UNIVERSIDADE FEDERAL DE OURO PRETO
INSTITUTO DE CIÊNCIAS EXATAS E BIOLÓGICAS
DEPARTAMENTO DE COMPUTAÇÃO

JHONATAN GOMES DE SOUZA

EXPLORANDO A REDE CIENTÍFICA DE COMPUTAÇÃO NO BRASIL: ABORDAGEM INTEGRADA COM BERT, ANÁLISE DE AGRUPAMENTO E OPENALEX

Ouro Preto, MG
2025

JHONATAN GOMES DE SOUZA

EXPLORANDO A REDE CIENTÍFICA DE COMPUTAÇÃO NO BRASIL: ABORDAGEM INTEGRADA COM BERT, ANÁLISE DE AGRUPAMENTO E OPENALEX

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal de Ouro Preto como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Eduardo José da Silva Luz
Coorientador: Vander Luis de Souza Freitas

Ouro Preto, MG
2025

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
INSTITUTO DE CIENCIAS EXATAS E BIOLOGICAS
DEPARTAMENTO DE COMPUTACAO

FOLHA DE APROVAÇÃO

Jhonatan Gomes de Souza

Explorando a Rede Científica de Computação no Brasil: Abordagem Integrada com BERT, Análise de Agrupamento e OpenAlex

Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal
de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Ciência da Computação

Aprovada em 03 de Setembro de 2025.

Membros da banca

Eduardo José da Silva Luz (Orientador) - Doutor - Universidade Federal de Ouro Preto
Vander Luis de Souza Freitas (Coorientador) - Doutor - Universidade Federal de Ouro Preto
Valéria de Carvalho Santos (Examinadora) - Doutora - Universidade Federal de Ouro Preto
AUGUSTO FERREIRA GUILARDUCCI (Examinador) - Msc - Programa de Pós-Graduação em Ciência da Computação - UFOP

Eduardo José da Silva Luz, Orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 03/09/2025.

Documento assinado eletronicamente por **Eduardo Jose da Silva Luz**, **PROFESSOR DE MAGISTERIO SUPERIOR**, em 03/09/2025, às 19:46, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0 , informando o código verificador **0965554** e o código CRC **DB53B1C0**.

R. Diogo de Vasconcelos, 122, - Bairro Pilar Ouro Preto/MG, CEP 35402-163
Telefone: 3135591692 - www.ufop.br

AGRADECIMENTOS

Agradeço, primeiramente, a Deus, Jesus e ao Espírito Santo, pois sem Ele nada disso seria possível. À minha família por todo apoio e suporte e aos meus amigos e colegas pelo companheirismo ao longo desta jornada. Ao meu orientador, Eduardo Luz e ao meu coorientador Vander Freitas pela orientação e todo valioso conhecimento compartilhados comigo. Por fim, agradeço a UFOP e a todos os professores e profissionais que contribuíram para a minha formação.

Resumo

O campo da Ciência da Ciência (*Science of Science*) tem ganhado relevância ao analisar dados de pesquisa para compreender os fatores que impulsionam o sucesso científico. Esta monografia aplica técnicas de Processamento de Linguagem Natural e aprendizado não supervisionado para mapear a estrutura temática e os perfis de impacto da comunidade de pesquisadores vinculados a Programas de Pós-Graduação em Computação no Brasil. Utilizando dados de 1.511 pesquisadores, extraídos da OpenAlex, Plataforma Lattes e do relatório quadrienal da CAPES. Foram gerados *embeddings* para cada pesquisador com o modelo SciBERT a partir de seus tópicos de pesquisa. A aplicação do algoritmo de clusterização HDBSCAN sobre esses *embeddings* revelou a existência de 6 comunidades temáticas distintas, incluindo um grande *cluster* central focado nos fundamentos da computação e grupos interdisciplinares na interface com ciências da vida, física e humanidades. A análise quantitativa demonstrou que, embora o núcleo da computação apresente o maior volume de publicações, a pesquisa interdisciplinar com ciências da vida alcança um impacto significativamente superior em termos de citações e índice h. Adicionalmente, o estudo encontrou uma forte correlação entre a excelência dos programas (nota CAPES) e a diversidade temática, com programas de nota 7 exibindo pesquisadores em todas as comunidades identificadas. Estes resultados oferecem um mapa detalhado da pesquisa em computação no país e sugerem que o fomento à diversidade e à interdisciplinaridade são estratégias chave para o avanço científico

Palavras-chave: Ciência da Ciência. Pós-graduação. Aprendizado de Máquina. BERT. Processamento de Linguagem Natural.

Abstract

The field of Science of Science has gained significant relevance by analyzing research data to understand the factors that drive scientific success. This monograph applies Natural Language Processing and unsupervised learning techniques to map the thematic structure and impact profiles of the community of researchers associated with Computer Science postgraduate programs in Brazil. Using data from 1,511 researchers, extracted from the OpenAlex database, the Lattes platform, and the CAPES quadrennial evaluation report, embeddings were generated for each researcher using the SciBERT model based on their research topics. The application of the HDBSCAN clustering algorithm on these embeddings revealed the existence of 6 distinct thematic communities, including a large central cluster focused on the fundamentals of computing and interdisciplinary groups at the interface with life sciences, physics, and humanities. Quantitative analysis showed that while the core of computing has the highest volume of publications, interdisciplinary research with life sciences achieves a significantly higher impact in terms of citations and h-index. Furthermore, the study found a strong correlation between the excellence of the postgraduate programs (CAPES score) and their thematic diversity, with top-rated programs (grade 7) showing representation across all identified communities. These results provide a detailed map of computer science research in the country and suggest that fostering diversity and interdisciplinarity are key strategies for scientific advancement.

Keywords: Science of Science, Postgraduate, Machine Learning, BERT, Natural Language Processing.

Lista de Ilustrações

LISTA DE ABREVIATURAS E SIGLAS

DECOM	Departamento de Computação
UFOP	Universidade Federal de Ouro Preto
CSILab	Laboratório de Computação de Sistemas Inteligentes
PPG	Programa de Pós-Graduação
XML	Extensible Markup Language
CNPq	Conselho Nacional de Desenvolvimento Científico e Tecnológico
URL	Uniform Resource Locator
ID	Identity
ORCID	Open Researcher and Contributor ID
API	Application Programming Interface
SciSci	Science of Science
NMI	Normalized Mutual Information
MI	Mutual Information
CEID	Canonical External ID
DOI	Digital Object Identifier
ISSN	International Standard Serial Number
CBOW	Continuous Bag of Words
BERT	Bidirectional Encoder Representations from Transformers
PLN	Processamento de Linguagem Natural
PDF	Portable Document Format

Sumário

1 Introdução

Nos últimos anos, com o aumento da disponibilidade de dados de pesquisas científicas, a análise desses dados tem emergido como um campo de estudo importante e multifacetado, proporcionando percepções sobre as dinâmicas da pesquisa acadêmica. Essa área de estudo é chamada de Ciência da Ciência do inglês *Science of Science* (SciSci), e se vale da hipótese de que com uma compreensão mais profunda dos fatores por trás do sucesso da ciência, podemos melhorar as perspectivas da ciência como um todo para ser mais efetiva na resolução de problemas sociais (FORTUNATO et al., 2018).

Para a presente monografia foi construída uma base de dados em colaboração com outras pesquisas como Guilarducci et al. (2025) e Vasconcelos et al. (2025). Para isso foram utilizadas as informações extraídas da OpenAlex [1] (PRIEM; PIWOWAR; ORR, 2022), que é uma fonte de metadados acadêmicos com API (Application Programming Interface) e código fonte abertos. A OpenAlex é organizada como um grafo direcionado e heterogêneo composto por cinco entidades acadêmicas que são *Works, Authors, Venues, Institutions* e *Concepts*, e as conexões entre elas.

Além disso, na montagem da base de dados também foram utilizados dados retirados da plataforma Lattes [2], que é disponibilizada pelo CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico), e reúne dados de pesquisadores que atuam no Brasil e de seus respectivos trabalhos. Também, utiliza-se o relatório de avaliação quadrienal de 2021 [3] realizado pela CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) e que tem seus dados disponibilizados na plataforma Sucupira [4], mais especificamente o relatório de ciência da computação, que avalia os Programas de Pós-Graduação (PPGs) em computação no Brasil com notas de 3 a 7.

A literatura recente, como evidenciado por estudos como Alsentzer et al. (2019) e Khattak et al. (2019), tem aplicado técnicas avançadas de processamento de linguagem natural (PLN), como BERT (DEVLIN et al., 2018), GPT (RADFORD et al., 2018), e outras variantes de modelos de linguagem baseados em *transformers* (VASWANI et al., 2017), para criar representações vetoriais a partir de textos. Essas representações, conhecidas como *embeddings*, oferecem uma rica fonte de dados para análise de padrões e tendências. Estes tipos de representações podem ser usadas em dados acadêmicos, para vetorizar dados de autores como resumos de artigos e seus tópicos de pesquisa e buscar nesses *embeddings* características entre esses autores.

Entretanto, apesar dos avanços significativos, observa-se um *gap* na literatura quanto

[1] <https://openalex.org/>
[2] <https://lattes.cnpq.br/>
[3] <https://www.gov.br/capes/pt-br/acesso-a-informacao/acoes-e-programas/avaliacao/avaliacao-quadrienal/resultado-da-avaliacao-quadrienal-2017-2020>
[4] <https://sucupira.capes.gov.br/sucupira/>

à aplicação dessas técnicas para investigar se os *clusters* formados a partir desses *embeddings* para professores de PPGs em computação no Brasil podem explicar características específicas dos autores, como área de atuação, status de bolsista de produtividade, número de colaborações, número de citações, índice h, número de artigos publicados, entre outros.

Essa pesquisa concentra-se em explorar o espaço de *embeddings* de autores de artigos científicos brasileiros, mais especificamente, autores credenciados como professores em PPGs no campo da computação no Brasil. São gerados e analisados conjuntos de *embeddings* desses professores, em busca de características que conectem esses autores e seus respectivos programas de pós-graduação.

Dessa forma, este estudo busca responder à seguinte Questão de Pesquisa Primária (QPP): "Técnicas de Processamento de Linguagem Natural como a geração de *embeddings* juntamente com técnicas de aprendizado não supervisionado como a *clusterização* sobre dados de produção científica pode revelar a estrutura temática e os perfis de impacto da comunidade de professores credenciados a Programas de Pós-Graduação em computação no Brasil?". Para isso, a presente monografia busca responder as seguintes Questões de Pesquisa Secundárias (QPSs):

- **Questão de Pesquisa Secundária 1 (QPS1)**: Podemos identificar e caracterizar, de forma não supervisionada, comunidades temáticas distintas de pesquisadores em computação no Brasil a partir de *embeddings* gerados de seus tópicos de pesquisa mais frequentes?
- **Questão de Pesquisa Secundária 2 (QPS2)**: Os perfis de produtividade e impacto acadêmico (medidos por número de trabalhos publicados, contagem de citações, índice h e índice i10) variam significativamente entre as diferentes comunidades temáticas identificadas?
- **Questão de Pesquisa Secunrária 3 (QPS3)**: Existe uma correlação entre a excelência de um Programa de Pós-Graduação em Computação no Brasil (medida pela nota atribuída pela CAPES) e a diversidade temática de seus pesquisadores?

As respostas a estas perguntas podem oferecer novas perspectivas sobre a estrutura e a dinâmica das comunidades científicas de computação no Brasil, contribuindo para uma compreensão mais profunda das interações acadêmicas e suas implicações.

