



UNIVERSIDADE FEDERAL DE PERNAMBUCO

CENTRO DE INFORMÁTICA

CURSO SISTEMAS DE INFORMAÇÃO

LUIZ GUYLHERME AVELINO RODRIGUES

**PLANEJAMENTO COLABORATIVO DOCENTE: UM SISTEMA DE
RECOMENDAÇÃO PARA FOMENTAR CONEXÕES ENTRE PROFESSORES**

Recife

2025

LUIZ GUYLHERME AVELINO RODRIGUES

**PLANEJAMENTO COLABORATIVO DOCENTE: UM SISTEMA DE
RECOMENDAÇÃO PARA FOMENTAR CONEXÕES ENTRE PROFESSORES**

Trabalho de conclusão de cursos
apresentado ao Programa de Graduação
do curso de Sistemas de Informação da
Universidade Federal de Pernambuco.

Área de concentração: Interação
Humano-Computador

Orientador (a): Alex Sandro Gomes

Coorientador (a): Carlos José Pereira da Silva

RECIFE-PE

2025

Ficha de identificação da obra elaborada pelo autor,
através do programa de geração automática do SIB/UFPE

Rodrigues, Luiz Gylherme Avelino.

Planejamento colaborativo docente: um sistema de recomendação para
fomentar conexões entre professores / Luiz Gylherme Avelino Rodrigues. -
Recife, 2025.

38 p. : il., tab.

Orientador(a): Alex Sandro Gomes

Coorientador(a): Carlos José Pereira da Silva

Trabalho de Conclusão de Curso (Graduação) - Universidade Federal de
Pernambuco, Centro de Informática, Sistemas de Informação - Bacharelado,
2025.

Inclui referências.

1. planejamento colaborativo de aulas. 2. sistemas de recomendação. 3.
aprendizado colaborativo. 4. prática docente. 5. planejamento de aulas. I.
Gomes, Alex Sandro . (Orientação). II. Silva, Carlos José Pereira da .
(Coorientação). IV. Título.

000 CDD (22.ed.)

Luiz Guylherme Avelino Rodrigues

**PLANEJAMENTO COLABORATIVO DOCENTE: UM SISTEMA DE
RECOMENDAÇÃO PARA FOMENTAR CONEXÕES ENTRE PROFESSORES**

Trabalho de Conclusão de Curso
apresentado ao Curso de Graduação em
Sistemas de Informação da Universidade
Federal de Pernambuco, como requisito
parcial para obtenção do título de
bacharel em Sistemas de Informação.

Aprovado em: 04/04/2025

BANCA EXAMINADORA

Prof. Dr. Alex Sandro Gomes(Orientador)

Universidade Federal de Pernambuco

Prof. Dr. Leandro Maciel Almeida(Presidente e Examinador Interno)

Universidade Federal de Pernambuco

Dedico este trabalho à minha mãe, e professora Aldeni Avelino da Silva Rodrigues, e para o meu pai, Ricardo de Moraes Rodrigues.

RESUMO

Este trabalho propõe desenvolver um sistema de recomendação baseado em técnicas de inteligência artificial (IA) para aprimorar a colaboração entre professores na plataforma Inovaula.com. A pesquisa visa integrar técnicas avançadas de IA para personalizar conexões entre docentes, facilitando o planejamento colaborativo de aulas. O sistema utiliza a geração de dados sintéticos para simular cenários educacionais e validar o modelo de recomendação. O algoritmo de recomendação proposto, HNSW-DSM, foi desenvolvido para equilibrar a relevância pedagógica e a diversidade nas sugestões, além de lidar com o desafio do “cold start” em sistemas de recomendação. Os resultados evidenciaram que o sistema conseguiu gerar recomendações personalizadas, mesmo para novos usuários, e oferecer um desempenho eficiente na execução das tarefas. A pesquisa conclui que a abordagem pode promover a colaboração de maneira mais estratégica, conectando professores com base em suas competências pedagógicas e interesses. No entanto, a validação em um ambiente real e a ampliação da base de dados são etapas necessárias para a aplicação em larga escala.

Palavras-chave: planejamento colaborativo de aulas; sistemas de recomendação; aprendizado colaborativo; prática docente; planejamento de aulas.

ABSTRACT

This paper proposes the development of a recommendation system based on artificial intelligence techniques to enhance collaboration among teachers on the Inovaula.com platform. The study aims to integrate advanced AI techniques to personalize connections between educators, facilitating collaborative lesson planning. The system uses synthetic data generation to simulate educational scenarios and validate the recommendation model. The proposed recommendation algorithm, HNSW-DSM, was developed to balance pedagogical relevance and diversity in suggestions, while addressing the "cold start" challenge in recommendation systems. Results indicated that the system was able to generate personalized recommendations, even for new users, and provide efficient performance in executing tasks. The research concludes that this approach can promote collaboration in a more strategic way, connecting teachers based on their pedagogical skills and interests. However, real-world validation and data scaling are necessary steps for large-scale implementation.

Keywords: collaborative lesson planning; recommender systems; collaborative learning; teaching practice; lesson planning.

1. Introdução

A educação contemporânea enfrenta o paradoxo do isolamento profissional dos professores, que, apesar de atuarem em comunidades escolares, frequentemente se veem restritos a interações limitadas a pequenos grupos, dificultando a troca de conhecimento e a construção de estratégias pedagógicas colaborativas (Bakkenes et al., 1999; Ostovar-Nameghi & Sheikahmadi, 2016). Esse isolamento tem impactos negativos no bem-estar dos professores e na qualidade do ensino (Ostovar-Nameghi & Sheikahmadi, 2016).

A colaboração entre docentes é uma estratégia eficaz para mitigar esse isolamento, promovendo o aprimoramento pedagógico e a troca de práticas de co-criação (Queiros, 2022), especialmente por meio do planejamento colaborativo de aulas (Collaborative Lesson Planning – CLP) (Mendoza et al., 2022).

Embora plataformas digitais tenham sido exploradas para facilitar a colaboração entre professores, elas ainda enfrentam desafios significativos, como a falta de mecanismos que conectem os docentes de maneira significativa e personalizada, resultando em interações muitas vezes superficiais (Tammets et al., 2022; Liu & Zou, 2014). A falta de personalização nas conexões, considerando interesses pedagógicos, áreas de atuação e padrões de comportamento, limita a criação de redes colaborativas realmente eficazes.

Diante desse cenário, o objetivo geral deste trabalho é desenvolver um sistema de recomendação, capaz de analisar padrões de comportamento docente e sugerir conexões personalizadas, com foco na melhoria do planejamento colaborativo de aulas e na criação de redes de aprendizado contínuo.

Para alcançar esse objetivo, este trabalho visa, primeiramente, analisar sistemas de colaboração docente existentes e suas efetividades, a fim de identificar as lacunas que o modelo proposto pretende superar. Em seguida, serão definidos os critérios que orientarão a recomendação de conexões entre professores, com base em seus interesses pedagógicos, áreas de atuação e comportamentos na plataforma. O desenvolvimento de um protótipo funcional do sistema de recomendação será o próximo passo, com a implementação das técnicas de

inteligência artificial para sugerir conexões personalizadas. Por fim, será realizada uma avaliação das recomendações do sistema, verificando assim a eficácia do sistema do cenário de planejamento docente colaborativo.

Nas seções subsequentes, será detalhado o referencial teórico, que abrange os conceitos fundamentais sobre sistemas de recomendação e a colaboração docente. Em seguida, a metodologia adotada será descrita, incluindo as etapas de geração de dados sintéticos, desenvolvimento do algoritmo de recomendação e os procedimentos de avaliação do sistema. A análise dos resultados será apresentada, destacando a eficácia do sistema proposto na promoção de conexões entre professores e na personalização das recomendações. Por fim, serão discutidas as conclusões do estudo, suas limitações e as perspectivas para futuros aprimoramentos e validações em ambientes reais.

2. Inovaula.com

O **Inovaula** é uma plataforma digital voltada para o planejamento pedagógico, auxiliando professores na criação de práticas educacionais por meio de três componentes colaborativos principais: planos de aula, recursos educacionais e banco de questões.

Os planos de aula oferecem modelos personalizáveis adaptáveis por tema, disciplina e ano escolar, permitindo a definição de objetivos, conteúdos, metodologias e critérios de avaliação. Além disso, integram recursos multimídia, possibilitam acompanhamento do progresso e incentivam a colaboração entre docentes (Silva et al., 2024; Ferreira, 2015).

