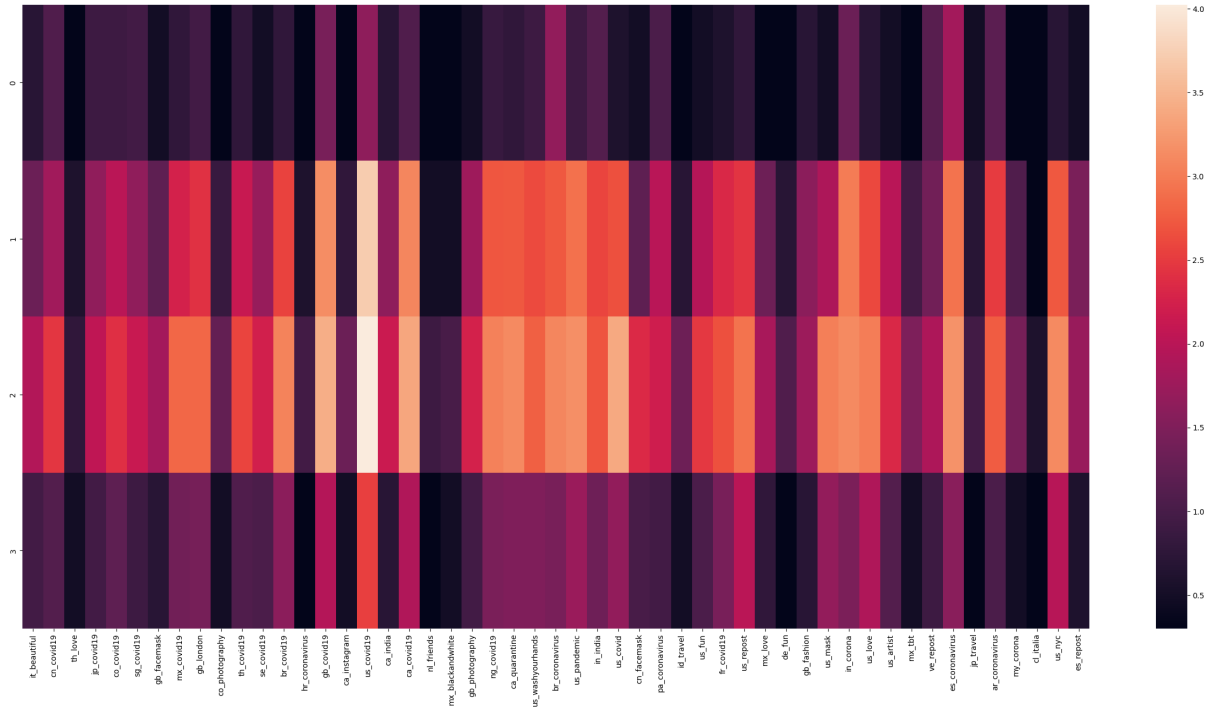
Figura 29 – *Heatmap* ilustrando as semelhanças com a hashtag "*confinement*" da França. Da mais similar "*fr_confinement*" à menos similar "*nz_stayhealthy*", em escala única.



Fonte: Autor.

Na plotagem mensal, o aumento do valor máximo da escala do *heatmap*, passando de 3.0 na análise semanal para 3.5, reflete a diferença na agregação dos dados e no comportamento dos termos ao longo de um período mais longo. Esse aumento indica que, ao se considerar uma granularidade mensal, houve um maior número de ocorrências de termos.

Figura 30 – *Heatmap* ilustrando as semelhanças com a hashtag "*beautiful*" da Itália. Da mais similar "*it_beautiful*" à menos similar "*es_repost*", em escala única.



Fonte: Autor.

Nessa análise, visualizamos uma nova relação entre o termo "*beautiful*" da Itália e "covid19" da China, que na análise semanal era substituída por "*beautiful*" dos EUA. Além dessa mudança de correlação entre os termos, o aumento da escala do *heatmap* de 2.0 na análise semanal para 4.0 na análise mensal mostra que os dados são, de fato, mais agregados, capturando picos de atividade acumulados ao longo de várias semanas. Os termos analisados foram destacados nas matrizes originais com a cor azul.