1.1 Justificativa

A justificativa para este estudo reside na crescente importância da análise de redes formadas por cientistas (FORTUNATO et al., 2018) no entendimento das dinâmicas de pesquisa e colaboração no ambiente acadêmico. A capacidade de mapear e entender as relações e influências entre pesquisadores de diferentes regiões geográficas e áreas de atuação é fundamental para a promoção de uma comunidade científica mais integrada e colaborativa. Além disso, a identificação

de padrões em termos de produtividade, colaborações e impacto científico pode auxiliar na formulação de políticas e estratégias para o fomento da pesquisa (FORTUNATO et al., 2018).

A utilização de técnicas avançadas de PLN e aprendizado de máquina, como a geração de *embeddings* a partir de *concept tags* extraídas da OpenAlex, pode representar um avanço na análise de dados científicos. No entanto, ainda existe uma lacuna no que se refere à aplicação dessas técnicas para explorar características específicas dos autores e suas interconexões. Este estudo, portanto, visa oferecer uma análise detalhada para o problema.

1.2 Objetivos

1.2.1 Objetivo Geral

O objetivo geral desta monografia é investigar se os *clusters* de *embeddings* de professores credenciados em PPGs de Computação do Brasil, em especial os atuantes em Mestrado e/ou Doutorado acadêmicos, gerados utilizando técnicas de processamento de linguagem natural a partir de dados da OpenAlex (FORTUNATO et al., 2018), plataforma Lattes e relatório da CAPES, podem explicar os perfis de impacto e a estrutura da comunidade dos autores.

1.2.2 Objetivos Específicos

Os objetivos específicos deste projeto são:

- Construir uma base de dados de professores dos programas de pós-graduação em computação do Brasil a partir de dados extraídos da OpenAlex, Plataforma Lattes e relatório da CAPES de avaliação dos programas de pós-graduação de computação no Brasil;
- Utilizar modelos baseados em *transformers* para criar *embeddings* de autores com base em seus tópicos de pesquisa mais frequentes;
- Agrupar os *embeddings* gerados em *clusters* para identificar padrões e agrupamentos naturais entre os autores;

1.3 Organização da Monografia

O restante do documento está organizado na seguinte forma: o Capítulo 2 apresenta o embasamento teórico e os trabalhos relacionados. O Capítulo 3: descreve a Metodologia que foi aplicada no desenvolvimento do trabalho. No Capítulo 4 são apresentados os resultados obtidos na monografia. Por fim, o Capítulo 5 apresenta as conclusões desta pesquisa e aponta trabalhos futuros.

1.3.1 Estrutura da Monografia

O presente trabalho segue a seguinte estrutura:

Capítulo 1: Introdução.

Capítulo 2: Embasamento teórico e trabalhos relacionados.

Capítulo 3: Desenvolvimento.

Capítulo 4: Resultados e Discussões.

Capítulo 5: Consclusões finais.

2 Revisão Bibliográfica

2.1 Fundamentação Teórica

2.1.1 Science of Science

Como já mencionado anteriormente, a presente monografia é um projeto de Ciência da Ciência (SciSci), que segundo Fortunato et al. (2018) é um campo de estudo que utiliza de grandes conjuntos de dados sobre produções científicas para estudar os mecanismos por trás da prática da ciência. A SciSci abrange desde escolhas de problemas de pesquisa até trajetórias profissionais e o progresso em determinada área, com o objetivo de aumentar a capacidade de êxito dos cientistas e melhorar as perspectivas da ciência como um todo para abordar de forma mais eficaz os problemas sociais.

Para Fortunato et al. (2018) a SciSci foi impulsionada por dois fatores principais: o aumento da disponibilidade de dados científicos e o fácil acesso a eles; a colaboração de cientistas de diferentes áreas que desenvolveram habilidades baseadas em grandes volumes de dados, analisando e gerando modelos que conseguem captar o desenvolvimento da ciência. Esse campo de estudo integra diferentes áreas, dados e técnicas, como ciência das redes e aprendizado de máquina, que é o caso da presente pesquisa, que visa através de técnicas de aprendizado de máquina buscar padrões e entender a dinâmica da pesquisa científica na pós-graduação em computação no Brasil.

2.1.2 OpenAlex

Para a construção da base de dados científica utilizada nesse projeto de SciSci foram extraídos dados da OpenAlex, que é uma fonte de metadados acadêmicos e está descrita detalhadamente em Priem, Piwowar e Orr (2022). Os autores retratam que a OpenAlex é totalmente aberta (cem por cento dados abertos, API aberta e código fonte aberto) e pode ser acessada de forma gratuita e sem a necessidade de um cadastro através da API, de um download completo de um *snapshot* da base de dados ou de uma interface gráfica disponibilizada na web, o que garante a reprodutibilidade desta monografia.

A base de dados está organizada em forma de um grafo direcionado heterogêneo formado de cinco entidades como retratado na Figura 2.1. Cada entidade possui um ID próprio da OpenAlex que funciona como uma chave primária dentro da base de dados e um CEID (Canonical External ID) sempre que possível, que é um ID canônico de um sistema externo utilizado para aumentar a interoperabilidade, além de possuírem outros IDs externos quando factível.

As entidades presentes na base de dados da OpenAlex são: *works*, que representam

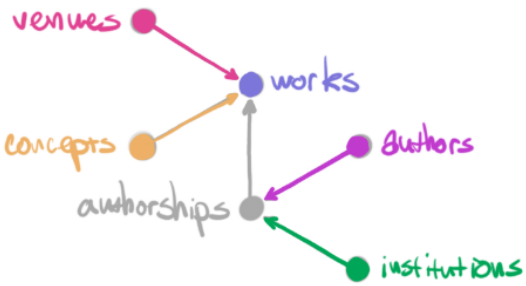

Figura 2.1 – Esquema do modelo de dados da OpenAlex (PRIEM; PIWOWAR; ORR, 2022).

documentos acadêmicos como periódicos, livros, conjuntos de dados e teses e tem como CEID o DOI (Digital Object Identifier); *authors*, que são pessoas que produzem *works*, o CEID para *authors* é o ORCID; *venues* (atualmente *sources* segundo o manual da OpenAlex [1]), que são definidos como locais que hospedam *works* e tem como CEID o *linking* ISSN (International Standard Serial Number); *institutions* que são organizações das quais autores são afiliados e o seu CEID é o ROR (Research Organization Registry) ID e *concepts* que são abstrações das ideias sobre o que os trabalhos tratam e o CEID dos *concepts* é o *wikidata* ID (PRIEM; PIWOWAR; ORR, 2022). Os dados recentes de cada entidade podem ser acessados pelo site de status disponibilizado pela plataforma como mostra a Figura 2.2.

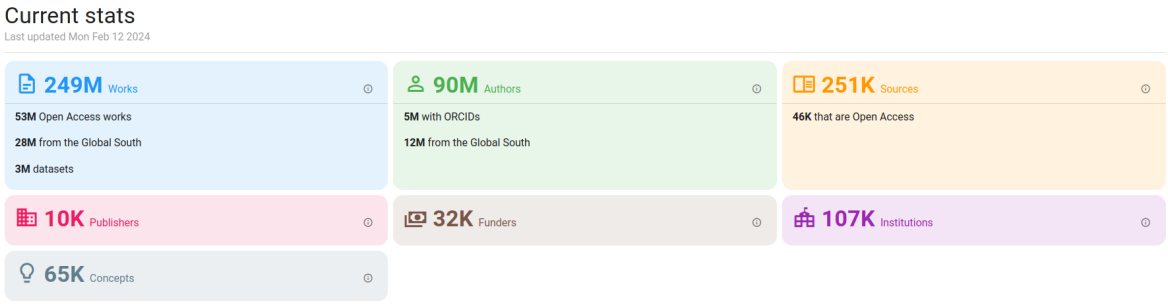

Figura 2.2 – Estado dos dados da OpenAlex acessado no dia 12/02/2024 em <https://openalex.org/stats>.

2.1.2.1 Authors e Works

Uma das principais entidades fornecidas pela OpenAlex que será utilizada nesse trabalho para extrair dados de publicações dos pesquisadores é a *authors*. Alguns dos campos do objeto *authors* que são de interesse neste trabalho são:

- ***affiliations*** - é uma lista de objetos representando instituições em que o autor declarou filiação em suas publicações. Contém dados da instituição como ID da OpenAlex, ROR,

[1] <https://help.openalex.org/>

nome da instituição, código do país, tipo da instituição e uma lista com os anos que o autor declara afiliação à essa instituição, como pode ser visto no exemplo da Figura 2.3.

```
affiliations: [
    {
        institution: {
            id: "https://openalex.org/I201448701",
            ror: "https://ror.org/00cvxb145",
            ...
        },
        years: [2018, 2019, 2020]
    },
    {
        institution: {
            id: "https://openalex.org/I74973139",
            ror: "https://ror.org/05x2bcf33",
            ...
        },
        years: [2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019]
    }
]
```

Figura 2.3 – Exemplo do campo *affiliations* dentro do objeto *author* retirado de <https://docs.openalex.org/api-entities/authors/author-object>.

- ***cited_by_count*** - é um número inteiro que representa o número de trabalhos que citou algum dos trabalhos desse autor.
- ***display_name*** - é uma *string* única contendo o nome do pesquisador.
- ***id*** - é o ID da OpenAlex para esse autor.
- ***ids*** - é uma lista contendo a *string* de todos os IDs externos encontrados para esse autor. Quando possível essa string é representada por uma URL. Temos como possíveis IDs externos: o ID da OpenAlex, o ORCID, o scopus, o twitter e a página na wikipedia. Quando o pesquisador não possui algum desses IDs, o campo simplesmente não existe dentro do objeto.
- ***summary_stats*** - são métricas de citações para esse pesquisador, como o fator de impacto, o índice h e o índice i10.
- ***works_count*** - o número de trabalhos que esse cientista publicou.
- ***x_concepts*** - é uma lista de conceitos mais utilizados nos trabalhos desse pesquisador, sendo alguns dos campos de cada objeto: o ID do conceito na OpenAlex, o nome do conceito (Ex: "*Computer science*", "*Artificial intelligence*" e "*Machine learning*") e o *score* desse conceito para o autor, que é um número real que varia de 0 a 100 e representa a força

de associação entre esse autor e o conceito. Um exemplo completo de *x_concepts* para um determinado pesquisador pode ser visto na Figura 2.4.

```
x_concepts: [
    {
        id: "https://openalex.org/C41008148",
        wikidata: null,
        display_name: "Computer science",
        level: 0,
        score: 97.4
    },
    {
        id: "https://openalex.org/C17744445",
        wikidata: null,
        display_name: "Political science",
        level: 0,
        score: 78.9
    }
]
```

Figura 2.4 – Exemplo do campo *x_concepts* dentro do objeto *author* retirado de <https://docs.openalex.org/api-entities/authors/author-object>.

Um dos campos dentro da entidade *author* é o *works_api_url*, que é uma URL que nos retorna um objeto contendo os campos "*meta*" e o campo "*results*", que é uma lista contendo todos os trabalhos deste pesquisador. Um exemplo de resultado para essa requisição pode ser visto na Figura 2.5.

O objeto *meta* retorna informações que nos ajudam a navegar pelos resultados, como a quantidade de trabalhos encontrados, a página atual (que pode ser mudada utilizando o argumento *page* na requisição) e a quantidade de resultados por página (que pode ser alterado via parâmetro com um valor entre 1 e 200). A abordagem de paginação suporta no máximo 10.000 resultados. Para mais que isso é necessário utilizar a abordagem de paginação com cursor, que a cada requisição retorna um *token* de cursor para acessar a próxima página.