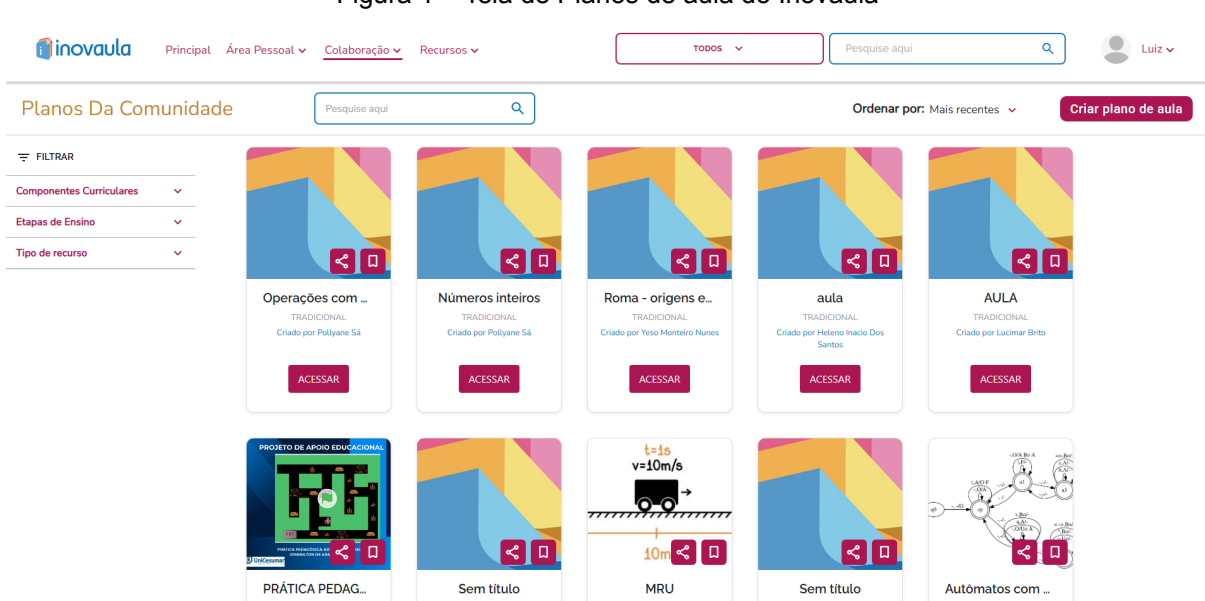
Os recursos educacionais incluem vídeos explicativos, infográficos, fichas de exercícios e jogos interativos, organizados por disciplina e tema curricular, tornando o aprendizado mais dinâmico e acessível (Silva et al., 2024).

O banco de questões possibilita a criação de avaliações personalizadas, oferecendo diversos formatos de perguntas (múltipla escolha, verdadeiro ou falso, dissertativas), além de um sistema de correção automática e geração de relatórios detalhados sobre o desempenho dos alunos (Ferreira, 2015).

Dessa forma, o Inovaula fortalece a prática pedagógica ao incentivar a colaboração, disponibilizar materiais didáticos diversificados e facilitar o monitoramento do aprendizado. Além disso, a plataforma conta com diversas outras ferramentas que apoiam os professores, como a criação de comunidades de prática, o compartilhamento de recursos educacionais abertos e a possibilidade de integração com outras plataformas de gestão educacional (Silva et al., 2024; Ferreira, 2015).

Na Figura 1, pode-se ver brevemente a tela de Planos de Aula da plataforma. Pode-se ver que são planos de aula públicos, que podem ser visualizados e avaliados por outros professores, que também podem realizar download desses planos.

Figura 1 – Tela de Planos de aula do Inovaula



Fonte: Inovaula.com

3. Referencial Teórico

3.1 Modelo SECI

Um modelo teórico que embasa essa dinâmica é o SECI – Socialização, Externalização, Combinação e Internalização – proposto por Nonaka e Takeuchi (1995). O modelo teoriza como o conhecimento tácito é convertível em

conhecimento explícito e, subsequentemente, reintegrado na prática pedagógica. A socialização possibilita a troca de saberes por meio da interação direta, enquanto a externalização transforma esses conhecimentos informais em formatos mais sistematizados; a combinação organiza e integra esse saber, e a internalização o coloca em prática. Todo esse processo é um ciclo, como mostrado na Figura 2.

Figura 2 – Ciclo de interação do modelo SECI



Fonte: <https://gestaodoconhecimentounis.blogspot.com/2016/10/nonaka-e-takeushi.html>

3.2 Sistemas de Recomendação

Sistemas de Recomendação (SR) são mecanismos que personalizam a filtragem de informações, sendo projetados para prever automaticamente e destacar itens que possam ser relevantes para os usuários (Gonzalez; Gamacho, 2020). Os SR, então, desempenham um papel fundamental ao sugerirem informações relevantes, proporcionando uma experiência positiva em plataformas de e-commerce e serviços de streaming (Deng et al., 2014).

De acordo com Carl Bentzer et al. (2023), os SR operam por meio de algoritmos que analisam padrões comportamentais e preferências dos usuários, adaptando-se dinamicamente para sugerir itens relevantes. Estes algoritmos utilizam técnicas que permitem a análise de grandes volumes de dados e a tomada de decisões personalizadas sem intervenção humana direta (Gonzalez; Gamacho, 2020). Eles aprendem o perfil do usuário com base em suas interações passadas e utilizam esse conhecimento para prever itens que possam ser de seu interesse, caracterizando o uso de Machine Learning (ML).

Por conseguirem analisar e interpretar dados complexos, adaptando suas recomendações ao longo do tempo, os RS representam uma aplicação prática da IA. Essa adaptabilidade é alcançada por meio de algoritmos de ML, que permitem ao sistema aprender padrões e ajustar suas previsões conforme novas informações se tornam disponíveis.

Figura 3 – Exemplo de um sistema de recomendação baseado em conteúdo



Fonte: <https://didatica.tech/sistemas-de-recomendacao/>

A Figura 3 mostra um exemplo básico de um RS, no cenário de filtragem baseada em conteúdo, onde 2 perfis que previamente haviam feito uma pesquisa em comum, ao um dos usuários ter feito outra pesquisa, o RS identificou como uma potencial recomendação para o outro usuário.

Portanto, os sistemas de recomendação exemplificam como a IA pode ser aplicada para aprimorar a experiência do usuário, facilitando a descoberta de itens relevantes em meio a grandes volumes de dados. (J. Bobadilla et al., 2012; Pereira et al., 2021; Pereira et al., 2022; Pereira et al., 2024).

4. Método

Este trabalho propõe desenvolver um sistema de recomendação para potencializar a colaboração entre professores na plataforma Inovaula.com. O sistema utilizará técnicas de IA para analisar padrões de comportamento, interesses e competências dos docentes, gerando recomendações personalizadas que incentivem conexões

estratégicas entre pares para o planejamento colaborativo de aulas. Dessa forma, a IA atuará como mediadora inteligente, fomentando a troca de conhecimentos e a criação de parcerias estratégicas entre docentes.

4.1 Paradigma

Este trabalho adota uma abordagem de métodos mistos (Halcomb, 2015), estruturada em três eixos principais: (1) geração de dados sintéticos para simular cenários educacionais complexos, (2) desenvolvimento de um algoritmo de recomendação (HNSW-DSM) e (3) validação teórica do sistema por meio da verificação dos dados gerados pelas recomendações do sistema, como a diversidade das recomendações, o tempo de resposta e o equilíbrio entre relevância pedagógica e popularidade.

4.2 Procedimentos

O método fundamenta-se na premissa de que dados sintéticos, em um cenário de escassez de dados reais, gerados via Funções de Distribuição de Probabilidade (PDFs), permitem replicar padrões comportamentais de professores em ambientes digitais, conforme proposto por Pasinato et al. (2013). Esses dados alimentam o algoritmo HNSW-DSM (Mengzhao Wang et al., 2025; Guo et al., 2018), que combina técnicas de *approximate nearest neighbor search* (ANNS) com ajustes contextuais para equilibrar popularidade e relevância pedagógica. A validação ocorre em um ambiente controlado, onde métricas como análise de dispersão das recomendações e tempo de resposta são aplicadas.

4.2.1 Geração de Dados Sintéticos

A realização de experimentos com a base de dados da plataforma Inovaula.com apresenta limitações relacionadas à incompletude das informações dos professores e dos planos de aula, impactando a eficácia de modelos de recomendação. Muitos professores não preenchem dados essenciais, como formação acadêmica e áreas de atuação, além da baixa popularidade de diversos perfis e materiais educacionais. Essas limitações dificultam o treinamento de modelos de IA, especialmente no tratamento de novos usuários, o cenário de cold start (Pasinato et al., 2013).

Para mitigar esses problemas, optou-se pela geração de dados sintéticos. Essa técnica, amplamente estudada na literatura e utilizada em modelos de ML, permite simular diferentes cenários educacionais e diversificar as situações avaliadas pelo sistema de recomendação (Pasinato et al., 2013).

A modelagem dos dados sintéticos seguiu uma abordagem baseada em PDFs, garantindo que as características dos professores e suas interações na plataforma fossem representadas de forma realista e coerente com um ambiente educacional digital (Baker & Siemens, 2014). Cada variável foi modelada independentemente, considerando sua distribuição estatística apropriada para refletir padrões comportamentais esperados.