A análise da semelhança de comportamento dos termos de forma mensal pode resultar em menos informações e menos semelhanças entre termos diferentes do que a análise semanal. Primeiramente, a granularidade dos dados é diferente: análises semanais fornecem uma resolução mais fina, capturando variações e tendências de curto prazo, revelando picos específicos de interesse ou atividade em resposta a eventos imediatos, como novas políticas de isolamento ou campanhas de mídia social.

Em contrapartida, a agregação mensal pode suavizar essas variações de curto prazo, obscurecendo picos e vales que são visíveis nas análises semanais, levando a uma visão mais homogênea dos dados, onde as nuances e flutuações são menos evidentes. Além disso, o tamanho do conjunto de dados, que abrange apenas quatro meses, amplifica essas limitações. Com um período de tempo limitado, a agregação mensal pode esconder variações significativas e reduzir a capacidade de identificar correlações e tendências significativas, ao contrário da análise semanal, que proporciona uma visão mais detalhada e rica em

informações.

6 CONSIDERAÇÕES FINAIS

Durante o desenvolvimento deste trabalho, foram identificadas diversas dificuldades e limitações, principalmente relacionadas ao recorte nos dados devido à informação de geolocalização. A escolha de publicações com coordenadas geográficas exatas, embora permita uma análise mais confiável baseada em localização, reduziu significativamente a quantidade de dados recuperados. Apenas 1% dos *tweets* coletados continham coordenadas GPS, limitando a representatividade do conjunto de dados e, consequentemente, a abrangência das análises.

Com dados coletados por um período maior, do início ao fim do surto, seria esperado que outros padrões de crescimento de casos emergissem. A análise temporal mais longa poderia revelar variações no comportamento das discussões e nas tendências relacionadas aos casos da pandemia, permitindo uma compreensão mais completa de sua evolução.

A escolha da granularidade influencia diretamente os resultados e as semelhanças entre os termos, pois define o nível de detalhamento da análise ao longo do tempo. Na granularidade semanal, capturamos padrões mais imediatos e variações de curto prazo, revelando reações rápidas a eventos específicos. Já a granularidade mensal suaviza essas variações de curto prazo, destacando tendências mais duradouras e conexões que se consolidam ao longo do tempo. Como resultado, termos que se correlacionam fortemente em um curto período podem não manter essa relação na análise mensal, enquanto novas semelhanças mais consistentes surgem. Assim, a granularidade escolhida determina quais padrões serão evidenciados e a intensidade das correlações.

Como trabalhos futuros, podemos combinar dados do Twitter com informações de outras plataformas de mídia social, o que pode proporcionar uma visão mais holística das discussões e tendências globais. Também é possível desenvolver *dashboards* interativos que permitam a exploração dinâmica dos dados e resultados, facilitando a compreensão e a comunicação dos insights obtidos.

Em resumo, este trabalho contribuiu para a compreensão das dinâmicas de comunicação durante a pandemia de COVID-19, mas também destacou a necessidade de abordagens mais abrangentes e sofisticadas para lidar com os desafios apresentados pelo uso de dados de mídias sociais. As melhorias propostas e as direções sugeridas para trabalhos futuros oferecem um caminho promissor para aprofundar e expandir as análises realizadas.

Referências

ACHARYA, A. S. et al. Sampling: Why and how of it. *Indian Journal of Medical Specialties*, v. 4, n. 2, p. 330–333, 2013. Citado na página 6.

AHMED, A. et al. Visual analysis of history of world cup: A dynamic network with dynamic hierarchy and geographic clustering. In: SPRINGER. *Visual Information Communication*. [S.l.], 2010. p. 25–39. Citado na página 18.

ALHOSAINI, H. et al. Api recommendation for mashup creation: A comprehensive survey. *The Computer Journal*, Oxford University Press, v. 67, n. 5, p. 1920–1940, 2024. Citado na página 13.

ARCHAMBAULT, D.; PURCHASE, H.; PINAUD, B. Animation, small multiples, and the effect of mental map preservation in dynamic graphs. *IEEE Transactions on Visualization and Computer Graphics*, IEEE, v. 17, n. 4, p. 539–552, 2010. Citado 3 vezes nas páginas 19, 20 e 21.

BBCNEWS. *Coronavirus: Worst Economic Crisis Since 1930s Depression, IMF Says*. 2020. Acesso em: 31 jul. 2024. Disponível em: <https://www.bbc.com/news/world-europe-51810673>. Citado na página 51.