Já o objeto *results* retorna uma lista de objetos *work*, que são as obras deste autor. Alguns campos do objeto *work* que são de interesse para esse trabalho são:

- ***abstract_inverted_index*** - O *abstract* (resumo) do trabalho no formato de *inverted index*, que codifica a informação sobre as palavras do *abstract* e suas posições no texto, como podemos ver na Figura 2.6.
- ***authorships*** - uma lista de objetos *authorship* que contém dados para cada autor como o ID na OpenAlex, o nome, a posição na lista de autores do trabalho, e seu país. Dentro do objeto *authorship* também é possível encontrar o objeto *insitutions* que contém dados das instituições afiliadas pelos cientistas no contexto desse trabalho.

```
{
    "meta": {
        "count": 245684392,
        "db_response_time_ms": 929,
        "page": 1,
        "per_page": 25
    },
    "results": [
        {
            "id": "https://openalex.org/W1775749144",
            "doi": "https://doi.org/10.1016/s0021-9258(19)52451-6",
            "title": "PROTEIN MEASUREMENT WITH THE FOLIN PHENOL REAGENT",
            // more fields (removed to save space)
        },
        {
            "id": "https://openalex.org/W2100837269",
            "doi": "https://doi.org/10.1038/227680a0",
            "title": "Cleavage of Structural Proteins during the Assembly of the Head o
            // more fields (removed to save space)
        },
        // more results (removed to save space)
    ],
    "group_by": []
}
```


Figura 2.5 – Exemplo do resultado da requisição da URL do campo *works_api_url* dentro da entidade *author* retirado de <https://docs.openalex.org/api-entities/works/get-lists-of-works>.

- ***best_oa_location*** - é um objeto com a melhor localização de acesso livre disponível para este trabalho seguindo critérios estabelecidos pela plataforma OpenAlex. Contém dados do trabalho como a URL da página web, a versão que pode ser: "*publishedVersion*", "*acceptedVersion*"e "*submittedVersion*"; uma URL com um link direto para o PDF (Portable Document Format) do trabalho quando disponível, a sua licença e um objeto *source* que contém informações da fonte de onde foi retirado o trabalho como o ID dessa fonte na OpenAlex, o nome e a organização que hospeda a fonte.

- ***cited_by_api_url*** - é uma URL que sua requisição retorna uma lista de obras que cita este trabalho.

- ***cited_by_count*** - o número de citações que esta obra recebeu.

- ***concepts*** - uma lista de conceitos atribuídos a este trabalho assim como os *x_concepts* da entidade *author* possui os campos de interesse: ID do conceito na OpenAlex, o *score* do conceito em relação ao trabalho e o nome do conceito.

- ***corresponding_author_ids*** - uma lista com os IDs da OpenAlex para cada autor da obra.

```
abstract_inverted_index: {
    Despite: [
        0
    ],
    growing: [
        1
    ],
    interest: [
        2
    ],
    in: [
        3,
        57,
        73,
        110,
        122
    ],
    Open: [
        4,
        201
    ],
    Access: [
        5
    ],
    ...
}
```

Figura 2.6 – Exemplo do campo *abstract_inverted_index* dentro do objeto *work* retirado de <https://docs.openalex.org/api-entities/works/work-object>.

- ***corresponding_institutions_ids*** - uma lista com os IDs da OpenAlex para cada instuição dos autores no contexto do trabalho.
- ***doi*** - o DOI para esta obra.
- ***id*** - o ID da OpenAlex para este trabalho.
- ***keywords*** - uma lista com as palavras-chave extraídas do título da obra, juntamente com um *score* que indica o quão confiável é essa palavra chave.
- ***language*** - o idioma que o texto foi escrito.
- ***ngrams_url*** - uma URL que retorna uma lista de grupos de palavras e frases (objetos do tipo *ngram*) que compõem o trabalho. O objeto *ngram* é composto dos campos: ngram, que é um grupo de palavras, números ou letras que existem ao longo do texto e pode variar entre 1 e 5 *tokens*; *ngram_count*, que indica quantas vezes esse *ngram* aparece no texto; ngram_tokens, que é um valor inteiro com o número de *tokens* que o *ngram* é composto e *term_frequency*, que é a frequência que esse *ngram* ocorreu na obra.
- ***referenced_works*** - uma lista com os IDs da OpenAlex para os trabalhos que esta obra cita.

- ***related_works*** - uma lista com os IDs da OpenAlex para os trabalhos relacionados a esta obra. Esta lista é obtida através de um algoritmo da plataforma que filtra os artigos recentes com conceitos em comuns com a obra em questão.

- ***topics*** - uma lista com os melhores tópicos que representam este trabalho com no máximo três itens, um exemplo pode ser visto na Figura 2.7.

```
topics: [
    {
        id: "https://openalex.org/T12419",
        display_name: "Analysis of Cardiac and Respiratory Sounds",
        score:  0.9997,
        subfield: {
            id: 2740,
            display_name: "Pulmonary and Respiratory Medicine"
        }
        field: {
            id: 27,
            display_name: "Medicine"
        }
        domain: {
            id: 4,
            display_name: "Health Sciences"
        }
    }
    ...
]
```

Figura 2.7 – Exemplo do campo *topics* dentro do objeto *work* retirado de <https://docs.openalex.org/api-entities/works/work-object>.

- ***title*** - o título completo da obra em formato de *string*.

- ***type*** - o tipo do trabalho, alguns exemplos são: *article*, *book-chapter*, *dissertation*, *book*, *paratext*, *dataset*, *report*, etc.

2.1.3 Tokens e Vocabulário

Documentos são sequências de caracteres que podem ser uma frase, um parágrafo, um livro ou qualquer tipo de texto. O conjunto de todos os documentos de interesse forma o conjunto de dados, mais conhecido como *corpus* (SAUL, 2023).

Os *tokens* são unidades básicas e indivisíveis de um documento e definir o que constitui um *token* é uma escolha de projeto. Por exemplo, podemos escolher representar cada *token* de um texto, sendo cada um, uma palavra desse texto, ou simplesmente cada caractere do texto (ZHANG et al., 2023), sendo o mais usual utilizar as palavras do *corpus* como *tokens*. O conjunto de todos

os *tokens* exclusivos em todos os documentos do nosso *corpus* formam o nosso vocabulário (ou dicionário).

2.1.4 One-hot encoding

Inicialmente, as palavras, ou *tokens*, eram representadas utilizando vetores *one-hot*. Nesse sentido, dado um vocabulário, de N palavras, onde cada palavra corresponde a um número inteiro diferente, que pode variar de 0 a $N-1$, o vetor de representação de uma palavra de índice i pode ser obtido tomando um vetor de $0's$ e atribuindo ao elemento na posição i o valor de 1 (ZHANG et al., 2023).

Como exemplo do *one-hot encoding*, dada a frase "Eu gosto de cães e de gatos" e o vocabulário ["Eu", "gosto", "de", "cães", "e", "gatos"], a representação para a frase poderia ser dada pelo seguinte conjunto de vetores *one-hot*:

$$\{\begin{bmatrix}1\\0\\0\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\1\\0\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\1\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\1\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\0\\1\\0\end{bmatrix}, \begin{bmatrix}0\\0\\1\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\0\\0\\1\end{bmatrix}\}.$$

Mesmo que os vetores de palavras *one-hot* sejam fáceis de construir e possam ser usados de entrada paras as redes neurais, eles não são uma boa escolha, pois, não conseguem expressar similaridade entre duas palavras diferentes (ZHANG et al., 2023), ou seja, não conseguem capturar informações sintáticas e semânticas das palavras. Além disso, não são escaláveis, visto que a dimensão dos vetores é dada pelo tamanho do vocabulário, e em problemas reais temos vocabulários de tamanhos elevados.

2.1.5 Word Embeddings

A linguagem natural é um sistema complexo utilizado para representar significados, onde a palavra é a unidade básica de significado. *Word vectors* são vetores utilizados para representar uma palavra e suas características computacionalmente. A técnica de mapear palavras em vetores de valores reais é chamada de *word embedding* (ZHANG et al., 2021). Essa técnica mapeia cada palavra do *corpus* em um vetor de tamanho pré-determinado que deve ser passado ao modelo no formato de um hiper-parâmetro (SAUL, 2023).

Com isso, a ideia é de que ao representar características das palavras, os *word embeddings* captem informações sintáticas ou semânticas das mesmas em suas diferentes dimensões, e as representações vetoriais de palavras com sentidos e contextos semelhantes fiquem próximas umas às outras no espaço vetorial multidimensional, como por exemplo, as relações de gênero e conjugação verbal que podem ser vistas na Figura 2.8.

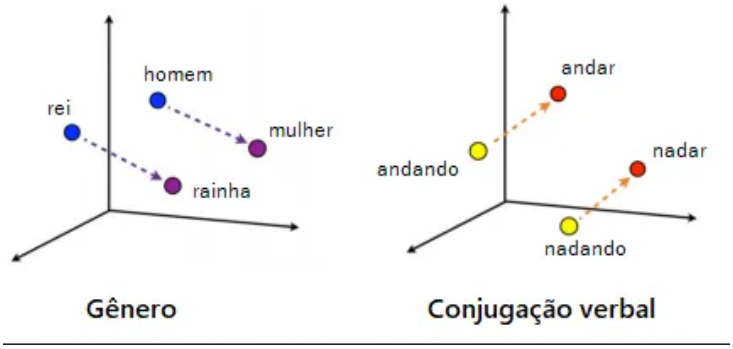

Figura 2.8 – Exemplo de relações semânticas em *embeddings*. Fonte: (FONSECA, 2021).

Com isso, os *word embeddings* suprem os problemas apresentados por representações como o *one-hot encoding*, que não conseguia retirar significado das palavras e tinha dimensões muito maiores quando comparado aos vetores de *embeddings*. Além disso, ganhamos propriedades úteis com os *embeddings*, como as analogias entre palavras, por exemplo: rainha está para rei assim como mulher está para homem. Isso pode ser feito através de medidas de similaridade entre vetores, como por exemplo a similaridade de cossenos, que permite calcular o cosseno do ângulo entre dois vetores utilizando um produto interno normalizado e tem como resultado um valor entre -1 e 1, sendo que quanto mais próximo de 1 maior a similaridade entre os vetores, enquanto que cossenos próximos a -1 indica o oposto, conforme segue:

$$cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \, \|\mathbf{B}\|} = \frac{\sum_{i=1}^{N} A_i B_i}{\sqrt{\sum_{i=1}^{N} A_i^2} \sqrt{\sum_{i=1}^{N} B_i^2}}, \tag{2.1}$$

onde A_i e B_i são os i-ésimos valores dos vetores A e B respectivamente.

2.1.6 Word2Vec

O *word2vec* (MIKOLOV et al., 2013a) foi um dos primeiros algoritmos para gerar *embeddings* e foi proposto para resolver os problemas de métodos de representações de palavras como o *one-hot encoding*. Ele mapeia cada palavra para um vetor de comprimento fixo e esses *embeddings* podem expressar melhor semelhanças e relacionamentos entre diferentes palavras (ZHANG et al., 2023).