A base de dados sintética gerada foi composta por duas entidades principais: **‘professores’** e **‘planos de aula’**. A entidade **‘professor’** contém várias características. O **ID** é a chave primária que identifica de maneira única cada professor. O **‘nível de ensino’** pode ser do Ensino Fundamental ou Ensino Médio, com base na área de atuação de cada professor. A **‘área de atuação’** categoriza o professor em áreas específicas para o nível de ensino, como física no Ensino Médio e ciências no Ensino Fundamental. Além disso, a variável **‘seguidores’** representa a quantidade de outros professores que seguem esse professor.

Já a entidade **plano de aula** é caracterizada por um **ID**, a qual é a chave primária do plano de aula. O **ID do professor** serve como chave estrangeira, estabelecendo um relacionamento 1:N com a entidade professor. O **nível de ensino** no plano de aula especifica se ele é destinado ao Ensino Fundamental ou Ensino Médio, e a **área de atuação** define a disciplina ou tema do plano de aula. O campo **visualizações** registra o número de vezes que o plano de aula foi visualizado por outros professores. O campo **avaliações** armazena a nota média dada por outros professores ao plano de aula, e a **quantidade de avaliações** indica quantas vezes o plano recebeu uma avaliação. Finalmente, o campo **downloads** registra quantas vezes o plano de aula foi baixado por outros professores.

Essa estrutura pode ser vista na Figura 4, que mostra um exemplo dessa estrutura em um json.

Figura 4 – JSON de exemplo da estrutura dos dados

```
{
  "id": 6,
  "level": "Fundamental",
  "area": "História",
  "followers": 1075,
  "name": "Hadassa Mendonça",
  "learning_experiences": [
    {
      "teacher_id": 6,
      "area": "História",
      "level": "Fundamental",
      "views": 1415,
      "rating": 6.1,
      "rating_qtd": 663,
      "downloads": 123.0
    }
  ]
},
```

Fonte: O autor (2025).

Os professores foram categorizados conforme dois níveis de ensino: Ensino Fundamental e Ensino Médio, distribuídos igualmente entre as áreas específicas de cada nível. A divisão foi estabelecida para garantir uma proporção equilibrada entre os docentes de diferentes áreas de atuação educacionais, permitindo que o sistema de recomendação avaliasse padrões de interação entre os grupos. Essa distribuição

assegura diversidade suficiente na base sintética, permitindo simular recomendações contextualizadas dentro de cada segmento escolar.

O número de seguidores foi gerado por uma distribuição exponencial, refletindo a tendência de poucos professores altamente influentes e muitos com menor visibilidade. Para simular novos usuários, 10% dos perfis foram gerados sem seguidores. Esse modelo tenta refletir a dinâmica real de redes educacionais, onde professores com maior reputação atraem mais conexões, enquanto novos usuários apresentam menor visibilidade inicial.

O número de planos de aula seguiu uma distribuição controlada: 30% dos professores com 1 plano, 40% com 2, 25% com 3, 3% com 4 e 2% com 5 planos, considerando a hipótese de que professores mais engajados produzem mais conteúdos.

Tabela 1 – Distribuição de Avaliações

Categoria de Avaliação	Faixa de Notas	Percentual de Planos Avaliados
Rating 1 (baixa)	0.1 a 4.99	10%
Rating 2 (média)	5.0 a 6.99	40%
Rating 3 (alta)	7.0 a 10.0	50%

Fonte: O autor (2025).

Cada avaliação foi associada a uma distribuição uniforme em faixas pré-definidas, conforme a Tabela 1. A quantidade de avaliações foi calculada para ser parcialmente proporcional ao número de seguidores, podendo ter um valor aleatório entre 0 e 1,5 vezes o número de seguidores. Essa estrutura tenta simular cenários realistas de feedback na plataforma, permitindo avaliar a influência das avaliações na recomendação de conteúdos.

A distribuição das visualizações e downloads seguiu um modelo baseado na relação entre popularidade da área, número de seguidores e avaliação recebida. Para o número de visualizações, optou-se por um valor mínimo igual ao número de avaliações (pois um plano precisa ser visualizado antes de ser avaliado), e um máximo estimado entre 2 a 3 vezes o número de seguidores, refletindo diferentes níveis de engajamento. Já o número de downloads foi definido com base em uma

probabilidade proporcional à avaliação do plano e à visibilidade do professor, conforme a fórmula:

$$Downloads = \frac{Visualizações}{Total\ de\ professores\ por\ área} \times \frac{Avaliação\ média}{10}$$

Esse cálculo garante que professores com alta avaliação e maior visibilidade tenham uma maior taxa de downloads de seus materiais, assim como também o inverso. Tal medida é fundamentada na ideia de que a avaliação dos conteúdos e as visualizações podem ter um impacto direto na taxa de engajamento com os materiais, considerando que um plano de aula que teve muitas avaliações, dificilmente teria poucos downloads.

A validade dos dados gerados foi avaliada por meio da análise de estatísticas descritivas e distribuição das variáveis, garantindo que os padrões sintéticos seguissem tendências realistas observadas em ambientes educacionais digitais.

- **Verificação da coerência estrutural:** comparação entre a dispersão das variáveis.
- **Análise gráfica:** a distribuição dos dados foi visualizada para garantir diversidade e evitar padrões enviesados.

4.2.2 Desenvolvimento do Algoritmo de Recomendação

A seleção do algoritmo de recomendação foi guiada por dois critérios principais: **(i) capacidade de identificar padrões relevantes nas interações dos professores na plataforma** e **(ii) eficiência na execução, garantindo escalabilidade para um grande volume de dados**. Inicialmente, considerou-se o uso do algoritmo K-Nearest Neighbors (KNN) (Guo et al., 2018), devido à sua simplicidade e interpretabilidade. No entanto, sua aplicação revelou limitações, tanto na dificuldade de lidar com base de dados grandes (ZHANG, 2022), quanto na modelagem da popularidade dos professores, o que motivou a adoção de uma abordagem mais avançada, baseada no Hierarchical Navigable Small World (HNSW), combinada com o método de similaridade por difusão (DSM - Diffusion Similarity Method) (Guo et al., 2018).

A implementação do modelo foi desenvolvida em Python, escolhido por sua ampla adoção em projetos de *Machine Learning* e disponibilidade de bibliotecas otimizadas (e.g., FAISS para ANNS, NumPy para operações vetoriais).

4.2.2.1 Seleção e Adaptação do Algoritmo

O algoritmo K-Nearest Neighbors (KNN) é amplamente utilizado em sistemas de recomendação devido à sua capacidade de identificar itens semelhantes com base em métricas como distância euclidiana ou similaridade do cosseno (Wang et al., 2006). No contexto deste estudo, o KNN permitia recomendar professores com base em características compartilhadas, como área de atuação, formação acadêmica e práticas pedagógicas.

Entretanto, o KNN apresenta algumas limitações importantes: foco excessivo na similaridade estrutural, e complexidade alta para busca em bases de dados grandes. A recomendação ocorre entre professores com perfis semelhantes, sem considerar fatores como popularidade ou impacto pedagógico na plataforma. Para contornar essa limitação, procurou-se na literatura por abordagens que integrassem a popularidade. O modelo KNN-DSM proposto por Guo et al, 2018 mostrou-se eficaz para esse cenário. Porém, a dificuldade do KNN em fazer buscas em bom tempo em bases de dados grandes limitou o desempenho do modelo. Por isso optou-se por um modelo que combinasse a eficiência do HNSW (Mengzhao Wang et al., 2025), além de incorporar ajustes baseados em popularidade e relevância educacional por meio do DSM.

O HNSW foi escolhido como substituto do KNN por sua capacidade de estruturar grandes volumes de dados em múltiplas camadas hierárquicas, permitindo buscas mais rápidas e eficientes (Mengzhao Wang et al., 2025). A principal vantagem do HNSW é a possibilidade de criar conexões entre elementos distantes no espaço vetorial, reduzindo a complexidade da busca para $O(\log n)$, sem comprometer a precisão das recomendações.

Além da melhoria na eficiência computacional, foi necessário incorporar um mecanismo que equilibrasse a popularidade e relevância pedagógica, de modo a evitar que somente professores altamente engajados fossem recomendados. Para

isso, foi implementado o DSM, proposto por Guo et al. (2018), que introduz parâmetros para ajuste contextual.

4.2.2.2 Pré-processamento

Antes da aplicação do algoritmo de recomendação, os dados gerados foram definidos como o conjunto de treino, e passaram por um processo de pré-processamento, garantindo a uniformidade e estruturação adequada das informações. As principais etapas foram:

- Codificação de variáveis categóricas: transformação de atributos como área de atuação e nível de ensino em valores numéricos via LabelEncoder.
- Normalização dos dados: ajuste das variáveis quantitativas (número de seguidores, quantidade de planos de aula, média de avaliações, total de visualizações, downloads) para um intervalo entre 0 e 1, utilizando MinMaxScaler, evitando que variáveis com escalas maiores dominem a similaridade.
- Construção da matriz de perfis: cada professor foi representado como um vetor multidimensional contendo suas características numéricas.

