Revista Eletrônica de Iniciação Científica em Computação, 2026, 24:1
ISSN: 3085-8461 • doi: 10.5753/reic.2026.8494

Artigo de Pesquisa/Research Paper

Calibração de Sistemas de Recomendação com LLMs: Otimização de *Prompts* para Balancear Precisão, Diversidade e Justiça

Calibration of Recommendation Systems with LLMs: Prompt Optimization to Balance Accuracy, Diversity, and Fairness

Henrique Sekido [Universidade de São Paulo | *riqueysekio4@usp.br*]
Gabriel Prenassi [Universidade Federal de São João del-Rei | *prenassigabriel@aluno.ufsj.edu.br*]
Leonardo Rocha [Universidade Federal de São João del-Rei | *lcrocha@ufsj.edu.br*]
Marcelo G. Manzato [Universidade de São Paulo | *mmanzato@icmc.usp.br*]

Instituto De Ciências Matemáticas e de Computação, Universidade de São Paulo, Av. Trab. São Carlense, 400 - Centro, São Carlos - SP, 13566-590, Brasil.

Resumo. Sistemas de Recomendação (SsR) são essenciais em plataformas digitais, mas enfrentam problemas como o viés de popularidade, que reduz diversidade e justiça. Técnicas de calibração buscam alinhar recomendações às preferências do usuário, geralmente por pós-processamento. Com o surgimento de Modelos de Linguagem de Grande Escala (LLMs), como o Llama, a engenharia de prompts surge como alternativa para personalização. Este estudo investiga a calibração baseada em LLMs e a compara a métodos tradicionais. Também avaliamos diferentes estratégias de prompts por métricas de acurácia, diversidade e justiça. Os resultados indicam que LLMs podem melhorar simultaneamente personalização e equidade nos sistemas de recomendação.

Abstract. Recommender Systems (RSs) are essential in digital platforms but face challenges such as popularity bias, which reduces diversity and fairness. Calibration techniques aim to better align recommendations with user preferences, typically through post-processing. With the emergence of Large Language Models (LLMs), such as Llama, prompt engineering has become an alternative approach to personalization. This study investigates LLM-based calibration and compares it with traditional methods. We also evaluate different prompting strategies using metrics that encompass accuracy, diversity, and fairness. The results indicate that LLM-based prompting strategies can simultaneously improve personalization and fairness in recommender systems.

Palavras-chave: Sistemas de Recomendação, LLM, Engenharia de *Prompts*, Viés de Popularidade

Keywords: Recommender Systems, LLM, Prompt Engineering, Popularity Bias

Recebido/Received: 16 June 2026 • **Aceito/Accepted:** 16 June 2026 • **Publicado/Published:** 10 July 2026

1 Introdução

Sistemas de Recomendação (SsR) estão amplamente presentes em plataformas digitais, auxiliando na sugestão de produtos, locais e músicas, com o objetivo de oferecer conteúdo relevante e personalizado [Quadrana *et al*., 2018; Werneck *et al*., 2020]. Apesar dos avanços em aprendizado de máquina, desafios como o viés de popularidade ainda persistem, favorecendo itens amplamente consumidos e reduzindo a diversidade das recomendações [Klimashevskaia *et al*., 2024]. Esse problema impacta não apenas a satisfação dos usuários, mas também a descoberta de novos conteúdos. Diante desse cenário, a calibração surge como uma estratégia para mitigar esses efeitos, buscando alinhar as recomendações ao perfil real do usuário. Tradicionalmente aplicada como etapa de pós-processamento, essa abordagem utiliza técnicas como reponderação e interpolação de *rankings* para equilibrar categorias ou níveis de popularidade [Atauchi *et al*., 2025].

Nesse cenário, os Grandes Modelos de Linguagem (LLMs) têm sido explorados como alternativa complementar às estratégias tradicionais de calibração em SsR, por meio de engenharia de *prompts* e de cadeias de pensamento [Gao *et al*., 2025], além de estratégias de otimização em linguagem natural para controlar recomendações [Ortega *et al*., 2024]. Estudos recentes indicam que a otimização de *prompts* pode gerar ganhos expressivos em métricas de *ranking*, alcançando melhorias de até 20% em NDCG@10 [Wang *et al*., 2025]. Ainda assim, a maioria desses trabalhos concentra-se predominantemente na precisão, deixando em segundo plano dimensões como diversidade, cobertura, justiça e viés de popularidade. Além disso, embora existam comparações entre abordagens tradicionais de calibração e versões baseadas em LLMs [Ortega *et al*., 2024], tais investigações não exploram sistematicamente diferentes estratégias de *prompting*.