O *word2vec* possui dois modelos de treinamento: o *skip-gram* (MIKOLOV et al., 2013b) e o *continuous bag of words (CBOW)* (MIKOLOV et al., 2013a). Para a representação de significado semântico das palavras o *word2vec* se baseia nas probabilidades condicionais de que podemos predizer algumas palavras utilizando palavras que estão ao seu redor dentro do texto. Tanto o *skip-gram* quanto o CBOW são modelos auto-supervisionados (*self-supervised*), visto que aprendem com um conjunto de dados que não é explicitamente rotulado (ZHANG et al., 2023).

O modelo *skip-gram* assume que uma palavra pode ser usada para gerar as palavras ao seu redor em uma sequência de texto. Já o modelo CBOW assume que uma palavra central pode ser gerada a partir das palavras ao seu redor em uma sequência de texto.

2.1.7 Transformers

Modelos como o *word2vec* atribuem o mesmo vetor pré-treinado para a mesma palavra, independente do contexto da palavra, se esse existir. Formalmente, temos que uma representação sem contexto de qualquer *token* x é uma função $f(x)$ que recebe apenas x como entrada. Com a complexidade polissêmica e semântica na linguagem natural, modelos de representação independentes de contexto possuem limitações (ZHANG et al., 2021). Por exemplo, a palavra "banco" nas frases "a menina sentou no banco" e "o menino foi ao banco sacar dinheiro" tem significado completamente diferente nas duas frases; com isso, a mesma palavra poderia receber diferentes representações de acordo com seu contexto. Isso motiva o desenvolvimento de modelos de representação de palavras sensíveis ao contexto, onde a representação das palavras depende de seus contextos. Essa representação formalmente é uma função $f(x, c(x))$, sendo x o *token* e $c(x)$ seu contexto (ZHANG et al., 2021). A arquitetura *Transformer*, proposta por Vaswani et al. (2017) surge para preencher essa lacuna, gerando representações que levam em consideração o contexto dos *tokens*.

Antes do *Transformer*, os modelos como as Redes Neurais Recorrentes (RNNs), processavam o texto sequencialmente, o que impedia a paralelização e tornava lento o treinamento desses modelos, além de dificultar a captura de dependências entre *tokens* distantes nos textos. (VASWANI et al., 2017). O *Transformer* baseia-se exclusivamente nos mecanismos de atenção, eliminando a recorrência. E, é baseado em uma arquitetura composta por uma pilha de codificadores do inglês *encoders* e uma pilha de decodificadores do inglês *decoders* conforme a Figura 2.9.

O componente principal da arquitetura *Transformer* é a camada de *self-attention*. Ela permite que o modelo, ao processar uma palavra pondere a importância de todas as outras palavras da sequência. O *Multi-Head Attention* executa esse processo várias vezes em paralelo para que o modelo se concentre em diferentes aspectos do texto.

2.1.8 BERT

O BERT (*Bidirectional Encoder Representations from Transformers*) surge como um modelo que consegue codificar o contexto de um *token* bidirecionalmente e requer mudanças mínimas em sua arquitetura para operar em uma ampla gama de tarefas de PLN (DEVLIN et al., 2018). Usando uma camada de *encoder* da arquitetura *Transformers* (VASWANI et al., 2017) pré-treinada, o BERT pode representar qualquer *token* baseado em seu contexto bidirecional (ZHANG et al., 2021).

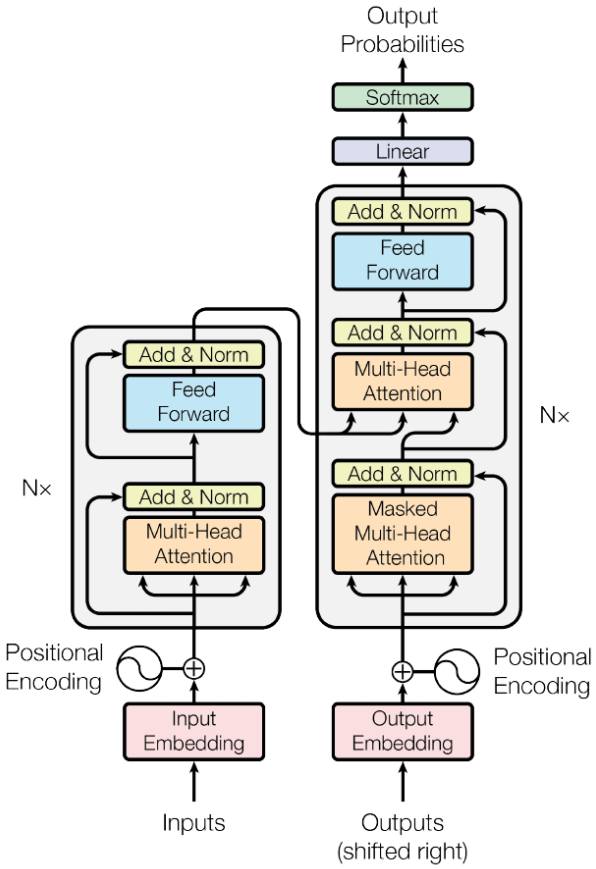

Figura 2.9 – Arquitetura *Transformer* (VASWANI et al., 2017).

O BERT permite como entrada um único texto ou um par de textos, isso porque, algumas tarefas de PLN, como análise de sentimentos, utiliza uma única entrada. Já outras tarefas como inferência de linguagem natural, necessitam de um par de sentenças como entrada (ZHANG et al., 2021). Uma entrada composta de uma única sentença deve ser feita na forma: *token* especial "$< cls >$", seguido dos *tokens* da sentença de entrada e por fim o *token* especial "$< sep >$". Já uma entrada composta de um par de sentenças deve ser feita na forma: *token* especial "$< cls >$", seguido dos *tokens* do primeiro texto, seguido do *token* especial "$< sep >$", os *tokens* do segundo texto de entrada e por fim de "$< sep >$" (ZHANG et al., 2023). Para diferenciar os pares de textos, os *segment embeddings* (segmentos de *embeddings*) e_A e e_B são adicionados aos *token embeddings* (*tokens* de *embeddings*) da primeira e da segunda sentença, respectivamente. Para entradas de apenas um texto, apenas o e_A é usado.

O BERT usa o *Transformer encoder* (VASWANI et al., 2017) como arquitetura bidirecional, e assim como ele, adiciona *positional embeddings* (*embeddings* posicionais) para cada posição da sua sequência de entrada. Porém, diferente do *encoder Transformer* original, o BERT usa *embeddings* posicionais aprendíveis. Como pode ser visto na Figura 2.10, os *embeddings* da sequência de entrada do BERT são uma soma dos "*token embeddings*", "*segment embeddings*" e "*positional embeddings*"(ZHANG et al., 2023).

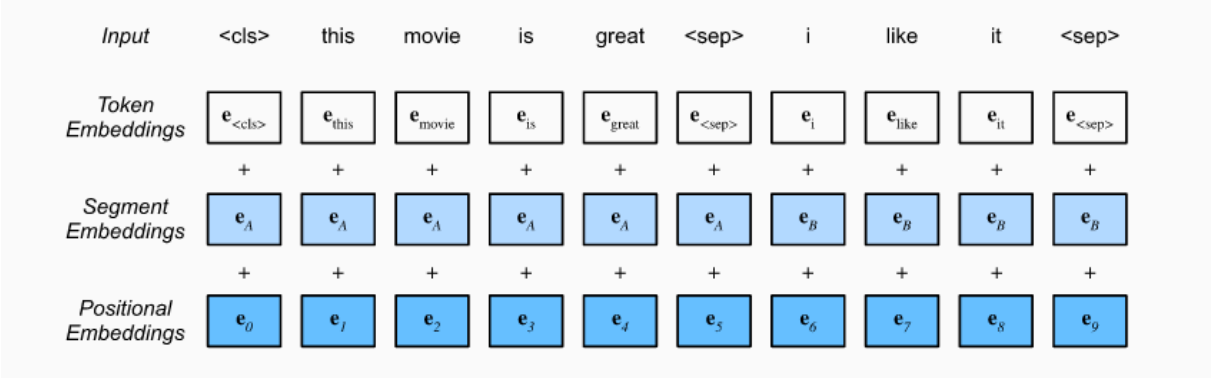

Figura 2.10 – *Embeddings* da camada de entrada do BERT (ZHANG et al., 2023).

A representação de sequência de entrada mostrada anteriormente, é utilizada para calcular a função de perda na fase de pré-treinamento do BERT. A fase de pré-treinamento é composta por duas tarefas: *masked language modelling* e *next sentence prediction*. Após o pré-treinamento, o BERT pode ser adaptado para tarefas específicas através do *fine-tuning*, onde uma camada de saída é adicionada e os parâmetros do modelo são ajustados nos dados rotulados da tarefa (DEVLIN et al., 2018).

2.1.8.1 Masked Language Modeling

A técnica *masked language model* é utilizada pelo BERT para codificar o contexto bidirecionalmente para representar cada *token*. O BERT aleatoriamente mascara *tokens* e utiliza os *tokens* de contexto bidirecional para predizer os *tokens* mascarados (ZHANG et al., 2023).

Na fase de pré-treinamento, 15% dos *tokens* são selecionados aleatoriamente como *tokens* mascarados para a predição. Para predizer o *token* sem "trapacear" o BERT o substitui por um *token* especial "$< mask >$". Como o *token* especial "$< mask >$" não deve aparecer na camada de ajuste fino, o BERT segue a seguinte abordagem para evitar esse problema quando um *token* é selecionado para ser mascarado: 80% das vezes ele recebe o *token* especial "$< mask >$"; 10% das vezes ele recebe um *token* aleatório e 10% das vezes ele permanece inalterado (ZHANG et al., 2023).

2.1.8.2 Next Sentence Prediction

Por fim, para ajudar a entender a relação entre duas sequências de texto, o BERT utiliza a *next sentence prediction* como tarefa de classificação binária, que tenta predizer se duas sequências são realmente consecutivas. Ou seja, ao gerar pares de sentenças, na metade do tempo elas são verdadeiramente consecutivas com rótulo "verdadeiro"; enquanto, na outra metade do tempo a segunda frase é selecionada aleatoriamente com o rótulo "falso" e as sentenças não são verdadeiramente consecutivas (ZHANG et al., 2021).

2.1.9 SciBERT

Apesar da eficácia do BERT, seu desempenho é focado em textos de domínio geral, por exemplo da *Wikipedia*, que foi uma das fontes de dados utilizadas para o seu pré-treinamento. Dados científicos possuem vocabulário e estrutura próprios. Para ajudar com esse desafio, Beltagy, Lo e Cohan (2019) desenvolveram o SciBERT, um modelo BERT pré-treinado em uma base de dados de artigos científicos. Os autores descobriram que mais da metade das palavras mais comuns nos textos científicos não estão presentes no BERT original. Ao ser treinado nessa base de dados específica, o SciBERT obteve melhorias estatisticamente significativas em relação ao BERT em uma variedade de tarefas de PLN no domínio científico (BELTAGY; LO; COHAN, 2019).

2.1.10 UMAP

Os *embeddings* gerados pelo SciBERT possuem centenas de dimensões. Para visualizar e explorar esses dados, é necessário projetá-los em um espaço de baixa dimensionalidade, 2D no nosso caso. O UMAP (Uniform Manifold Approximation and Projection), criado por McInnes, Healy e Melville (2018), é um algoritmo de redução de dimensionalidade com uma robusta base matemática e ótimos resultados práticos.