Uma demonstração da matriz com área e nível de ensino codificados, e o restante dos dados normalizados pode ser vista na Figura 5.

Figura 5 – Exemplo de matriz do pré-processamento

	Área de ensino	Nível de ensino	Seguidores	Número de Planos	Média das Avaliações	Total de Visualizações	Média de Downloads	Média de Qtd. de Avaliações
Professor A	0.75	1.0	0.4936	0.5	0.396	0.5445	0.332	0.2888
Professor B	0.8333	0.0	0.4486	0.5	0.765	0.2225	0.1673	0.1461
Professor C	0.75	0.0	0.0998	0.75	0.9089	0.0909	0.009	0.0436
Professor D	0.25	0.0	0.16	0.5	0.8022	0.1758	0.0725	0.068
Professor E	0.0	0.0	0.6718	0.5	0.553	0.6566	0.6646	0.3695

Fonte: O autor (2025).

Os encoders e scalers foram armazenados em arquivos tipo .pkl, permitindo a padronização de novos dados na fase de recomendação.

4.2.2.3 Parâmetros de Ajuste Contextual

Os parâmetros $\lambda = 0,5$ e $\beta = 0,5$ foram definidos para equilibrar a influência da popularidade na recomendação, seguindo a abordagem de *Difusion Similarity Method* (DSM) proposta por Guo et al. (2018). A lógica é:

- λ (lambda): Controla o peso da popularidade do candidato. Valores próximos a 1 priorizam professores com alta visibilidade (ex.: muitos seguidores), enquanto valores menores favorecem perfis menos populares.
- β (beta): Atenua a influência da popularidade do professor consultado. Isso evita que professores já populares recebam recomendações redundantes, promovendo diversidade.

A escolha $\lambda = \beta = 0,5$ busca um equilíbrio empírico, entre relevância e diversidade, alinhado ao objetivo de fomentar trocas heterogêneas no modelo SECI (Nonaka & Takeuchi, 1995).

4.2.2.4 Cálculo de Popularidade Contextual

A popularidade foi calculada considerando uma combinação linear ponderada de diferentes métricas pedagógicas, conforme a equação:

$$Popularidade = 0.2S + 0.2V + 0.3(A * Q) + 0.2D + 0.1Pl$$

Onde S é Número de seguidores; V é o Total de visualizações dos planos de aula; A é a Avaliação média dos planos; Q é a Quantidade de avaliações recebidas; D é o Número de downloads dos planos; e Pl o Número de planos de aula publicados.

Pesos diferenciados foram aplicados para refletir a importância relativa de cada métrica. Avaliações tiveram maior peso (**30%**), por indicar a qualidade e o impacto prático dos planos de aula. Os demais parâmetros foram divididos igualmente em 20% de relevância, na ideia de haver certo equilíbrio entre eles, enquanto a quantidade de planos de aula sendo o menos relevante (10%),

fundamentado na ideia de que não necessariamente ter muitos planos publicados torna o professor popular (ABDOLLAHPOURI et al., 2021). No final, o resultado é normalizado para ficar entre 0 e 1.

4.2.2.5 Treinamento e Predição

O treinamento do sistema de recomendação inicia-se com a etapa de pré-processamento dos dados do conjunto de treino, na qual são extraídas e normalizadas as características dos professores, formando uma matriz vetorial representativa dos perfis docentes.

Em seguida, o índice HNSW é inicializado com hiperparâmetros cuidadosamente escolhidos: o número de conexões por nó (M) e o parâmetro de construção (efConstruction), que controla o esforço computacional na fase de inserção, e k-neighbors, o qual é a quantidade de professores que o modelo deve retornar como recomendação. Os valores utilizados no modelo para esses parâmetros foram respectivamente: 32 (M), 200 (efConstruction), 3 (k-neighbors). Com exceção do conjunto de teste, em que k-neighbors foi modificado para 10.

O método Train_HNSW (Figura 6) recebe a matriz de perfis docentes normalizados e constrói, em tempo aproximado linear, a estrutura de vizinhança hierárquica que permitirá buscas eficientes. Simultaneamente, calcula-se um dicionário de popularidade contextual, no qual cada identificador de professor é associado a uma pontuação escalar normalizada, obtida pela agregação ponderada de seguidores, visualizações e avaliações. Ademais, mapeia-se uma tabela de contexto que agrupa os docentes por par (área, nível), facilitando buscas direcionadas em domínios específicos.

Após o treinamento, o modelo é salvo em três arquivos distintos: o índice HNSW (formato .faiss), os perfis vetoriais dos professores (em formato .npy) e os metadados essenciais (em .pkl), que incluem os mapeamentos de identificadores e popularidade. Tais passos podem ser vistos no pseudo algoritmo abaixo.

Figura 6 – Pseudo algoritmo de treinamento

```

Function Train_Model(teachers, lesson_plans, matrix):

    hnsw_index = Initialize_HNSW(matrix_dimensions, connections)
    set_hnsw_index_params(hnsw_index, efConstruction)

    Train_HNSW(hnsw_index, matrix)

    teacher_popularity = {}
    For each teacher in teachers:
        Calculate popularity based on followers, views, ratings, etc.
        Store in popularity dictionary

    context_mapping = {}
    For each teacher in teachers:
        context_key = (teacher.area, teacher.level)
        If context_key not in context_mapping:
            context_mapping[context_key] = []
        Append teacher.id to context_mapping[context_key]

    metadata = {
        'teacher_popularity': teacher_popularity,
        'context_mapping': context_mapping
    }

    Save_metadata(metadata, 'metadata.pkl')

```

Fonte: O autor (2025).

Durante a etapa de predição, o sistema recebe o índice do docente de consulta e um indicador de “cold-start”. Inicialmente, recupera-se o vetor de perfil associado ao índice e avalia-se a popularidade relativa desse docente. Se o caso for de cold-start ou se a popularidade for inferior a 0.1, o algoritmo seleciona os K docentes mais populares dentro do mesmo contexto (mesma área e nível) e amostra aleatoriamente K finalistas, promovendo variedade nas recomendações.

Caso contrário, procede-se a uma busca HNSW, configurando o parâmetro efSearch para $K \times F$, sendo F o fator de busca, de modo a equilibrar abrangência e desempenho. A busca retorna distâncias e índices de candidatos; descartam-se o próprio docente e valores inválidos. Paralelamente, extraem-se os $K \times F$ docentes mais populares no mesmo contexto, gerando o conjunto populacional. O conjunto final é obtido pela união desses dois grupos de candidatos.

Para cada par (consulta, candidato), calcula-se a distância normalizada; em seguida, a aplicação do método DSM acontece ao gerar uma pontuação ajustada. Ordena-se então os candidatos pelo score decrescente, retendo os K melhores, e realiza-se uma amostragem aleatória final de tamanho K para promover diversidade

no conjunto de saída. Deste modo, o modelo entrega recomendações que aliam similaridade semântica e popularidade contextual equilibradamente.

Figura 7 – Pseudo algoritmo de predição

```
Function Predict(teacher_index, cold_start_flag):

    teacher_id      = teacher_id_mapping[teacher_index]
    query_popularity = teacher_popularity[teacher_id]
    is_cold_start   = cold_start_flag OR (query_popularity < 0.1)
    context_ids     = context_mapping[(teacher.area, teacher.level)]

    If is_cold_start Then:
        sorted_context_ids = sort(context_ids, by=teacher_popularity, descending=True)
        expanded_ids      = take(sorted_context_ids, k_neighbors * diversity_factor)
        sampled_ids       = random_sample(expanded_ids, k_neighbors)
        Return sampled_ids

    Else:
        k_search      = k_neighbors * search_factor # quantos vizinhos buscar
        index.hnsw.efSearch = k_search
        (dists, idxs)  = index.search(profile_vector(teacher_index), k_search)

        hnsw_candidates = [i for i in idxs if i != teacher_index and i != -1]

        sorted_ctx      = sort(context_ids, by=teacher_popularity, descending=True)
        pop_candidates   = take(sorted_ctx, k_search)

        combined        = union(hnsw_candidates, pop_candidates)

        distances       = compute_L2_distances(teacher_index, combined, dists, idxs)

        max_d           = max(distances)
        norm_dists      = [d / max_d for d in distances]

        adjusted_scores = {
            cand: lambda_param * (1 - norm_dists[i])
                  + beta_param * teacher_popularity[cand]
            for i, cand in enumerate(combined)
        }

        sorted_combined = sort(combined, by=adjusted_scores[cand], descending=True)
        top_candidates  = take(sorted_combined, k_neighbors * diversity_factor)
        final_candidates = random_sample(top_candidates, k_neighbors)

    Return final_candidates
```

Fonte: O autor (2025).