Diante dessas limitações, investigamos o potencial dos LLMs na calibração de SsR sob uma perspectiva multivariada, analisando simultaneamente múltiplas métricas de avaliação e o impacto da otimização de *prompts*. Propomos uma abordagem em duas etapas: (i) comparar métodos tradicionais e LLMs na tarefa de recomendação e calibração; e (ii) examinar como distintas estratégias de *prompting* influenciam os resultados. Nesse contexto, o estudo é guiado pelas seguintes questões de pesquisa: (**RQ1**) O uso de estratégias baseadas em LLMs proporciona ganhos em relação

a métodos tradicionais de recomendação, quanto a métricas como precisão, diversidade, cobertura e justiça? (**RQ2**) Quanto à otimização de *prompts* de estratégias baseadas em LLMs pode aprimorar a qualidade das recomendações em termos de precisão, diversidade, cobertura e justiça?

Para responder a essas questões, comparamos métodos tradicionais de calibração com estratégias baseadas em LLMs, avaliando diferentes estratégias de *prompting* sob múltiplas dimensões da qualidade das recomendações, incluindo precisão, justiça, cobertura e viés de popularidade. Adicionalmente, empregamos uma métrica agregada baseada na Teoria da Utilidade Multiatributo (*Multi-Attribute Utility Theory* - MAUT) [Carvalho and Rocha, 2020], permitindo analisar de forma integrada os *trade-offs* entre esses critérios. De modo geral, os resultados evidenciam que estratégias baseadas em LLMs alcançam ganhos consistentes em precisão e promovem melhor equilíbrio entre as demais dimensões avaliadas quando contrastadas com abordagens tradicionais. Por outro lado, a otimização de *prompts* envolve compensações que devem ser consideradas de acordo com os objetivos específicos de cada aplicação. Esse trabalho de iniciação científica resultou na publicação de um artigo no WebMedia 2025 (A3) [Prenassi *et al*., 2025].

2 Trabalhos Relacionados

O viés de popularidade e a calibração em SsR têm sido amplamente investigados devido ao seu impacto na diversidade, justiça e satisfação do usuário. Um marco na área é o trabalho de [Steck, 2018], que propôs um método de pós-processamento para alinhar a distribuição dos itens recomendados às preferências históricas do usuário, especialmente em relação a gêneros. Posteriormente, [Abdollahpouri *et al*., 2021] classificaram usuários segundo sua sensibilidade à popularidade (nicho, diverso e *blockbuster*), adaptando as recomendações a esses perfis. Outras abordagens passaram a integrar modelos preditivos com técnicas de calibração: [Sacilotti *et al*., 2023] combinaram predição e pós-processamento para reduzir o viés mantendo relevância, enquanto [de Souza and Manzato, 2024] incorporaram mecanismos de calibração diretamente ao treinamento ao modificar o BPR (*Bayesian Personalized Ranking*) [Rendle *et al*., 2012]. Em comum, tais estratégias concentram-se majoritariamente em ajustes numéricos de escores e *rankings*. Mais recentemente, com a incorporação de LLMs, novas possibilidades passaram a emergir. Trabalhos como [Liu *et al*., 2023] exploram engenharia de *prompts* para recomendação, enquanto revisões como [Wu *et al*., 2024] sistematizam paradigmas de recomendação generativa. Evidências indicam que LLMs podem reduzir o viés de popularidade em cenários *zero-shot* [Lichtenberg *et al*., 2024], e que a calibração via *prompting* pode aprimorar relevância e personalização [Zhao *et al*., 2021].

Apesar dos avanços observados tanto nas abordagens tradicionais quanto nas estratégias baseadas em LLMs, ainda são limitadas as análises que investigam, de forma integrada, o impacto de diferentes estratégias de *prompting* sob múltiplas dimensões da qualidade das recomendações. Em especial, carecem estudos que comparem sistematicamente métodos clássicos de calibração com abordagens baseadas em LLMs considerando simultaneamente precisão, justiça, cobertura e viés de popularidade. Nesse contexto, **este trabalho propõe uma análise comparativa e multicritério abrangente, examinando como estratégias de otimização de *prompts* influenciam a calibração em SsR baseados em LLMs**.

3 Recomendação Baseada em LLM

3.1 Configuração do Modelo e *Pipeline* de Recomendação

Em nossa proposta de recomendação com LLM, adotamos o *Llama3.1-8b-instruct*, quantizado em 4 *bits*, reduzindo o consumo de memória sem prejuízo de desempenho [Hu *et al*., 2021]. O modelo foi escolhido por ser aberto e amplamente utilizado na literatura [Fonseca *et al*., 2025]. As recomendações são geradas por meio do *prompt* da Figura 1, composto por um `System Prompt`, que define o contexto da tarefa, e um `User Prompt`, instanciado com os 20 itens consumidos pelo usuário no conjunto de treino. A geração foi configurada com limite de 1000 *tokens*, *temperature* 0.7 e *top-p* 0.9, mantendo *use_cache* ativado para otimizar a inferência. O limite de *tokens* foi definido para evitar truncamentos na geração das dez recomendações solicitadas, enquanto os valores de *temperature* e *top-p* foram adotados para equilibrar a diversidade, controlar a geração e garantir aderência ao formato esperado.