O UMAP baseia-se na teoria do aprendizado *manifold* (manifold learning), que assume que os dados de alta dimensão, residem em uma esturutra geométrica de baixa dimensão dentro do espaço de maior dimensão. O objetivo do método é encontrar uma projeção de baixa dimensão que preserve a estrutura topológica essencial desse *manifold*. Para isso, o UMAP constrói uma representação topológica dos dados no espaço original e, em seguida, otimiza a representação de baixa dimensão para que ela seja o mais estruturalmente similar à original. O UMAP apresenta vantagens significativas quando comparado a outras técnicas de redução de dimensionalidade populares, como melhor preservação da estrutura global dos dados e alta eficiência e escalabilidade (MCINNES; HEALY; MELVILLE, 2018).

2.1.11 HDBSCAN

Para identificar quantitativamente e objetivamente as comunidades de pesquisa em computação no Brasil é necessário um algoritmo de agrupamento. O HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise) (CAMPELLO; MOULAVI; SANDER, 2013) é um algoritmo de clusterização moderno que se baseia em estimativas de densidade para encontrar agrupamentos de diferentes formas e tamanhos.

O seu antecessor, o DBSCAN, sofre um limitação, a necessidade de um parâmetro de distância global (*epsilon*). Isso impede que o DBSCAN identifique *clusters* em diferentes níveis de densidade, o que é comum em dados do mundo real. O HDBSCAN supera essa limitação. Em vez de encontrar agrupamentos de uma única densidade, ele constrói uma hierarquia

completa de todas as possíveis partições que o DBSCAN encontraria para todos os valores de *epsilon* possíveis. A partir dessa hierarquia, ele extrai os *clusters* mais estáveis e persistentes (CAMPELLO; MOULAVI; SANDER, 2013).

2.2 Trabalhos Relacionados

Newman (2001) analisou a estrutura das redes de colaboração científica buscando entender como os cientistas se conectam e formam comunidades. Os nós da rede são os autores e as ligações entre eles se eles publicaram algum artigo juntos. Os dados foram retirados de bases de dados como a *MEDLINE*, *Los Alamos e-Print Archive* e *NCSTRL*. Alguns pontos observados foram que as redes formam *small worlds*, onde um par de cientistas escolhidos aleatoriamente são normalmente separados por um curto caminho de conhecidos entre eles e também a presença de um *giant component* que conecta a maioria dos pesquisadores de uma área. Esta monografia busca seguir este caminho de pesquisa aplicando técnicas modernas de PNL a rede de professores de PPGs em computação no Brasil.

Seguindo a linha de pesquisa de Newman (2001), Clauset, Arbesman e Larremore (2015) analisaram a rede de contratação de docentes nos Estados Unidos e Canadá das áreas de Ciência da Computação, Negócios e História. Este estudo revelou uma forte e assimétrica hierarquia de prestígio, onde uma minoria das instituições forma a maioria dos docentes, mostrando que o prestígio da instituição de doutoramento é um ótimo preditor do destino de um pesquisador. Já em Lee, Clauset e Larremore (2021) buscou-se entender os mecanismos dinâmicos que geram e mantêm essas hierarquias. O estudo mostrou que as hierarquias de prestígio geralmente são estáveis e auto-reforçadas, que é muito difícil as instituições que estão no topo saírem e as que não estão chegarem lá. De forma análoga, a presente monografia busca analisar a rede científica nacional de computação através da estrutura hierárquica dada pelas notas da CAPES, seguindo uma metodologia diferente, baseada na análise do agrupamento dos *embeddings* dos pesquisadores.

Acuna, Allesina e Kording (2012) utilizaram aprendizado de máquina em um conjunto de dados de mais de 3000 neurocientistas para tentar predizer o sucesso de um pesquisador. Eles desenvolveram um modelo de regressão linear para prever o índice h de um pesquisador dentro de 1, 5 e 10 anos. Algumas das variáveis preditivas principais identificadas pelo estudo são o índice h atual, o total de artigos, o tempo de carreira, o número de periódicos distintos em que publicou e o número de artigos publicados em periódicos considerados de elite. Já em Sinatra et al. (2016) foi feita a análise da carreira de milhares de cientistas e descobriram que o trabalho de maior impacto de um cientista pode ocorrer em qualquer momento da sua carreira. Para isso foi proposto um modelo Q, demonstrando ser um preditor robusto de sucesso para um cientista. O modelo Q sugere que a habilidade de produzir trabalhos de alto impacto é uma característica intrínseca e duradoura de um pesquisador, embora a sorte desempenhe um papel

em cada publicação individual. O presente estudo busca relacionar métricas de sucesso dos pesquisadores de PPGs em computação no Brasil com características extraídas do conteúdo semântico de suas publicações.

Ganguly e Pudi (2017) buscaram criar *embeddings* para representar artigos científicos a partir de uma abordagem unificada entre o aprendizado através dos dados textuais e da rede de citações utilizando redes neurais. Esta abordagem é baseada no *Word2Vec* e busca, dado um artigo, prever os artigos em seu contexto, onde temos o contexto textual que são artigos textualmente similares e contexto da rede que são artigos que citam ou são citados. A presente monografia busca primeiramente gerar os *embeddings* a partir dos dados textuais dos autores, mais especificamente dos seus tópicos de pesquisa e depois analisar as relações entre eles na rede.

Já em Murray et al. (2020), foi proposta a utilização do algoritmo de *embedding word2vec* para estudar padrões de mobilidade em três bases de dados diferentes sendo uma base de dados de itinerários de voos nos Estados Unidos, outra de reservas de hospedagem na Coreia do Sul e por fim a de maior relevância para este trabalho, a de mobilidade de afiliações científicas retirada da base de dados *Web of Science*. Os autores mostram que os *embeddings* conseguem capturar relações linguísticas, culturais e hierárquicas na mobilidade e que fornecem uma medida de distância melhor que a distância geográfica. Mais especificamente nos dados científicos, os *embeddings* conseguiram capturar dados como regiões geográficas, hierarquias de prestígio e o compartilhamento do idioma. Por fim, os autores concluem que essa abordagem pode ser utilizada para diversos domínios de mobilidade e medição de distâncias e recomenda para trabalhos futuros, a geração dos *embeddings* com a técnica BERT (DEVLIN et al., 2018), que é o modelo que é utilizado nesta monografia para gerar os *embeddings* dos indivíduos afiliados aos PPGs acadêmicos de computação no Brasil.

Por fim, em Szluka, Csajbók e Győrffy (2023), é analisada a relação entre os indicadores bibliométricos e as classificações universitárias das 300 primeiras universidades colocadas em quatro *rankings* internacionais. Os autores mostram que as variáveis bibliométricas que tem alto impacto podem variar em diferentes *rankings*, por exemplo, universidades com boas colocações em determinados *rankings* tem maior atividade em publicações em áreas específicas. Outra característica observada foi a importância do número de publicações e de publicações de alto impacto em determinados *rankings*. Traçando um paralelo, a presente monografia também busca encontrar relações entre a pesquisa científica e as notas dadas pelas CAPES aos PPGs de Ciência da Computação no Brasil.

3 Desenvolvimento

3.1 Base de Dados

Nesta monografia foi desenvolvido um conjunto de dados composto pelos docentes credenciados a PPGs acadêmicos em computação no Brasil, contendo informações sobre cada docente e os respectivos programas em que eles estão credenciados. Esses dados foram extraídos e combinados de três fontes distintas: OpenAlex (PRIEM; PIWOWAR; ORR, 2022), Plataforma Lattes [1] de 2023 a 2024 e relatório de avaliação quadrienal de 2017 a 2020 produzido pela CAPES [2].

O desenvolvimento da base de dados foi realizado por discentes do curso de ciência da computação e do PPG em Ciência da Computação da UFOP (Universidade Federal de Ouro Preto) e dois docentes orientadores, todos integrantes do CSILab (Laboratório de Computação de Sistemas Inteligentes) do DECOM (Departamento de Computação) da UFOP.

3.1.1 Extração de dados

Inicialmente, os dados dos PPGs e dos seus docentes foram retirados do relatório de avaliação quadrienal de 2017 a 2020 da CAPES dos programas de Ciência da Computação. Aqui foram utilizados somente os dados dos PPGs acadêmicos. Foi possível obter dados como código do PPG, nome do PPG, nome da instituição e nota do PPG. Cada programa recebe no relatório uma nota de 3 a 7, onde quanto mais alta a nota, melhor. Para receber notas 6 e 7 os programas devem atender a critérios bem específicos e são programas de excelência internacional, enquanto os de nota 4 e 5 são programas de excelência nacional e os PPGs de nota 3 na maioria são programas mais novos e com pouca maturidade. Além disso, da avaliação da CAPES também foi obtida a lista dos professores vinculados a cada PPG e seus dados como nome, ano do doutorado, regime de trabalho e carga horária.

Em posse desses dados, para cada professor presente na lista foi baixado o currículo dele na Plataforma Lattes manualmente em formato XML (Extensible Markup Language) e anotado se o docente era bolsista de produtivade ou não. Caso verdadeiro era anotado o extrato da bolsa de produtividade. A Plataforma Lattes é uma base de dados pública de currículos de pesquisadores, grupos de pesquisa e instituições disponibilizada e mantida pelo CNPq. Com isso, foi utilizado um *script* auxiliar escrito em linguagem de programação Python para retirar as informações de interesse dos arquivos XML, como a URL (Uniform Resource Locator) do currículo lattes do docente, a instituição em que ele realizou o doutorado, o seu orientador de doutorado e o seu

[1] <https://lattes.cnpq.br/>
[2] <https://www.gov.br/capes/pt-br/acesso-a-informacao/acoes-e-programas/avaliacao/avaliacao-quadrienal>

ORCID (Open Researcher and Contributor ID) quando possível, que é um identificador único para cientistas.

Por fim, para cada docente foi realizada manualmente a extração dos dados de identifação na OpenAlex com auxílio da API fornecida pela plataforma [3]. Foram obtidos os IDs (Identity) únicos da OpenAlex para o autor, seu supervisor de doutorado e a instituição em que o mesmo cursou o doutorado.

3.1.2 Descrição dos dados

A base de dados construída neste projeto atualmente contém as seguintes variáveis para cada docente:

- **Código do PPG:** É um identificador único composto de letras e números para cada PPG no Brasil. No conjunto de dados indica o código do PPG que o autor está vinculado;
- **Nome do PPG:** É uma sigla indicando o nome do PPG que o docente está vinculado, por exemplo: I (Informática), CC (Ciência da Computação), ESC (Engenharia de Sistemas e Computação), CCMC (Ciências da Computação e Matemática Computacional), etc;
- **Nota do PPG:** Nota atribuída ao PPG pelo relatório de avaliação quadrienal de 2021 da CAPES, esta nota pode variar de 3 a 7;
- **ID da instituição:** É um identificador único para cada instituição na OpenAlex. Trata-se do ID da instituição em que o professor faz parte do PPG;
- **Sigla da instituição:** É a sigla adotada por cada instituição para se referir a ela mesma. No conjunto de dados é a sigla da instituição em que o autor está vinculado;
- **Nome do docente:** Nome do docente em letras maíusculas;
- **Ano do doutorado:** Ano em que o autor concluiu o seu doutorado;
- **Regime de trabalho:** O regime de trabalho que o professor tem na instituição em que está vinculado, podendo ser integral ou dedicação exclusiva;
- **Carga horária:** Um valor inteiro representando a carga horária de trabalho do docente em sua instituição;
- **Link do Lattes:** Link do currículo lattes do pesquisador;
- **ID do autor:** Identificador único do docente na OpenAlex;
- **Bolsista de produtividade:** Pode assumir o valor verdadeiro ou falso e indica se o pesquisador possui ou não bolsa de produtividade do CNPq;

[3] <https://docs.openalex.org/>

- **Extrato da bolsa de produtividade:** Indica o tipo da bolsa de produtividade do autor, dentre as seguintes possibilidades: PQ2, PQ1D, PQ1C, PQ1B, PQ1A, DT2, DT1D, DT1C, DT1B e DT1A. Caso o pesquisador não possua bolsa de produtividade este campo recebe o valor nulo;
- **ID da instituição de doutorado:** É o identificador único na OpenAlex da instituição que o docente cursou seu doutorado;
- **Nome da instituição de doutorado:** É o nome da instituição onde o pesquisador concluiu seu doutorado;
- **Código do PPG de doutorado:** É um identificador único composto de letras e números para cada PPG no Brasil, portanto, caso o pesquisador tenha realizado seu doutorado em uma instituição fora do Brasil, esse campo recebe o valor nulo;
- **ID do orientador de doutorado:** É o identificador único na OpenAlex do supervisor de doutorado do docente;
- **Nome do orientador de doutorado:** É o nome do orientador de doutorado do docente.