BECK, F. et al. A taxonomy and survey of dynamic graph visualization. In: WILEY ONLINE LIBRARY. *Computer Graphics Forum*. [S.l.], 2017. v. 36, n. 1, p. 133–159. Citado na página 17.

BHATTARAI, B. P. et al. Big data analytics in smart grids: State-of-the-art, challenges, opportunities, and future directions. *IET Smart Grid*, Wiley Online Library, v. 2, n. 2, p. 141–154, 2019. Citado na página 1.

BIKAKIS, N. Big data visualization tools. *CoRR*, abs/1801.08336, 2018. Disponível em: <http://arxiv.org/abs/1801.08336>. Citado na página 5.

BUKOVINA, J. Social media big data and capital markets—an overview. *Journal of Behavioral and Experimental Finance*, v. 11, p. 18–26, 2016. ISSN 2214-6350. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2214635016300272>. Citado na página 23.

CLEVELAND, W. *Visualizing Data*. AT&T Bell Laboratories, 1993. ISBN 9780963488404. Disponível em: <https://books.google.com.br/books?id=V-dQAAAAMAAJ>. Citado na página 6.

CLIFFORD, H. T.; STEPHENSON, W. Book. *An Introduction to Numerical Classification / H. T. Clifford and W. Stephenson*. [S.l.]: Academic Press New York, 1975. xii, 229 p. : p. ISBN 0121767507. Citado na página 8.

CUKIER, K. Data, data everywhere. *Economist*, v. 394, n. 8671, p. 3–5, 2010. Citado na página 4.

DIEHL, S.; GÖRG, C. Graphs, they are changing. In: SPRINGER. *International Symposium on Graph Drawing*. [S.l.], 2002. p. 23–30. Citado na página 20.

EADES, P. A heuristic for graph drawing. v. 42, p. 149–160, 1983. Citado na página 18.

ERTEN, C. et al. Graphael: Graph animations with evolving layouts. In: SPRINGER. *International Symposium on Graph Drawing*. [S.l.], 2003. p. 98–110. Citado na página 20.

FRANCE24. *In Pictures: A Look Back, One Year After France Went Into Lockdown*. 2021. Acesso em: 31 jul. 2024. Disponível em: <https://www.france24.com/en/france/20210317-in-pictures-a-look-back-one-year-after-france-went-into-lockdown>. Citado na página 51.

FRISHMAN, Y.; TAL, A. Online dynamic graph drawing. *IEEE Transactions on Visualization and Computer Graphics*, IEEE, v. 14, n. 4, p. 727–740, 2008. Citado na página 18.

FRUCHTERMAN, T. M. J.; REINGOLD, E. M. Graph drawing by force-directed placement. *Software: Practice and Experience*, v. 21, n. 11, p. 1129–1164, 1991. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/spe.4380211102>. Citado 2 vezes nas páginas 18 e 19.

FURHT, B.; VILLANUSTRE, F. Introduction to big data. In: _____. *Big Data Technologies and Applications*. Cham: Springer International Publishing, 2016. p. 3–11. ISBN 978-3-319-44550-2. Disponível em: <https://doi.org/10.1007/978-3-319-44550-2_1>. Citado 4 vezes nas páginas 1, 2, 4 e 5.

GALDINO, N. Big data: Ferramentas e aplicabilidade. *XXII SEGeT. Simpósio de Excelência em Gestão e Tecnologia. Associação Educacional Dom Bosco. Resende. Rio de Janeiro*, 2015. Citado na página 4.

GALEANO, P.; PEÑA, D. Data science, big data, and statistics. v. 28, n. 2, p. 289–329, Jun 2019. ISSN 1863-8260. Disponível em: <https://doi.org/10.1007/s11749-019-00651-9>. Citado na página 1.

GANDOMI, A.; HAIDER, M. Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, v. 35, n. 2, p. 137–144, 2015. ISSN 0268-4012. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0268401214001066>. Citado na página 4.

GÖRG, C. et al. Dynamic graph drawing of sequences of orthogonal and hierarchical graphs. In: SPRINGER. *International Symposium on Graph Drawing*. [S.l.], 2005. p. 228–238. Citado 2 vezes nas páginas 18 e 21.