Todo o processo foi desenvolvido em Python, utilizando as bibliotecas FAISS, NumPy, SQLAlchemy e Joblib para garantir desempenho, modularidade e persistência dos modelos. Os experimentos foram realizados em um notebook Dell G15 equipado com processador Intel Core i5-13450HX de 13ª geração, e placa

gráfica NVIDIA RTX 4050. No entanto, todo o processamento foi executado exclusivamente pela CPU, sem utilização dos recursos da GPU.

4.2.2.6 Avaliação da Efetividade

A avaliação da efetividade do sistema de recomendação foi conduzida a partir de um subconjunto de teste, extraído do conjunto de treino. Para isso, selecionou-se aleatoriamente 1.000 professores para compor esse conjunto de teste. A avaliação foi feita com base em três dimensões analíticas: distribuição dos scores ajustados, relação entre a distância vetorial e popularidade, e desempenho computacional.

A análise da distribuição dos scores ajustados considerou a pontuação final atribuída pelo algoritmo aos perfis recomendados, considerando o balanceamento entre similaridade, popularidade e relevância contextual. Para essa métrica, dos cem mil professores gerados, foram sorteados mil de forma totalmente randômica. Para cada professor, o sistema gerou 10 recomendações, e para cada candidato, foi extraído o score ajustado conforme o modelo DSM. Esses dados foram separados por cenário e visualizados por meio de histogramas, possibilitando a análise da dispersão, tendência central e assimetria das pontuações atribuídas.

A relação entre a distância vetorial e popularidade foi investigada visando identificar possíveis vieses na recomendação. Para isso, em cada recomendação gerada nos cenários de cold start e de histórico consolidado, foram coletadas duas variáveis: a distância entre os vetores dos professores envolvidos, conforme definido pelo modelo HNSW; e a popularidade, calculada conforme a fórmula apresentada na seção 4.3.2.4. Os dados foram estruturados em pares e plotados em gráficos de dispersão, permitindo a observação visual da tendência. Para quantificar essa relação, foi aplicado o coeficiente de correlação de Spearman, que mede a dependência monotônica entre as variáveis envolvidas.

A análise do desempenho computacional consistiu na mensuração do tempo médio de execução das recomendações. Em cada teste, mediu-se o tempo necessário para carregar o vetor do professor, calcular o conjunto de candidatos e gerar as recomendações. A medição foi feita utilizando a biblioteca *time* da

linguagem Python, com o cronômetro sendo iniciado antes da chamada da função de recomendação e encerrado após o recebimento da lista. Ao final, os dados do tempo de execução de cada recomendação foram inseridos num histograma.

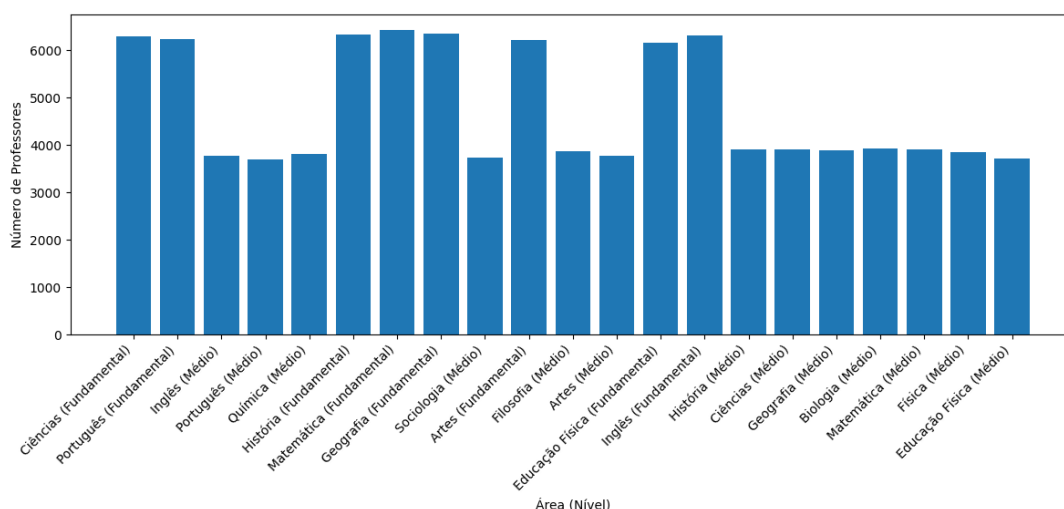
5. Resultados e Discussão

5.1 Resultado da Geração de Dados Sintéticos

A etapa de geração de dados sintéticos resultou em um conjunto robusto de informações acerca dos professores e de suas práticas pedagógicas simuladas, cerca de 100 mil professores e 190 mil planos de aula.

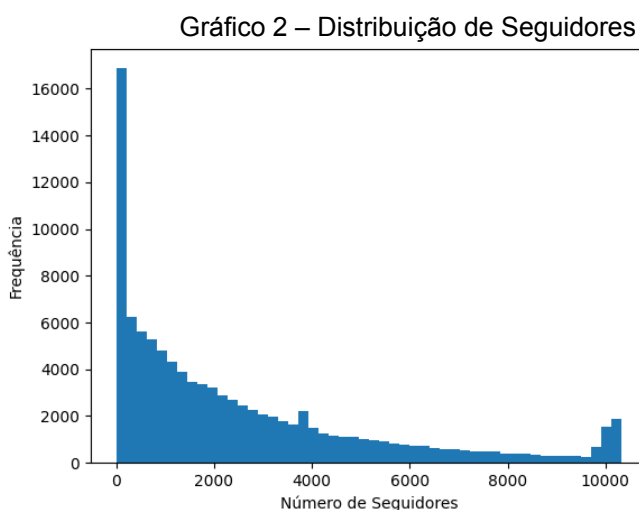
O Gráfico 1 apresenta a distribuição dos professores por área de conhecimento e nível de ensino. Nota-se que a geração sintética garantiu uma diversidade balanceada entre os segmentos do Ensino Fundamental e Médio. Esse balanceamento foi fundamental para assegurar que o sistema de recomendação não favoreça desproporcionalmente determinadas áreas do conhecimento, promovendo equidade nas sugestões geradas.

Gráfico 1 – Número de Professores por Área e Nível



Fonte: O autor (2025).

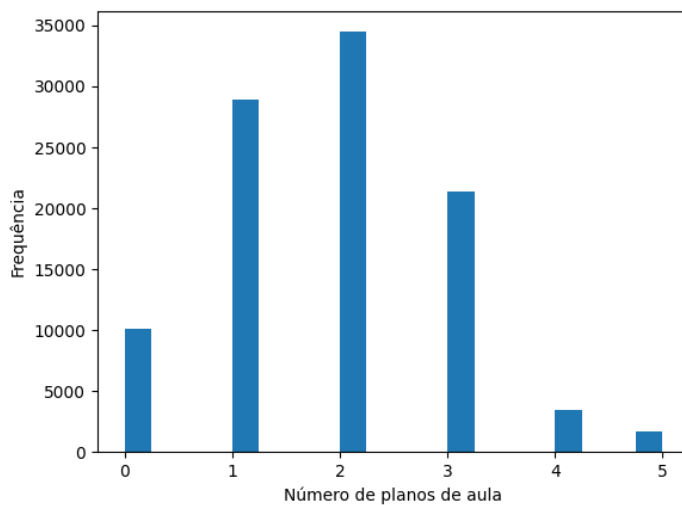
O Gráfico 2, que exibe a distribuição de seguidores, apresenta um comportamento esperado em redes sociais: a maioria dos professores tem poucos seguidores, enquanto poucos docentes concentram um número elevado de conexões. Esse padrão de distribuição exponencial, comum em redes sociais, reflete a realidade de plataformas online, onde alguns usuários se tornam hubs de influência enquanto a maioria mantém um alcance mais limitado. Para o sistema de recomendação, essa assimetria é relevante, ao indicar a necessidade de estratégias para evitar viés excessivo na recomendação baseada em popularidade.



Fonte: O autor (2025).

A distribuição de planos de aula por professor, mostrada no Gráfico 3, revela que a maioria dos docentes cria entre 1 e 3 planos de aula, enquanto somente uma pequena parcela produz 4 ou 5 planos. Esse padrão sugere haver um nível moderado de engajamento na produção de conteúdos pedagógicos no ambiente simulado. Essa observação reforça a necessidade de mecanismos de incentivo à criação de novos planos de aula, uma vez que professores mais produtivos tendem a enriquecer a plataforma com maior variedade de materiais.

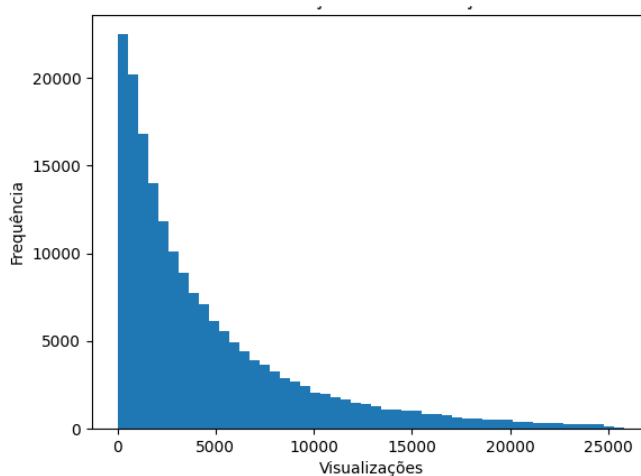
Gráfico 3 – Distribuição de Planos de Aula por Professor



Fonte: O autor (2025).

