



SISTEMA DE RECOMENDAÇÃO DE ATIVIDADES

**Thiago M. SERNAGLIA¹; Luis F. M. FERREIRA²; Marcos V. ALMEIDA³ ; Rodrigo F. ALEIXO⁴;
João P. S. SILVA⁵; Rubens M. MASSA⁶; Douglas C. D. BRAZ⁷;**

RESUMO

A empresa "A Recreativa" desenvolveu uma plataforma educacional chamada MIA, cujo objetivo é auxiliar professores, desde a educação infantil até o 3º ano do ensino médio, no planejamento de suas aulas. Utilizando ferramentas de Inteligência Artificial, a plataforma gera documentos chamados "Sequências Didáticas" a partir de dados fornecidos pelo professor. A finalidade é acelerar e assistir o processo de criação de planos de aula. Nesse contexto, foi proposto o desenvolvimento de um sistema de recomendação para a plataforma. Embora a MIA também gera outros tipos de documentos, como o "Roteiro de Aprendizagem" e o "Plano de Aula", o sistema de recomendação foi focado exclusivamente na "Sequência Didática". Para isso, foram criados múltiplos algoritmos de recomendação, que utilizam diversas métricas de distância e similaridade. Além disso, foi implementado um filtro no banco de dados para garantir recomendações mais limpas e relevantes. Por fim, foi desenvolvida uma API para que "A Recreativa" possa integrar os algoritmos de recomendação à sua plataforma.

Palavras-chave:

Plataforma; Planejamento; Educação; Algoritmos; API.

1. INTRODUÇÃO

A plataforma MIA, da empresa "A Recreativa", visa auxiliar professores no planejamento de aulas através da geração de "Sequências Didáticas", o foco do sistema de recomendação é exclusivamente neste documento, com o objetivo de reaproveitar atividades já existentes e, conseqüentemente, reduzir os custos operacionais da empresa, além de auxiliar e incentivar o acesso a materiais de ensino de qualidade com baixo custo em escolas.

O desenvolvimento partiu de uma análise inicial de uma base de dados que continha diversos documentos de planejamento de aulas. Essa base passou por um rigoroso processo de tratamento, incluindo a separação dos dados pertinentes e uma limpeza extensiva para remover inconsistências e ruídos, resultando em um conjunto de dados preparado para o sistema.

Foi proposto um sistema que, ao final do preenchimento de um formulário pelo professor, recomendaria uma sequência didática similar à já existente no banco de dados. Para isso, diversas abordagens e algoritmos foram considerados e analisados, como Contagem de Palavras,

¹Bolsista FAPEMIG, IFSULDEMINAS – *Campus* Poços de Caldas. E-mail: thiago.sernaglia@alunos.ifsuldeminas.edu.br

²Bolsista FAPEMIG, IFSULDEMINAS – *Campus* Poços de Caldas. E-mail: luis.monteiro@alunos.ifsuldeminas.edu.br

³Bolsista FAPEMIG, IFSULDEMINAS – *Campus* Poços de Caldas. E-mail: marcos.almeida@alunos.ifsuldeminas.edu.br

⁴Bolsista FAPEMIG, IFSULDEMINAS – *Campus* Poços de Caldas. E-mail: rodrigo.aleixo@alunos.ifsuldeminas.edu.br

⁵Bolsista FAPEMIG, IFSULDEMINAS – *Campus* Poços de Caldas. E-mail: joaopaulosantos418@gmail.com

⁶Diretor e CEO, A Recreativa. E-mail: rubens@arecreativa.com.br

⁷Orientador, IFSULDEMINAS – *Campus* Poços de Caldas. E-mail: douglas.braz@ifsuldeminas.edu.br

Similaridade Híbrida por Peso, Similaridade Híbrida por Posição, PageRank e um modelo *Ensemble* que combina os métodos anteriores.

Após a definição e finalização dos algoritmos, foi construída uma API em Python utilizando o framework "Flask". Essa API foi desenvolvida para permitir que a empresa "A Recreativa" possa integrar o sistema de recomendação de forma simples e eficiente em sua plataforma já existente.

2. FUNDAMENTAÇÃO TEÓRICA

A teoria necessária para a criação dos sistemas de recomendação, teve base nos fundamentos dos sistemas de recomendação e seus sistemas de filtragem, dado pelo curso de pós-graduação do Marcelo Manzato (2022) sobre Sistemas de Recomendação e filtragem (colaborativa, baseada em conteúdo e híbrida), além de claro nas métricas de similaridade e distância, que são amplamente utilizadas como base para os sistemas de recomendação (Coseno, Euclidiana, Pearson, etc). Por estarmos lidando exclusivamente com texto, conceitos sobre vetorização de dados como BoW (Bag of Words), TF-IDF (Term Frequency - Inverse Document Frequency) se tornaram essenciais.

Para a teoria e lógica do Contador de Palavras, segundo (SOUZA, 2006) um dos modelos mais simples de Sistemas de Recuperação de Informações é o modelo booleano, no qual é feito o registro das palavras-chave de um texto em outro, a diferença aqui é que ao invés de registrar a ocorrência das palavras-chave como True ou False, é registrado o número de vezes que ela ocorre.

Embarak (2021) aborda o uso de sistemas de recomendação com aprendizado de máquina para personalizar e adaptar a educação. A ideia central é analisar o desempenho e as características dos alunos para sugerir-lhes, de forma automática, os recursos e caminhos de aprendizagem mais adequados. Visando otimizar a experiência educacional e os resultados individuais.