GOROCHOWSKI, T. E.; BERNARDO, M. di; GRIERSON, C. S. Using aging to visually uncover evolutionary processes on networks. *IEEE Transactions on Visualization and Computer Graphics*, v. 18, n. 8, p. 1343–1352, 2012. Citado na página 19.

HAN, J.; PEI, J.; TONG, H. *Data Mining: Concepts and Techniques*. [S.l.]: Morgan kaufmann, 2022. Citado 2 vezes nas páginas 7 e 8.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. The elements of statistical learning: Data mining, inference, and prediction. *Springer Series in Statistics*, Springer, 2009. Citado na página 15.

HUNTER, J. D. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, IEEE COMPUTER SOC, v. 9, n. 3, p. 90–95, 2007. Citado na página 31.

JAMES, G. et al. *Introduction to Statistical Learning*. [S.l.]: Springer, 2013. Citado na página 14.

KAPLAN, A. M.; HAENLEIN, M. Users of the world, unite! the challenges and opportunities of social media. *Business Horizons*, v. 53, n. 1, p. 59–68, 2010. ISSN 0007-6813. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0007681309001232>. Citado na página 23.

KEIM, D. A. et al. Visual analytics: Scope and challenges. In: _____. *Visual Data Mining: Theory, Techniques and Tools for Visual Analytics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008. p. 76–90. ISBN 978-3-540-71080-6. Disponível em: <https://doi.org/10.1007/978-3-540-71080-6_6>. Citado na página 6.

KEOGH, E.; RATANAMAHATANA, C. A. Exact indexing of dynamic time warping. *Knowledge and Information Systems*, Springer, v. 7, p. 358–386, 2005. Citado 2 vezes nas páginas 8 e 9.

KHAYYAT, M. et al. Time series facebook prophet model and python for covid-19 outbreak prediction. *CMES - Computer Modeling in Engineering and Sciences*, v. 67, n. 3, p. 3781–3793, 2021. ISSN 1526-1492. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1526149220002064>. Citado na página 24.

KITCHIN, R.; MCARDLE, G. What makes big data, big data? exploring the ontological characteristics of 26 datasets. *Big Data & Society*, v. 3, n. 1, p. 2053951716631130, 2016. Disponível em: <https://doi.org/10.1177/2053951716631130>. Citado na página 4.

KOBOUROV, S. Force-directed algorithms. In: _____. [S.l.: s.n.], 2013. p. 383–408. Citado na página 18.

LAMSAL, R. Design and analysis of a large-scale covid-19 tweets dataset. *Applied Intelligence*, Springer, v. 51, p. 2790–2804, 2021. Citado 2 vezes nas páginas 24 e 33.

MATHIEU, E. et al. Coronavirus pandemic (covid-19). *Our World in Data*, 2020. Disponível em: <https://ourworldindata.org/coronavirus>. Citado 2 vezes nas páginas 49 e 57.

MODI, A. et al. Sentiment analysis of twitter feeds using flask environment: A superior application of data analysis. *Annals of Data Science*, Springer, v. 11, n. 1, p. 159–180, 2024. Citado na página 13.

NEWMAN, M. E. Fast algorithm for detecting community structure in networks. *Physical Review E*, APS, v. 69, n. 6, p. 066133, 2004. Citado na página 22.

PERER, A.; SUN, J. Matrixflow: Temporal network visual analytics to track symptom evolution during disease progression. In: AMERICAN MEDICAL INFORMATICS ASSOCIATION. *AMIA Annual Symposium Proceedings*. [S.l.], 2012. v. 2012, p. 716. Citado 3 vezes nas páginas 18, 21 e 22.

PEZOA, F. et al. Foundations of json schema. In: INTERNATIONAL WORLD WIDE WEB CONFERENCES STEERING COMMITTEE. *Proceedings of the 25th International Conference on World Wide Web*. [S.l.], 2016. p. 263–273. Citado na página 14.

PROBST, D.; REYMOND, J.-L. Visualization of very large high-dimensional data sets as minimum spanning trees. *Journal of Cheminformatics*, v. 12, n. 1, p. 12, Feb 2020. ISSN 1758-2946. Disponível em: <https://doi.org/10.1186/s13321-020-0416-x>. Citado na página 1.