3.1.3 Enriquecimento dos dados

A base de dados descrita anteriormente foi enriquecida com os dados retirados da OpenAlex (PRIEM; PIWOWAR; ORR, 2022). Com o ID da OpenAlex de cada docente, a fase de enriquecimento foi automatizada. Um *script* desenvolvido em linguagem de programação Python, utilizando a biblioteca *pyalex* [4], foi executado para consultar a API da OpenAlex para cada professor presente na base de dados a fim de coletar os seguintes dados:

- **Métricas de Produtividade:** *works_count* (número total de trabalhos) e *cited_by_count* (número total de citações recebidas);
- **Métricas de impacto:** *h_index* (índice h) e *i10_index* (índice i10);
- **Perfil Semântico:** O campo *x_concepts*, que é uma lista de conceitos associados ao pesquisador, juntamente com um *score* que indica a força dessa associação. Este campo foi utilizado para gerar os *embeddings*.

Ao final desta etapa, foi consolidada uma única base de dados contendo informações demográficas e de afiliação da CAPES/Lattes, e as métricas de produtividade e impacto e perfil semântico da OpenAlex de cada pesquisador.

[4] <https://github.com/J535D165/pyalex>

3.2 Análise e Modelagem de Dados

Com a base de dados consolidada, foi desenvolvido um *pipeline* computacional utilizando a linguagem de programação *Python* para processar, modelar e analisar os dados dos pesquisadores. O objetivo deste *pipeline* era transformar os dados textuais associados a cada pesquisador em *embeddings* e, a partir delas, agrupar os dados de forma não supervisionada para posterior análise quantitativa. O processo é dividido em três etapas principais: engenharia de *features*, aplicação de algoritmo não supervisionado e análise comparativa.

3.2.1 Engenharia de *features*

A primeira etapa consistiu em criar uma representação semântica para cada pesquisador a partir dos conceitos retirados da OpenAlex, como se segue:

- **Criação do Documento do Autor (*author_document*):** Para cada pesquisador, foi gerada uma variável textual. Esta variável é uma concatenação dos seus principais tópicos de pesquisa, retirados do campo *x_concepts* da OpenAlex. O nome de cada conceito foi repetido um número de vezes proporcional ao seu *score* de associação da OpenAlex, criando um texto ponderado que reflete a especialidade de cada pesquisador.
- **Geração dos Embeddings:** As variáveis textuais geradas foram convertidas em *embeddings* utilizando o modelo SciBERT. A escolha do SciBERT se deu por ser uma variante do modelo BERT pré-treinada em uma base de dados de textos científicos, o que o torna mais apto a capturar as nuances semânticas do vocabulário acadêmico em comparação com modelos de linguagem de domínio geral. O resultado deste processo foi um vetor de 768 dimensões para cada pesquisador representando numericamente seu perfil temático de pesquisa.

3.2.2 Redução da Dimensionalidade e *Clusterização*

Os *embeddings* de 768 dimensões são muito densos para visualização e podem apresentar desafios para algoritmos de *clusterização*. Portanto, aplicamos técnica de redução da dimensionalidade e posteriormente buscamos identificar os grupos.

- **Redução da Dimensionalidade:** Foi utilizado o algoritmo UMAP para projetar os embeddings de 768 dimensões em um espaço bidimensional (2D). A escolha do UMAP se deu por sua capacidade de preservar tanto a estrutura topológica local quanto global dos dados. Esta etapa é crucial para visualização dos dados e para otimizar o desempenho do algoritmo de *clusterização*. O UMAP foi executado com os seguintes parâmetros: *n_neighbors* = 15; *min_dist* = 0.1 e *n_components* = 2.

- **Clusterização:** Sobre os *embeddings* normalizados de alta dimensão aplicamos o algoritmo HDBSCAN para identificar as comunidades temáticas. O HDBSCAN apresenta a vantagem de não exigir predefinição do número de *clusters* e de ser capaz de identificar *outliers* (ruídos). O HDBSCAN foi executado com os seguintes parâmetros: *min_cluster_size* = 15; *min_samples* = 5 e *metric* = *euclidean*.

3.2.3 Caracterização e Análise dos *Clusters*

Depois de clusterizar os dados, a etapa final consistiu em caracterizar os *clusters* gerados e realizar uma análise comparativa para responder às questões de pesquisa secundárias.

- **Caracterização Temática:** Para atribuir um significado semântico a cada *cluster*, foi agregado os *author_documents* de todos pesquisadores pertencentes a um mesmo grupo. Em seguida, foi extraído os cinco conceitos mais frequentes deste texto agregado para criar rótulos descritivos.
- **Análise Comparativa:** Com os clusters devidamente rotulados, foram investigadas as diferenças relevantes entre eles, focando a análise em três eixos de acordo com as QPSs:
  - **Perfil de Fomento**: Foi calculada a proporção de pesquisadores com bolsa de produtividade CNPq em cada *cluster*.
  - **Perfil de Impacto e Produtividade**: Foi analisada a distribuição das métricas bibliométricas, quantidade de trabalhos publicados (*works_count*), quantidade de citações recebidas (*cited_by_count*), índice h (*h_index*) e índice i10 (*i10_index*) para cada cluster, utilizando as medianas como medida de tendência central e *box plots* para visualização da dispersão.
  - **Diversidade Temática e Excelência:** Foi investigada a relação entre a diversidade de perfis de pesquisa e a excelência dos PPGs em computação, medindo o número de *clusters* temáticos distintos presentes nos programas com notas 6 e 7 na avaliação da CAPES.

4 Resultados

A aplicação do pipeline metodológico descrito no Capítulo 3 sobre a base de dados de 1511 pesquisadores válidos permitiu a extração de resultados quantitativos e qualitativos sobre os PPGs em computação no Brasil. Esta seção apresenta e discute esses resultados, organizando-os de acordo com as questões de pesquisa propostas.

4.1 Estrutura Temática da Comunidade de Pesquisadores (QPS1)

A primeira questão de pesquisa (QPS1) buscava identificar e caracterizar comunidades temáticas de forma não supervisionada. A aplicação do HDBSCAN resultou na identificação de 6 *clusters* temáticos principais e um grupo de 47 professores classificados como outliers. A caracterização semântica de cada cluster, baseada nos conceitos mais frequentes de seus membros, revelou a seguinte estrutura:

- **Cluster 4: Computer, Science, Engineering, Mathematics, System:** Esté é o maior *cluster* encontrado, e representa a base da pesquisa em computação no Brasil integrando sistemas, engenharia e matemática.
- **Cluster 5: Mathematics, Computer, Science, Graph, Combinatorics:** Um grupo com forte inclinação para problemas relacionados a grafos e otimização combinatória.
- **Cluster 0: Engineering, Physics, Science, Mechanics, Quantum:** Representa uma interface da computação com as engenharias e a física, possivelmente incluindo áreas como sistemas embarcados e computação quântica.
- **Cluster 3: Biology, Medicine, Science, Chemistry, Computer:** Um *cluster* interdisciplinar, focado na intersecção da computação com as ciências da vida, abrangendo áreas como bioinformática e informática médica.
- **Cluster 2: Science, Philosophy, Computer, Art, Humanities:** Este grupo representa a conexão da computação com as ciências humanas e artes, possivelmente abrangendo áreas como interação humano-computador, arte computacional e os aspectos filosóficos da ciência da computação.
- **Cluster 1: (Sem conceitos suficientes):** Um pequeno grupo de pesquisadores para os quais a OpenAlex não dispunha de conceitos suficientes para uma caracterização robusta.

- **Outliers e Interdisciplinares:** Este grupo contém 47 pesquisadores cujos perfis temáticos não se alinham densamente com nenhum dos *clusters* principais, sugerindo atuação em nichos de pesquisa únicos ou um alto grau de interdisciplinaridade.

A projeção UMAP (Figura 4.1) ilustra visualmente essa estrutura, mostrando o Cluster 4 como grande *cluster* central e os outros *clusters* menores e mais periféricos, o que reforça a ideia de um núcleo de pesquisa principal cercado por áreas mais especializadas ou interdisciplinares.

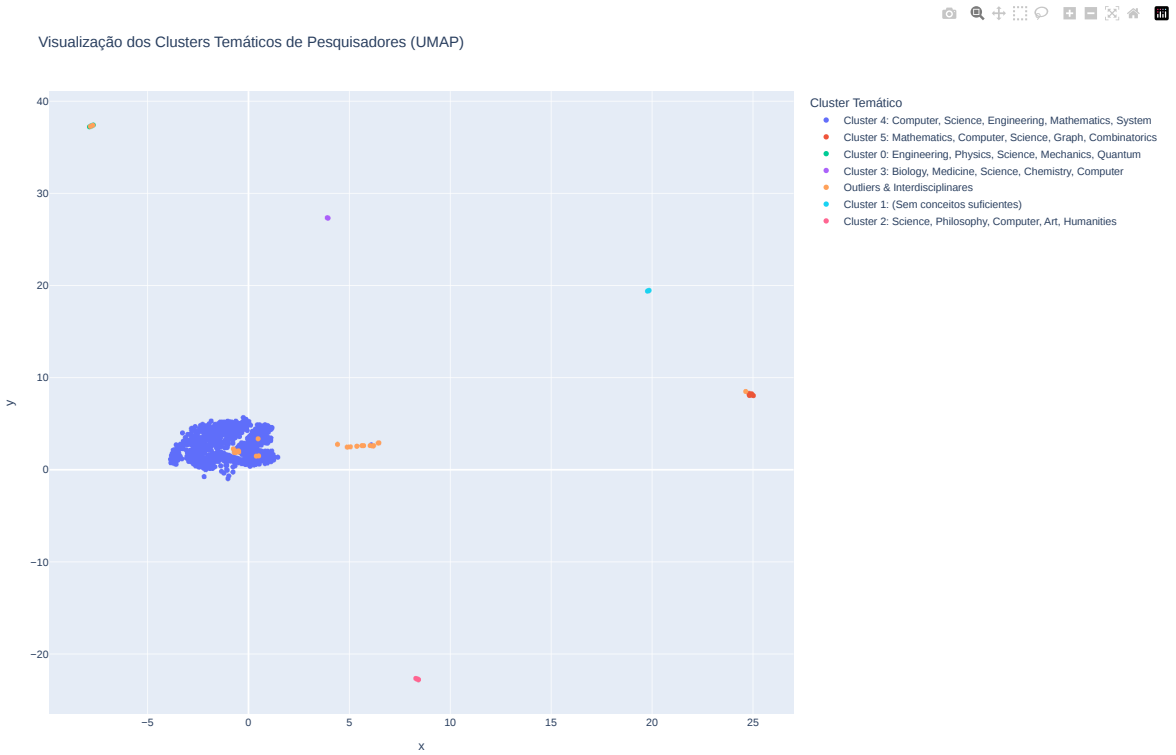

Figura 4.1 – Visualização dos *clusters* temáticos de pesquisadores projetados em 2D com UMAP.