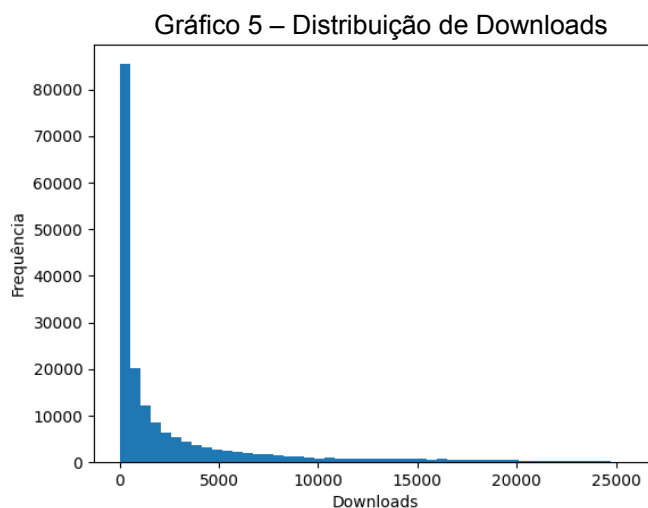
O Gráfico 4, que retrata a distribuição de visualizações dos planos de aula, apresenta um comportamento assimétrico com caudas longas. Isso significa que, enquanto alguns materiais recebem um volume expressivo de acessos, a maioria dos planos possui uma audiência mais modesta. Esse fenômeno pode ser atribuído à influência de fatores como a reputação dos professores, a estrutura de divulgação dos planos na plataforma e a relevância percebida dos conteúdos. O sistema de recomendação deve, portanto, balancear a visibilidade dos materiais para evitar um efeito de “ricos ficando mais ricos”, em que somente planos populares continuam recebendo atenção.

Gráfico 4 – Distribuição de Visualizações



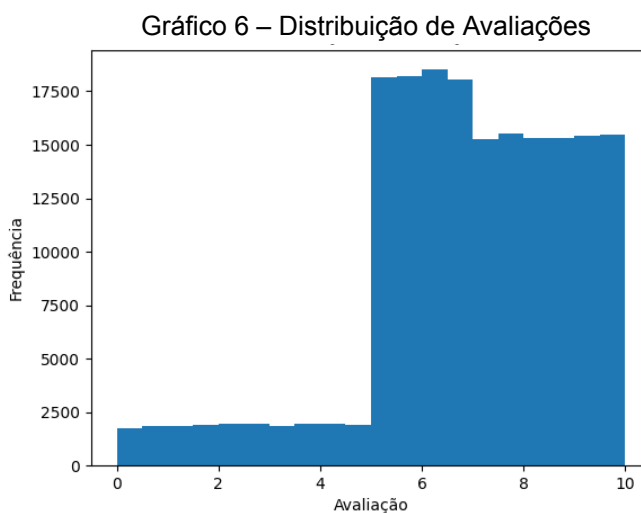
Fonte: O autor (2025).

O Gráfico 5, que ilustra a distribuição de downloads, revela que a maioria dos planos de aula apresenta taxas de download moderadas, enquanto alguns poucos são amplamente disseminados. Esse comportamento reforça a necessidade de um modelo de recomendação que não somente sugira planos de aula altamente populares, mas também promova a descoberta de conteúdos menos visíveis, mas de alta qualidade.



Fonte: O autor (2025).

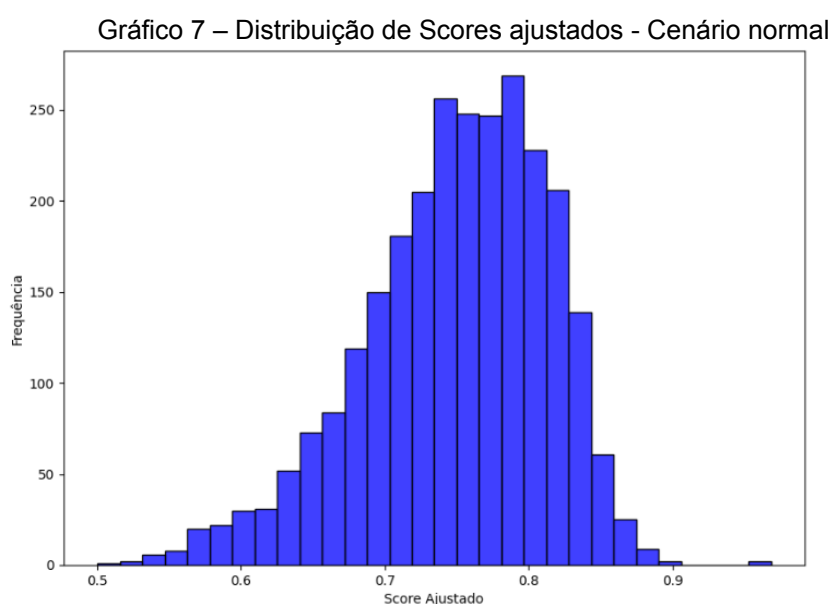
A distribuição de avaliações, exibida no Gráfico 6, demonstra a existência de diferentes perfis de docentes avaliadores. O sistema de recomendação deve considerar a confiabilidade e o perfil dos avaliadores para evitar distorções na priorização dos conteúdos recomendados.



Fonte: O autor (2025).

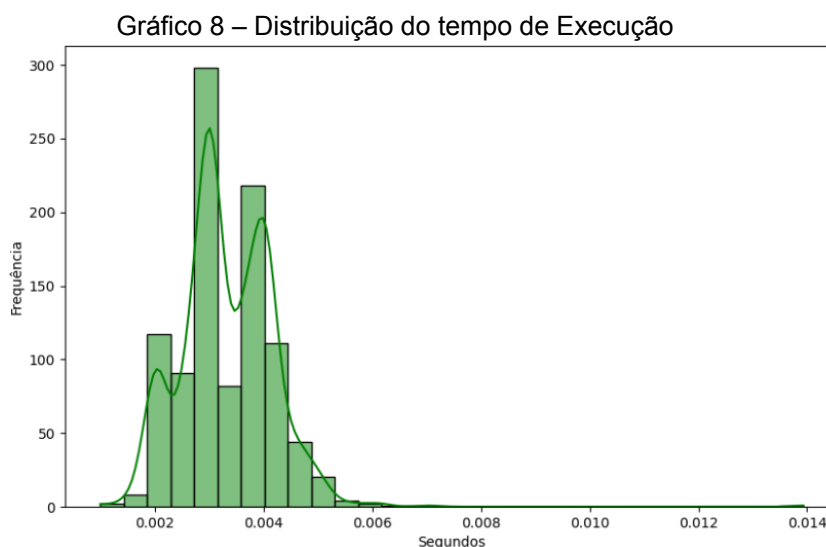
5.2 Resultado das Recomendações

No Gráfico 7, observa-se a distribuição dos scores ajustados no cenário convencional, onde os professores já possuem histórico de interações na plataforma. Nota-se uma maior concentração de recomendações com pontuação elevada, o que indica que o sistema conseguiu identificar perfis relevantes com maior precisão à medida que mais informações foram sendo disponibilizadas. Isso sugere que o algoritmo consegue aprender com os dados históricos e refinar suas sugestões, aumentando a pertinência das recomendações ao longo do tempo.



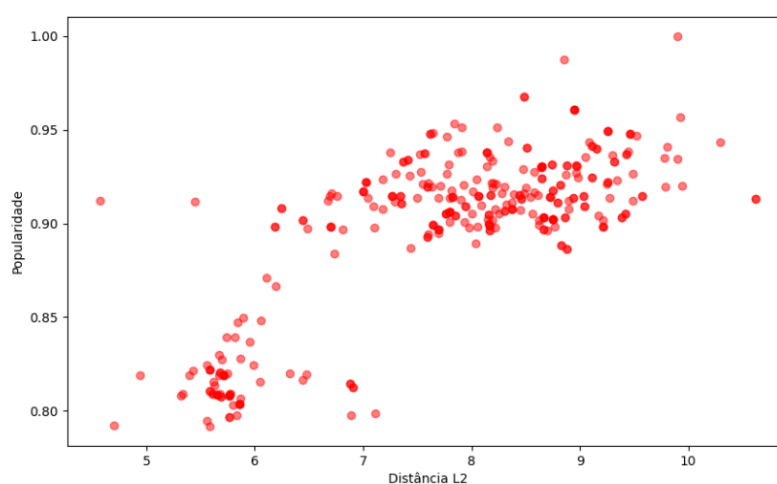
Fonte: O autor (2025).