PURCHASE, H. C.; GÖRG, C.; HOGGAN, E. How important is the "mental map"? - an empirical investigation of a dynamic graph layout algorithm. In: SPRINGER. *International Symposium on Graph Drawing*. [S.l.], 2006. p. 184–195. Citado na página 21.

QAZI, U.; IMRAN, M.; OFLI, F. Geocov19: A dataset of hundreds of millions of multilingual covid-19 tweets with location information. *SIGSPATIAL Special*, Association for Computing Machinery, New York, NY, USA, v. 12, n. 1, p. 6–15, jul 2020. Disponível em: <https://doi.org/10.1145/3404820.3404823>. Citado 6 vezes nas páginas 2, 24, 25, 27, 28 e 29.

RASCHKA, S. *Python Machine Learning*. [S.l.]: Packt Publishing LTD, 2015. Citado na página 30.

ROESSLEIN, J. Tweepy documentation. *Online] http://tweepy. readthedocs. io/en/v3*, v. 5, 2009. Citado na página 33.

SADIKU, M. et al. Data visualization. *International Journal of Engineering Research and Advanced Technology (IJERAT)*, v. 2, n. 12, p. 11–16, 2016. Citado na página 6.

SAKOE, H.; CHIBA, S. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, IEEE, v. 26, n. 1, p. 43–49, 1978. Citado na página 10.

SALVADOR, S.; CHAN, P. Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis*, IOS Press, v. 11, n. 5, p. 561–580, 2007. Citado 3 vezes nas páginas 2, 9 e 10.

SANTOS, C. Estatística descritiva. *Manual de Autoaprendizagem*, v. 2, 2007. Citado na página 5.

SCHMARZO, B. *Big Data: Understanding How Data Powers Big Business*. [S.l.]: John Wiley & Sons, 2013. Citado na página 5.

SEBEI, H.; TAIEB, M. A. H.; AOUICHA, M. B. Review of social media analytics process and big data pipeline. *Social Network Analysis and Mining*, v. 8, n. 1, p. 30, Apr 2018. ISSN 1869-5469. Disponível em: <https://doi.org/10.1007/s13278-018-0507-0>. Citado na página 23.

SINGHAL, A. Modern information retrieval: A brief overview. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, v. 24, n. 4, p. 35–43, 2001. Citado na página 7.

STIEGLITZ, S. et al. Social media analytics – challenges in topic discovery, data collection, and data preparation. *International Journal of Information Management*, v. 39, p. 156–168, 2018. ISSN 0268-4012. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0268401217308526>. Citado 3 vezes nas páginas 1, 2 e 23.

STRANG, G. *Linear Algebra and Its Applications*. 5th. ed. [S.l.]: Cengage Learning, 2016. Citado na página 15.

TAN, P. N.; STEINBACH, M.; KUMAR, V. *Introdução à Mineração de Dados*. [S.l.: s.n.], 2005. 500 p. Citado na página 7.

TAVENARD, R. et al. Tslearn, a machine learning toolkit for time series data. *Journal of Machine Learning Research*, v. 21, n. 118, p. 1–6, 2020. Disponível em: <http://jmlr.org/papers/v21/20-091.html>. Citado na página 32.

TUFTE, E. R. *The Visual Display of Quantitative Information*. 2nd. ed. Cheshire, CT: Graphics Press, 2001. Citado na página 12.

VILLELA, E. Faria de M. et al. Covid-19 outbreak in brazil: Adherence to national preventive measures and impact on people's lives, an online survey. *BMC Public Health*, v. 21, n. 1, p. 152, Jan 2021. ISSN 1471-2458. Disponível em: <https://doi.org/10.1186/s12889-021-10222-z>. Citado na página 51.

WANG, R. Y.; STRONG, D. M. Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, Routledge, v. 12, n. 4, p. 5–33, 1996. Disponível em: <https://doi.org/10.1080/07421222.1996.11518099>. Citado na página 5.

WASKOM, M. L. Seaborn: Statistical data visualization. *Journal of Open Source Software*, The Open Journal, v. 6, n. 60, p. 3021, 2021. Disponível em: <https://doi.org/10.21105/joss.03021>. Citado na página 31.

YOUNAS, M. Research challenges of big data. *Service Oriented Computing and Applications*, v. 13, n. 2, p. 105–107, Jun 2019. ISSN 1863-2394. Disponível em: <https://doi.org/10.1007/s11761-019-00265-x>. Citado na página 4.