4.2 Perfis de Produtividade e Impacto por Comunidade Temática (QPS2)

A segunda questão de pesquisa (QPS2) buscava entender se os perfis de produtividade e impacto acadêmico variavam entre as comunidades identificadas. A análise comparativa revelou diferenças significativas, confirmando a hipótese.

- **Fomento à Pesquisa (Bolsas de Produtividade):** A proporção de bolsistas de produtividade do CNPq não é uniforme entre os *clusters*. O Cluster 5 apresenta a maior proporção de bolsistas (50,0%), seguido pelo Cluster 3 com 43,8%. Em contraste, os grupos Outliers e Interdisciplinares com 14,9% e o Cluster 2 com 13,3% exibem menores taxas.

- **Métricas de Impacto e Produtividade:** A análise das distribuições das métricas bibliométricas (Figuras 4.2, 4.3, 4.4 e 4.5) revela perfis distintos para cada cluster.
  - **Contagem de Trabalhos (works_count):** O Cluster 4 se destaca pela maior mediana de publicações (92.0), indicando um alto volume de produção científica.
  - **Contagem de Citações (cited_by_count):** O Cluster 3 apresenta uma mediana de citações (1254.0) drasticamente superior aos outros grupos, incluindo o Cluster 4 (653.5)
  - **Índice H e Índice i10:** Seguindo a tendência das citações, o Cluster 3 também lidera com as maiores medianas de índice h (16.5) e índice i10 (24.5).

  Esses resultados mostram que, embora o núcleo da computação representado pelo Cluster 4 tenha maior número de publicações, a pesquisa na interface com ciências da vida (Cluster 3) alcança um impacto substancialmente maior em termos de citações e índices h e i10. Isso evidencia a relevância e a visibilidade da pesquisa interdisciplinar.

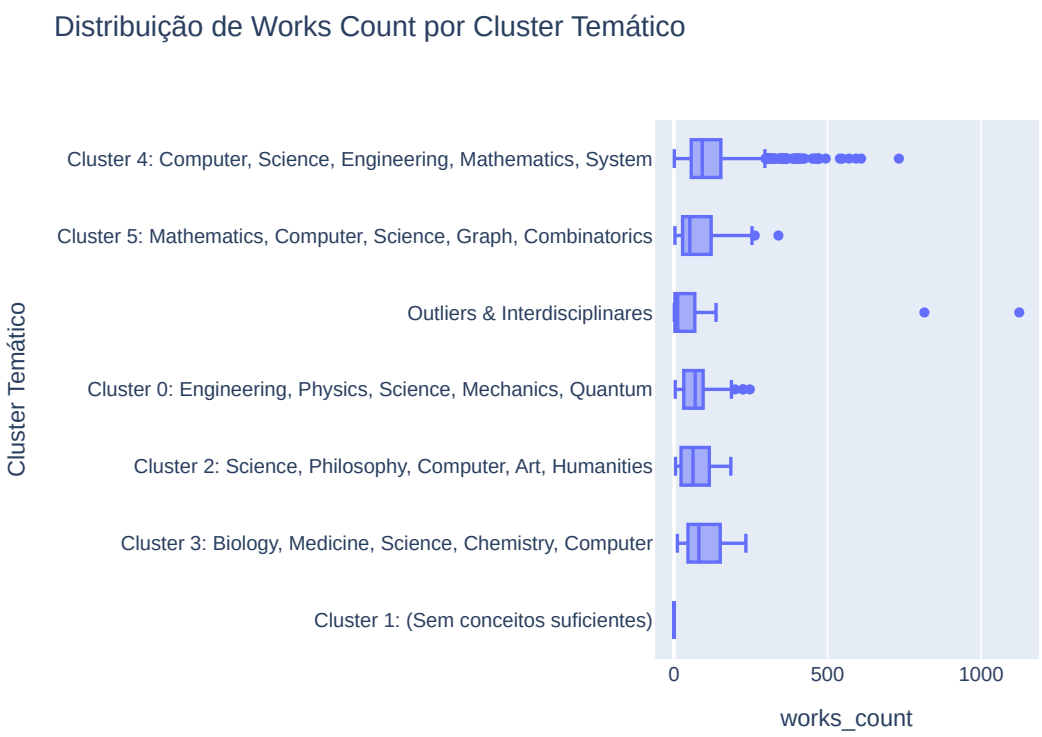

Figura 4.2 – Distribuição de Works Count (número de trabalhos publicados) por Cluster Temático.

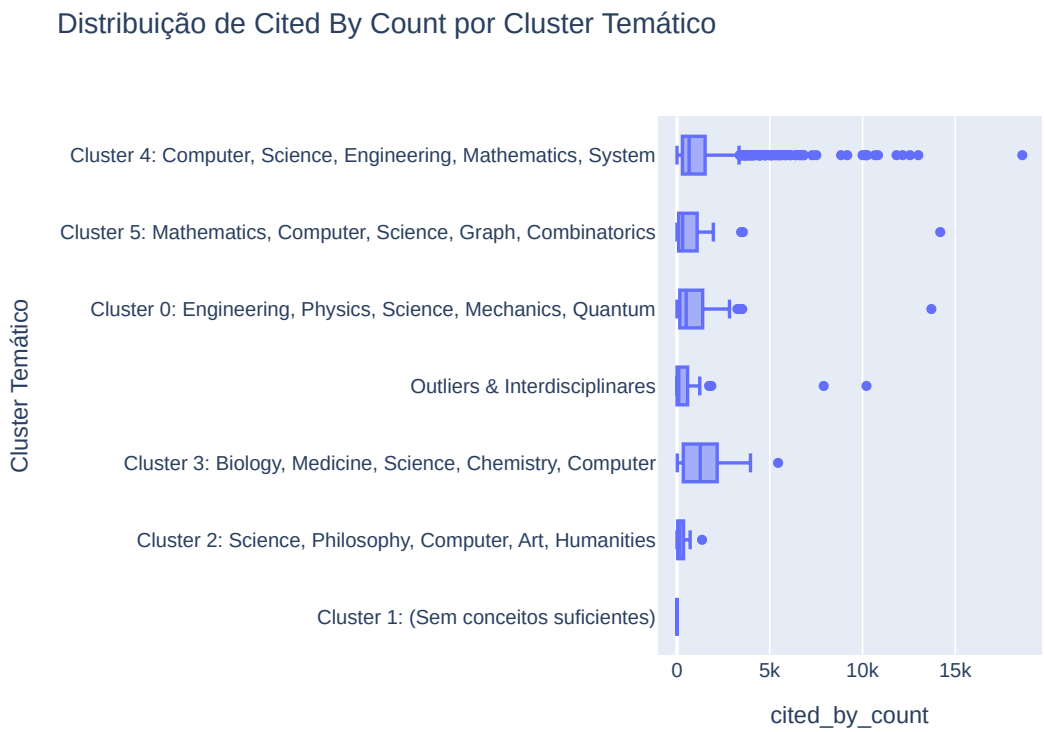

Figura 4.3 – Distribuição de Cited By Count (número de citações) por Cluster Temático.

4.3 Diversidade Temática e Excelência dos Programas (QPS3)

Por fim, a QPS3 buscava entender a correlação entre a excelência de um PPG em computação e a diversidade temática de seus pesquisadores. A análise dos PPGs com notas 6 e 7 da CAPES revelou uma forte evidência nesse sentido.

A UFMG, um programa de nota 7, demonstrou ser o mais diverso tematicamente, com professores representantes de todos os 6 clusters identificados, além do grupo de outliers. Outros programas de excelência, como UFRGS (nota 7) e USP (nota 7), também apresentaram alta diversidade, com 6 e 5 grupos temáticos respectivamente.

O detalhamento da diversidade na UFMG mostra que, embora a maioria dos seus professores estejam alocados no Cluster 4, o programa conta com pesquisadores de todas as outras áreas temáticas. Este resultado indica que PPGs de ponta em computação no Brasil se caracterizam por terem um núcleo forte e consolidado nas áreas centrais da computação, ao mesmo tempo que fomentam um ecossistema de pesquisa diversificado e plural.

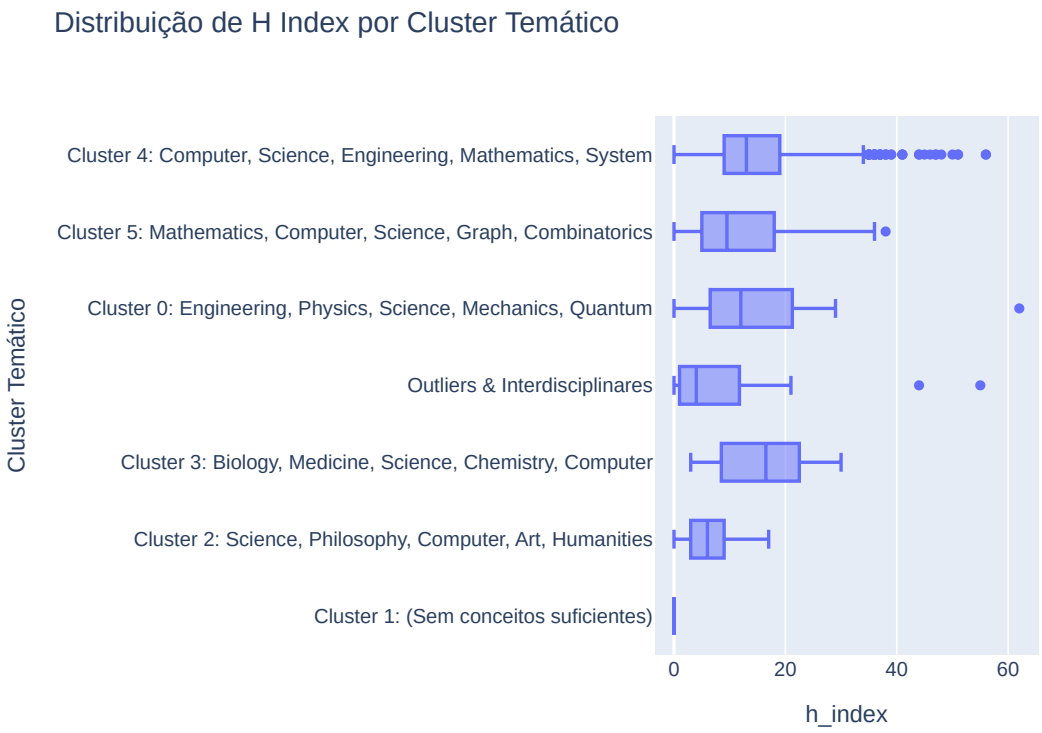

Figura 4.4 – Distribuição de h-index (índice h) por Cluster Temático.