O Gráfico 8 ilustra o tempo de execução das recomendações, evidenciando que o sistema manteve um desempenho consistente mesmo com um grande volume de dados. A implementação do modelo HNSW-FLASH resultou em uma busca eficiente, reduzindo o tempo necessário para gerar recomendações em comparação com métodos tradicionais, como o KNN puro. Isso garante escalabilidade à solução, permitindo que a plataforma ofereça sugestões em tempo real sem comprometer a experiência do usuário.



O Gráfico 9 apresenta a relação entre a distância vetorial de similaridade no modelo HNSW e a popularidade dos professores no cenário de cold start. O comportamento observado demonstra que, professores mais populares são de fato recomendados, já que a distância vetorial se mostrou grande. Isso demonstra que por serem diferentes dos perfis de cold start, os professores populares são vetorialmente mais distantes, ou seja, menos similares.

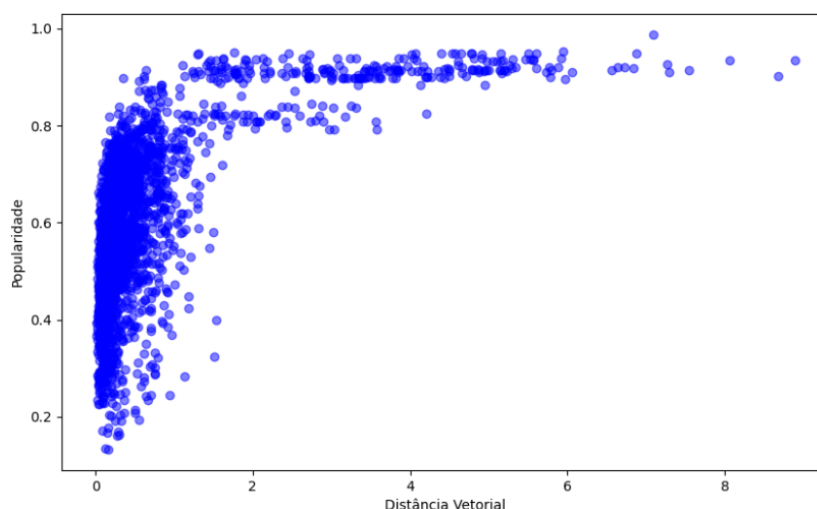
Gráfico 9 – Relação entre Distância vetorial e Popularidade - Cold Start



O Gráfico 10 apresenta a relação entre distância vetorial de similaridade e popularidade no cenário normal, ou seja, quando os professores já possuem um

histórico consolidado na plataforma. A análise do gráfico indica que a distância de similaridade entre os professores recomendados tende a diminuir conforme a popularidade aumenta. Isso significa que professores mais populares geralmente possuem perfis mais bem definidos na rede, facilitando a identificação de conexões mais precisas. No entanto, pode-se notar que os professores populares estão numa linha com o eixo de distância, e popularidade perto de 1, e que essa linha contém baixa concentração de recomendações.

Gráfico 10 – Relação entre Distância vetorial e Popularidade - Cenário Normal



Fonte: O autor (2025).

5.3 Discussão

A análise dos dados gerados a partir da simulação e da aplicação do algoritmo HNSW-FLASH-DSM evidenciou a efetividade do sistema de recomendação proposto em contextos distintos de uso, como o *cold start* e os cenários com histórico consolidado. Foi possível verificar que as hipóteses iniciais quanto à capacidade do modelo de promover conexões personalizadas, equilibrando popularidade e relevância pedagógica, foram confirmadas pelas métricas observadas.

A diversidade das recomendações, medida pela dispersão dos perfis sugeridos, demonstrou que o sistema consegue evitar redundâncias e promover a descoberta de conexões heterogêneas, corroborando com o modelo SECI ao estimular trocas que transcendem o conhecimento tácito. Essa heterogeneidade é

estratégica para o fortalecimento de comunidades docentes mais dinâmicas e menos segmentadas.

No cenário de cold start, os resultados indicaram que o modelo foi eficaz em atenuar os efeitos da ausência de histórico, recorrendo a mecanismos de filtragem contextual e popularidade. Tal comportamento reforça a robustez do modelo em lidar com a escassez de dados, problema recorrente em sistemas educacionais digitais (Gonzalez Camacho, 2020; Pasinato et al., 2013).

A relação entre a Distância vetorial e a popularidade revelou um comportamento desejável: embora no gráfico 11 os perfis que formam uma linha perto de popularidade 1, que representam os professores populares, aparentam ser menos recomendados, é importante observar que perfis populares que também são similares ocupam grande concentração perto da faixa de 0.8 e 0.6 de popularidade, o que pode indicar que de fato há um equilíbrio de frequência de recomendações entre popularidade e similaridade, demonstrando que os ajustes contextuais do DSM foram bem calibrados. Essa observação é relevante do ponto de vista do viés de popularidade, frequentemente criticado em sistemas de recomendação por reforçar assimetrias já existentes (Abdollahpouri et al., 2021).

O tempo de execução das recomendações também se manteve em parâmetros aceitáveis para aplicações em larga escala. A eficiência computacional do modelo validou a escolha dos componentes HNSW, que possibilitaram buscas vetoriais otimizadas mesmo diante de um volume expressivo de dados sintéticos. Esse achado é fundamental para garantir a viabilidade técnica da solução proposta em ambientes reais com múltiplos usuários simultâneos.

Por fim, a análise conjunta dos gráficos e métricas reforça a validade da abordagem proposta. Embora a avaliação tenha se restringido a um ambiente simulado, os resultados obtidos fornecem evidências empíricas consistentes de que o sistema consegue fomentar redes de colaboração docente mais estratégicas e diversificadas. A personalização das conexões, com base em múltiplos critérios pedagógicos e comportamentais, mostrou-se não somente possível, mas desejável, para superar o isolamento profissional e promover uma cultura de co-criação no planejamento de aulas.

6. Considerações Finais

Este trabalho apresentou o desenvolvimento de um sistema de recomendação baseado em inteligência artificial para promover a colaboração entre professores na plataforma Inovaula. A partir da geração de dados sintéticos e da implementação de um modelo híbrido de recomendação, foi possível testar diferentes cenários e analisar o impacto das recomendações personalizadas. Os resultados indicaram que a abordagem proposta pode ser promissora, mas ainda apresenta desafios que devem ser considerados em pesquisas futuras.

6.1 Limitações

Apesar dos esforços para simular cenários educacionais realistas, algumas limitações foram identificadas. Um dos principais desafios é a utilização dos parâmetros de área e nível de ensino com maior peso, limitando as recomendações a esse mesmo contexto. É comum haver professores atuando em mais de uma área de ensino e em diferentes níveis, como docentes de Matemática que também lecionam Física no ensino médio. A modelagem atual assume que cada professor pertence a uma única categoria, o que pode limitar a precisão das recomendações.

Além disso, embora o sistema tenha sido projetado para operar em um ambiente dinâmico, os testes foram realizados com um escopo restrito de dados simulados. Na prática, o sistema precisaria lidar com atualizações constantes dos perfis e conteúdos adicionados pelos usuários. Esse aspecto não foi abordado no presente estudo, tornando-se uma limitação para a aplicabilidade em tempo real.

Outro ponto a ser considerado é a validação do modelo. Embora os resultados tenham demonstrado coerência estatística, a avaliação foi realizada somente com dados sintéticos. Dessa forma, não há garantias de que as recomendações geradas correspondam às preferências e necessidades reais dos professores na plataforma.

6.2 Trabalhos Futuros

Diante das limitações identificadas, algumas direções podem ser exploradas para aprimorar o sistema. Primeiramente, recomenda-se expandir o escopo do modelo para contemplar docentes com atuação em múltiplas áreas e níveis de ensino, visto que muitas práticas pedagógicas contemporâneas envolvem interdisciplinaridade e diferentes faixas etárias. Essa ampliação do perfil docente não somente estreita o alinhamento entre as recomendações e a realidade da docência, mas também poderia revelar padrões de similaridade mais ricos e contextualizados.

Outro aspecto relevante reside na adoção de um mecanismo de treinamento contínuo, capaz de incorporar incrementalmente novas interações e métricas de popularidade. Ao atualizar dinamicamente tanto os vetores perfis quanto o índice HNSW, o sistema manteria sua representatividade em relação às variações temporais de engajamento, o que é especialmente importante em ambientes educacionais em constante evolução.

No que diz respeito à base de dados, torna-se necessário validar rigorosamente a distribuição de probabilidades subjacente ao cálculo de popularidade contextual. Essa verificação permitiria identificar possíveis vieses nos dados ou deficiências na modelagem estatística, orientando realizar ajustes que melhorem a fidelidade dos perfis vetoriais e a consistência das recomendações geradas.