O PageRank (Langville & Meyer, 2006), originado para mecanismos de busca e baseado em cadeias de Markov, atribui uma pontuação de importância a nós em um grafo. Ele mede a relevância de um item pela probabilidade de ser "visitado" e pela importância dos nós que o "apontam".

3. MATERIAL E MÉTODOS

O projeto foi desenvolvido em Python, utilizando bibliotecas como Pandas, Scikit-learn, NLTK e Flask, após uma revisão bibliográfica inicial. O trabalho começou com um rigoroso pré-processamento dos dados, que incluiu a limpeza de duplicatas, inconsistências, caracteres especiais, problemas de acentuação e outras formatações para padronizar os textos.

Com os dados devidamente tratados e padronizados, a etapa subsequente foi o desenvolvimento de múltiplas abordagens para o sistema de recomendação. Iniciou-se com um algoritmo simples de contagem de palavras, evoluindo para um modelo que utiliza o PageRank para o reconhecimento de padrões e a detecção de textos com maior similaridade. Em paralelo a isso

foram empregados diversos algoritmos de *embedding*, como Word2Vec, GloVe, BERT com transformers e TF-IDF. Ademais, utilizaram-se cálculos de similaridade, incluindo a distância euclidiana, a similaridade de cosseno e uma média ponderada entre ambas as métricas.

Assim, foram gerados cinco modelos distintos de sistema de recomendação, cada um com características específicas: um modelo baseado em contador de palavras; um segundo com *embedding*, vetorização e PageRank; um terceiro com *embedding*, vetorização e um cálculo híbrido por posição; um quarto com *embedding*, vetorização e cálculo híbrido por peso. Todos os modelos passaram por um filtro customizado baseado em *meta-features* para refinar os resultados.

Por fim, a solução foi implementada como uma API com o framework Flask. A API recebe um JSON com a descrição do usuário e o modelo escolhido, processa a requisição e retorna as três atividades mais similares.

4. RESULTADOS E DISCUSSÃO

Utilizando uma base com todas as medidas de limpeza tomadas, sem usar o filtro de *meta-features*, os resultados se mostraram confiáveis e os quatro algoritmos individuais (Contador de palavras, Pagerank, vetorização e um cálculo híbrido por posição, cálculo híbrido por peso) alcançaram acurácias entre 35.21% e 41.50%. Mesmo quando falham em encontrar exatamente a atividade, os resultados das buscas são satisfatórios, retornando atividades bem similares a descrição dada.

A introdução de *meta-features* — como ano de aplicação e título — reduziu o espaço de busca das atividades e elevou substancialmente o desempenho. Ao retornar a opção mais similar, todos os modelos individualmente passaram a alcançar entre 74,42% e 82.86% de acurácia, chegando a 89,47%–96,02% quando utilizado as 3 atividades mais similares. A Tabela 1 contém os resultados de cada sistema separadamente.

Modelo	Acurácia Sem Meta-features	Acurácia Com Meta-features	Acurácia Com Meta-features 3 atividades mais similares
Similaridade Híbrida Por Peso	41.50%	82.86%	96.02%
Similaridade Híbrida Por Posição	40.13%	74.54%	92.96%
PageRank	40.21%	74.42%	92.53%
Contador de Palavras	35.21%	81.88%	89.47%

(Tabela 1 - Resultados dos modelos do Sistema de Recomendação)

5. CONCLUSÃO

Apesar do desafio de trabalhar com uma base de dados pequena e recente, o projeto foi bem-sucedido, entregando algoritmos de recomendação precisos e eficazes. O sucesso é creditado, em grande parte, à etapa fundamental de análise e limpeza dos dados.

A solução final, implementada através de uma API, oferece flexibilidade para a plataforma escolher o algoritmo mais adequado para cada situação. Mais importante, o sistema cumpre os objetivos centrais do projeto: incentivar o acesso a materiais de ensino de qualidade com baixo custo em escolas e a redução de custos.

AGRADECIMENTOS

Os autores agradecem o apoio financeiro da Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) por meio do processo nº APQ-01864-24, edital 07/2024, que tornou possível a realização desta pesquisa e a publicação deste trabalho.

REFERÊNCIAS

EMBARAK, Mohamed E. Applying machine learning in education for personalization and adaptation. *Education and Information Technologies*, v. 26, n. 4, p. 4317-4335, 2021.

GARCÍA-PEDRAJAS, N.; MUÑOZ, A.; ORTEGA, J. A. Ensemble methods for classification. *IEEE Transactions on Knowledge and Data Engineering*, v. 17, n. 4, p. 497-508, 2005.

LANGVILLE, Amy N.; MEYER, Carl D. *Google's PageRank and Beyond: The Science of Search Engine Rankings*. Princeton: Princeton University Press, 2006.

MANZATO, Marcelo. [2022] *Sistemas de Recomendação: YouTube*, 2022. 14 vídeos. Disponível em :<https://www.youtube.com/playlist?list=PLih3eXOECaZavYbOB6KiSG4iLuBoR8LwV>. Acesso em: jan. 2025.

SOUZA, Renato Rocha. *Sistemas de Recuperação de Informações e Mecanismos de Busca na web: panorama atual e tendências*. *Perspectivas em Ciência da Informação*, v. 11, n. 2, p. 161-173, maio/ago. 2006.