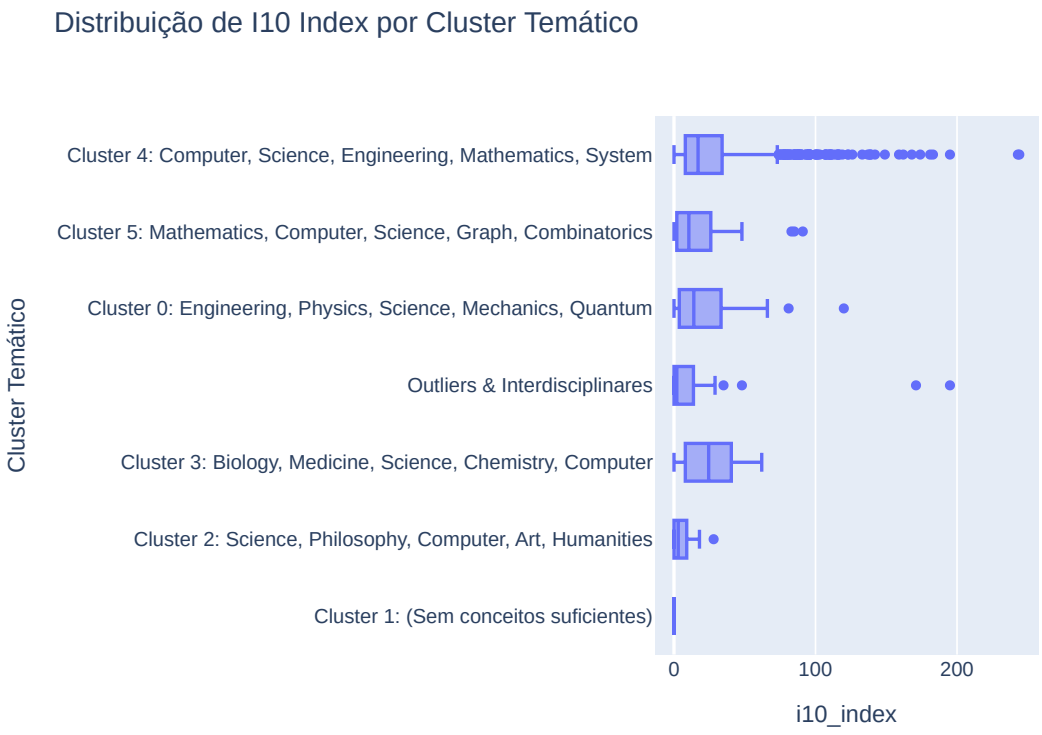

Figura 4.5 – Distribuição de i10-index (índice i10) por Cluster Temático.

5 Conclusão

Esta monografia se propôs a responder à seguinte questão de pesquisa primária (QPP): "Técnicas de Processamento de Linguagem Natural como a geração de *embeddings* juntamente com técnicas de aprendizado não supervisionado como a *clusterização* sobre dados de produção científica pode revelar a estrutura temática e os perfis de impacto da comunidade de professores credenciados a Programas de Pós-Graduação em computação no Brasil?". Os resultados apresentados e discutidos no Capítulo 4 permitem responder a essa questão de forma afirmativa.

A abordagem integrada, utilizando SciBERT para criação de *embeddings* semânticos e HDBSCAN para a *clusterização*, foi capaz de mapear uma imagem da pesquisa em computação no Brasil, revelando uma estrutura composta por um núcleo da computação e outros grupos temáticos especializados e interdisciplinares. Além disso, as análises quantitativa e comparativa demonstraram que essas comunidades possuem perfis de impacto, produtividade e fomento distintos e estatisticamente relevantes.

Os principais achados desta monografia são:

- **Estrutura da Comunidade:** Foram identificadas 6 comunidades temáticas distintas variando desde a matemática, grafos e otimização (Cluster 5), até intersecções com a física (Cluster 0), ciências da vida (Cluster 3) e ciências humanas (Cluster 2).
- **Impacto da Interdisciplinaridade:** A pesquisa na interface da computação com as ciências da vida (Cluster 3) demonstrou ter o maior impacto em termos de citações, índice h e índice i10, enquanto a base da computação (Cluster 4) possui maior volume de publicações.
- **Diversidade como Indicador de Excelência:** Há uma forte correlação positiva entre a excelência de um PPG em computação medida pela nota da CAPES e sua diversidade temática, tomando como exemplo a UFMG.

As implicações desses achados são relevantes para gestores de ciência e tecnologia, agências de fomento à pesquisa e para a comunidade acadêmica. Eles mostram a importância de valorizar e incentivar a pesquisa de forma interdisciplinar, que demonstra alto potencial de impacto. Adicionalmente, sugerem que a promoção da diversidade temática pode ser uma estratégia chave para o fortalecimento e a busca pela excelência nos PPGs em computação no Brasil.

Em suma, este trabalho demonstra o poder da aplicação de técnicas de PLN e aprendizado não supervisionado para o campo de SciSci, oferecendo um mapa detalhado e multifacetado da pesquisa em computação no Brasil.

5.1 Trabalhos Futuros

Apesar de bons resultados, este estudo possui limitações. A representação dos pesquisadores foi baseada nos conceitos fornecidos pela OpenAlex, que, embora eficaz, é uma abstração do trabalho completo de um autor. Além disso, a análise é um retrato estático, baseado na avaliação quadrienal de 2017 a 2020. Também, as universidades que possuem mais de um PPG tiveram seus PPGs agregados, onde cada universidade representa um programa.

Como trabalhos futuros:

- **Análise Dinâmica:** Realizar uma análise ao longo do tempo para observar como os pesquisadores transitam entre *clusters* temáticos ao longo de suas carreiras e como os próprios *clusters* evoluem com o tempo.
- **Redes de Colaboração:** Integrar dados de coautoria para analisar a rede de colaboração entre os diferentes *clusters* temáticos, investigando se a colaboração entre *clusters* gera maior impacto.
- **Modelagem Preditiva:** Desenvolver modelos de aprendizado de máquina para predizer o futuro impacto de um pesquisador júnior com base em sua trajetória inicial.
- **Refinamento da Representação:** Aprimorar a variável textual de cada autor (*author_document*) incorporando informações dos títulos e resumos das suas publicações, utilizando técnicas mais avançadas de sumarização textual.

REFERÊNCIAS

ACUNA, D. E.; ALLESINA, S.; KORDING, K. P. Predicting scientific success. *Nature*, Nature Publishing Group UK London, v. 489, n. 7415, p. 201–202, 2012.

ALSENTZER, E.; MURPHY, J. R.; BOAG, W.; WENG, W.-H.; JIN, D.; NAUMANN, T.; MCDERMOTT, M. Publicly available clinical bert embeddings. *arXiv preprint arXiv:1904.03323*, 2019.

BELTAGY, I.; LO, K.; COHAN, A. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*, 2019.

CAMPELLO, R. J.; MOULAVI, D.; SANDER, J. Density-based clustering based on hierarchical density estimates. In: SPRINGER. *Pacific-Asia conference on knowledge discovery and data mining*. [S.l.], 2013. p. 160–172.

CLAUSET, A.; ARBESMAN, S.; LARREMORE, D. B. Systematic inequality and hierarchy in faculty hiring networks. *Science advances*, American Association for the Advancement of Science, v. 1, n. 1, p. e1400005, 2015.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

FONSECA, C. *Word Embedding: fazendo o computador entender o significado das palavras*. 2021. <https://medium.com/turing-talks/word-embedding-fazendo-o-computador-entender-o-significado-das-palavras-92fe22745057>. Acessado: 15/02/2024.

FORTUNATO, S.; BERGSTROM, C. T.; BÖRNER, K.; EVANS, J. A.; HELBING, D.; MILOJEVIĆ, S.; PETERSEN, A. M.; RADICCHI, F.; SINATRA, R.; UZZI, B. et al. Science of science. *Science*, American Association for the Advancement of Science, v. 359, n. 6379, p. eaao0185, 2018.

GANGULY, S.; PUDI, V. Paper2vec: Combining graph and text information for scientific paper representation. In: SPRINGER. *European conference on information retrieval*. [S.l.], 2017. p. 383–395.

GUILARDUCCI, A. F.; VASCONCELOS, I. L. L.; LUZ, E. J. da S.; FREITAS, V. L. de S. Institutional hierarchy and asymmetry in brazilian computer science faculty hiring network. In: SBC. *Brazilian Workshop on Social Network Analysis and Mining (BraSNAM)*. [S.l.], 2025. p. 148–158.

KHATTAK, F. K.; JEBLEE, S.; POU-PROM, C.; ABDALLA, M.; MEANEY, C.; RUDZICZ, F. A survey of word embeddings for clinical text. *Journal of Biomedical Informatics*, Elsevier, v. 100, p. 100057, 2019.

LEE, E.; CLAUSET, A.; LARREMORE, D. B. The dynamics of faculty hiring networks. *EPJ Data Science*, Springer Berlin Heidelberg, v. 10, n. 1, p. 48, 2021.

MCINNES, L.; HEALY, J.; MELVILLE, J. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

MIKOLOV, T.; CHEN, K.; CORRADO, G.; DEAN, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

MIKOLOV, T.; SUTSKEVER, I.; CHEN, K.; CORRADO, G. S.; DEAN, J. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, v. 26, 2013.

MURRAY, D.; YOON, J.; KOJAKU, S.; COSTAS, R.; JUNG, W.-S.; MILOJEVIĆ, S.; AHN, Y.-Y. Unsupervised embedding of trajectories captures the latent structure of mobility. *arXiv preprint arXiv:2012.02785*, 2020.

NEWMAN, M. E. The structure of scientific collaboration networks. *Proceedings of the national academy of sciences*, The National Academy of Sciences, v. 98, n. 2, p. 404–409, 2001.

PRIEM, J.; PIWOWAR, H.; ORR, R. Openalex: A fully-open index of scholarly works, authors, venues, institutions, and concepts. *arXiv preprint arXiv:2205.01833*, 2022.

RADFORD, A.; NARASIMHAN, K.; SALIMANS, T.; SUTSKEVER, I. et al. Improving language understanding by generative pre-training. OpenAI, 2018.

SAUL, G. H. Uma aplicação do modelo de processamento de linguagem natural bert para classificação de notícias falsas. 2023.

SINATRA, R.; WANG, D.; DEVILLE, P.; SONG, C.; BARABÁSI, A.-L. Quantifying the evolution of individual scientific impact. *Science*, American Association for the Advancement of Science, v. 354, n. 6312, p. aaf5239, 2016.

SZLUKA, P.; CSAJBÓK, E.; GYŐRFFY, B. Relationship between bibliometric indicators and university ranking positions. *Scientific Reports*, Nature Publishing Group UK London, v. 13, n. 1, p. 14193, 2023.

VASCONCELOS, I. L. L.; GUILARDUCCI, A. F.; GERTRUDES, J. C.; MOREIRA, G. J. P.; FREITAS, V. L. de S.; LUZ, E. J. da S. Uncovering collaboration patterns in brazilian computer science graduate programs through network embeddings. In: SBC. *Brazilian Workshop on Social Network Analysis and Mining (BraSNAM)*. [S.l.], 2025. p. 106–119.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, Ł.; POLOSUKHIN, I. Attention is all you need. *Advances in neural information processing systems*, v. 30, 2017.

ZHANG, A.; LIPTON, Z. C.; LI, M.; SMOLA, A. J. Dive into deep learning. *arXiv preprint arXiv:2106.11342*, 2021.

ZHANG, A.; LIPTON, Z. C.; LI, M.; SMOLA, A. J. *Dive into Deep Learning*. [S.l.]: Cambridge University Press, 2023. <https://D2L.ai>.