Adicionalmente, é fundamental comparar o desempenho do método proposto com algoritmos de referência, tais como k-Nearest Neighbors e a versão pura do HNSW, bem como explorar variações avançadas dessas estruturas (por exemplo, HNSW-PQ ou HNSW-MR). Essa análise comparativa, baseada em métricas de acurácia, tempo de consulta e consumo de memória, fornecerá uma avaliação mais abrangente das vantagens e limitações do re-ranqueamento DSM.

Para validar a utilidade prática do sistema, estudos experimentais em ambiente real devem envolver docentes como participantes ativos. A coleta de feedback qualitativo e quantitativo permitirá mensurar a aceitação das

recomendações, a usabilidade da interface e o impacto efetivo na colaboração entre professores, fornecendo subsídios para ajustes de usabilidade e de modelo.

Do ponto de vista de escalabilidade, é oportuno investigar estruturas de dados mais robustas para a busca e seleção dos professores populares — atualmente realizada por meio de operações em arrays simples —, de modo a evitar gargalos em bases de registros milionários ou até bilionários. Testes de desempenho com conjuntos de dados significativamente maiores devem ser conduzidos para dimensionar adequadamente custos computacionais e tempos de resposta.

Por fim, recomenda-se aprofundar a análise dos efeitos das recomendações sobre a dinâmica colaborativa da plataforma, por meio de indicadores como taxa de aceitação das sugestões, tempo médio de interação entre pares de docentes e evolução das redes de coautoria ou co-planejamento. A incorporação contínua desses indicadores em ciclos iterativos de desenvolvimento contribuirá para tornar o sistema cada vez mais robusto e alinhado às necessidades do ecossistema educacional.

REFERÊNCIAS

BAKKENES, Inge; DE BRABANDER, Cornelis; IMANTS, Jeroen. Teacher Isolation and Communication Network Analysis in Primary Schools. **Educational Administration Quarterly**, v. 35, n. 2, p. 166-202, 1999.

OSTOVAR-NAMEGHI, Seyyed Ali; SHEIKHAHMADI, Mohsen. From Teacher Isolation to Teacher Collaboration: Theoretical Perspectives and Empirical Findings. **English Language Teaching**, v. 9, n. 5, p. 197-205, 2016. Disponível em: <http://dx.doi.org/10.5539/elt.v9n5p197>.

MENDOZA, Norman B.; CHENG, Eric C. K.; YAN, Zi. Assessing teachers' collaborative lesson planning practices: Instrument development and validation using the SECI knowledge-creation model. **Studies in Educational Evaluation**, v. 73, p. 101139, 2022. Disponível em: <https://doi.org/10.1016/j.stueduc.2022.101139>.

TAMMETS, Kairit; KHULBE, Manisha; SILLAT, Linda Helene; LEY, Tobias. A Digital Learning Ecosystem to Scaffold Teachers' Learning. **IEEE Transactions on Learning Technologies**, v. 15, n. 5, p. 619-632, out. 2022.

LIU, Shiqi; ZOU, Dandan. **Preliminary Study on Collaborative Lesson Planning based on Cloud Platform**. *Applied Mechanics and Materials*, v. 548-549, p. 1433-1437, 2014. Disponível em: <https://doi.org/10.4028/www.scientific.net/AMM.548-549.1433>.

SILVA, Carlos José Pereira da; GOMES, Alex Sandro; QUEIROS, Leandro Marques. **Inovaula: Ambiente Colaborativo de Aprendizagem para o Desenvolvimento Profissional de professores**. *Anais Estendidos do XIII Congresso Brasileiro de Informática na Educação (CBIE 2024)*, 2024. Disponível em: <https://repositorio.ufpe.br/handle/123456789/39519>

FERREIRA, Wellton Thiago Machado. **Evolução de modelo conceitual de sistema de informação para planejamento e compartilhamento de planos de aula**. 2015. Disponível em: <https://repositorio.ufpe.br/handle/123456789/39519>. Acesso em: 24 mar. 2025.

NONAKA, Ikujiro; TAKEUCHI, Hirotaka. **The knowledge-creating company: how Japanese companies create the dynamics of innovation**. 2. ed. New York: Oxford University Press, 1997.

GONZALEZ CAMACHO, Lesly Alejandra. **Sistema de recomendação baseado na relação de amizade entre usuários de redes sociais num cenário de cold-start**. 2020. 100 f. Dissertação (Mestrado em Engenharia de Computação) - Escola Politécnica, Universidade de São Paulo, São Paulo, 2020.

QUEIROS, Leandro Marques. Designing groupware for collaborative lesson planning for basic education teachers. 2022. 143 f. Tese (Doutorado em Ciência da Computação) – Universidade Federal de Pernambuco, Centro de Informática, Recife, 2022.

DENG, Shuiguang; HUANG, Longtao; XU, Guandong. Social network-based service recommendation with trust enhancement. **Expert Systems with Applications**, v. 41, n. 18, p. 8075-8084, 2014.

BENTZER, Carl; THULIN, Harry. **Recommender Systems Using Limited Dataset Sizes**. Stockholm: KTH Royal Institute of Technology, 2023. 41 p. Degree Project in Computer Science and Engineering, First cycle.

BOBADILLA, J.; ORTEGA, F.; HERNANDO, A.; GUTIÉRREZ, A. Recommender systems survey. **Knowledge-Based Systems**, v. 46, p. 109-132, 2013.

PEREIRA, Aluisio José. **Especificação de Sistema de Recomendação a partir de Análise de Interações em Rede Social Educacional**. 2021. 191 f. Dissertação (Mestrado em Ciência da Computação) – Centro de Informática, Universidade Federal de Pernambuco, Recife, 2021.

PEREIRA, Aluisio José; GOMES, Alex Sandro; PRIMO, Tiago Thompsen. **Especificação de Sistema de Recomendação Educacional de Incentivo às Interações em Plataforma Social de Aprendizagem**. **Revista Novas Tecnologias na Educação – RENOTE**, v. 20, n. 2, dez. 2022. Disponível em: <https://doi.org/10.22456/1679-1916.129143>.

PEREIRA, Aluisio José; GOMES, Alex Sandro; PRIMO, Tiago Thompsen. **Classificação de Interações com Indicadores de Engajamento dos Estudantes no Aprendizado On-line**. In: **XIII Congresso Brasileiro de Informática na Educação (CBIE 2024) e XXXV Simpósio Brasileiro de Informática na Educação (SBIE 2024)**, 2024. Anais [...]. Disponível em: <https://doi.org/10.5753/sbie.2024.242141>.

HALCOMB, Elizabeth; HICKMAN, Louise. Mixed methods research. *Nursing Standard*, London, v. 29, n. 32, p. 41–47, 2015.

PASINATO, Marden; MELLO, Carlos Eduardo; AUFAURE, Marie-Aude; ZIMBRÃO, Geraldo. **Generating synthetic data for context-aware recommender systems**. In: **BRICS CONGRESS ON COMPUTATIONAL INTELLIGENCE**, 11.; **BRAZILIAN CONGRESS ON COMPUTATIONAL INTELLIGENCE**, 11., 2013, Rio de Janeiro. **Anais [...]**. Rio de Janeiro: IEEE, 2013. p. 563-568.

WANG, Mengzhao; WU, Haotian; KE, Xiangyu; GAO, Yunjun; ZHU, Yifan; ZHOU, Wenchao. **Accelerating Graph Indexing for ANNS on Modern CPUs**. *arXiv*, [S. l.], p. 1-17, 2023. Disponível em: <https://arxiv.org/abs/2502.18113v1>.

GUO, Huifeng; TANG, Ruiming; YE, Yunming; LIU, Feng; ZHANG, Yuzhou. **An Adjustable Heat Conduction based KNN Approach for Session-based Recommendation.** *arXiv*, [S. l.], p. 1-11, 2018. Disponível em: <https://arxiv.org/abs/1807.05739v1>.

BAKER, Ryan S.; SIEMENS, George. Educational data mining and learning analytics. In: **DELFORGE, Roxane; GRAY, Elizabeth; HOWELL, Carla.** *Handbook of learning analytics*. 1. ed. [S.l.]: Routledge, 2014. p. 11-26. DOI: 10.1017/CBO9781139519526.016.

ZHANG, Shichao. **Challenges in KNN Classification.** *IEEE Transactions on Knowledge and Data Engineering*, v. 34, n. 10, p. 4663-4675, out. 2022. Disponível em: <https://doi.org/10.1109/TKDE.2022.3196341>. Acesso em: 18 abr. 2025.

WANG, Jun; DE VRIES, Arjen P.; REINDERS, Marcel J. T. **Unifying user-based and item-based collaborative filtering approaches by similarity fusion.** In: *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '06)*. New York: ACM, 2006. p. 501-508. DOI: 10.1145/1148170.1148257.

ABDOLLAHPOURI, Himan et al. *User-centered evaluation of popularity bias in recommender systems*. In: **Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization (UMAP '21)**, Utrecht, Netherlands, 2021. p. 119–129. DOI: [10.1145/3450613.3456821](https://doi.org/10.1145/3450613.3456821).