No `User Prompt`, a restrição de que as recomendações sejam baseadas em membros do elenco foi adotada como uma escolha experimental controlada, por se tratar de uma informação de conteúdo diretamente observável e relevante no domínio de filmes. Essa decisão permite orientar o LLM por um critério específico, reduzindo a ambiguidade da tarefa e tornando as recomendações mais comparáveis entre execuções diferentes. Embora essa restrição introduza um viés de conteúdo no processo de recomendação, ela foi mantida fixa em todos os experimentos, permitindo avaliar as estratégias sob a mesma condição experimental.

***Prompt* Inicial**

System Prompt:
Given movies/tv shows titles, provide the recommendations following pattern:
1. title (release year)
2. title (release year)
...

User Prompt:
I need exactly 10 movies or TV shows (based on the MOVIELENS 1M dataset) and your recommendations must be based on cast members.
Follow this strict format and do not include any explanations, duplicates, or corrections in your response:
1. title (release year)
2. title (release year)
3. title (release year)
...
I've watched:
{movies}

Figura 1. *Prompt* para gerar recomendações. O campo `{movies}` é substituído pelos filmes assistidos pelo usuário.

Devido à natureza livre das respostas em linguagem natural geradas pelo LLM, implementamos um processo de normalização e filtragem para garantir consistência com o conjunto de dados (MovieLens). Diferenças na formatação de títulos, como "Godfather, The (1972)" e "The Godfather (1972)", foram tratadas por meio de uma etapa de normalização, evitando inconsistências na validação. Para pequenas discrepâncias ortográficas, utilizamos a similaridade baseada

na distância de *Levenshtein*, o que permite reconhecer itens semanticamente equivalentes. Ademais, recomendações duplicadas ou inexistentes na coleção foram descartadas, assegurando a validade das listas geradas. O procedimento é repetido até obter dez recomendações válidas por usuário ou até atingir o limite de cinco tentativas consecutivas sem sucesso. Por fim, as recomendações finais são comparadas ao conjunto de teste individual para avaliação do desempenho.

3.2 Processo de Otimização de *Prompt*

Considerando a alta sensibilidade dos LLMs ao *prompt*, adotamos uma adaptação do método *OPRO* (***O**ptimization by **PRO**mpting* [Yang *et al*., 2023]), no qual o modelo atua como otimizador iterativo das instruções do campo `system` do *prompt*, isto é, o contexto fornecido. Partimos de um *prompt* inicial, cuja qualidade é avaliada por uma métrica previamente definida, e construímos um *meta-prompt* que descreve a tarefa, incorpora o histórico de instruções testadas com seus desempenhos e solicita novas versões. O conteúdo do campo `user` permanece fixo, enquanto as instruções geradas são avaliadas e comparadas. O procedimento, conduzido pelo *Llama3.1-8b-instruct* quantizado em 4 *bits*, é executado por seis iterações, nas quais o modelo propõe quatro novas instruções que competem com as três melhores existentes. As três de melhor desempenho passam a compor o *meta-prompt* a seguir, promovendo refinamento progressivo. A geração foi configurada para produzir até 1000 *tokens*, com amostragem ativada e parâmetros *temperature* 1.6, *top-p* 0.9 e *top-k* 40. Diferentemente da etapa de geração das recomendações finais, essa configuração foi adotada para favorecer maior exploração na criação de novas instruções candidatas. O valor mais alto de *temperature* aumenta a diversidade das propostas geradas, enquanto *top-p* e *top-k* restringem o espaço de amostragem, buscando evitar instruções excessivamente aleatórias ou pouco aderentes à tarefa. Ao final, a instrução selecionada ocupa o campo `system` do *prompt*, orientando a geração de recomendações.

A Figura 2 apresenta a estrutura do *meta-prompt* utilizado no processo de otimização. Esse *meta-prompt* orienta o modelo a gerar novas instruções candidatas para o campo `system`, considerando a descrição da tarefa, o histórico de instruções previamente avaliadas e seus respectivos desempenhos.

Como critério de avaliação, consideramos duas estratégias distintas, a maximização de *MAP* ou de *MAUT* [Carvalho and Rocha, 2020]. Esta última agrega, em uma única medida composta, os indicadores *NDCG*, *MAP*, *F1-score*, *LTC* e *RMSE*, todos detalhados na Seção 4.3. No primeiro caso, ao final do processo, seleciona-se a instrução com maior *MAP*. No segundo, a medida composta é recalculada a cada etapa com base apenas nas sete instruções comparadas, isto é, nas três melhores da rodada anterior e nas quatro recém-geradas, pois seu valor depende do conjunto corrente sob avaliação. Como essa medida não é diretamente comparável entre etapas distintas, seleciona-se a instrução com maior valor na etapa em que atinge seu máximo.

Adicionalmente, conduzimos uma avaliação preliminar para analisar o impacto do perfil dos usuários no processo de otimização. Em uma configuração, denominada ao longo do artigo como *random*, selecionamos aleatoriamente 100 usuários, dos quais 89 já obtinham 10 recomendações

System Prompt

Your task is to create a clear and effective instruction for a movie recommendation task using the MovieLens 1M dataset. The instruction you generate will be used by a model to recommend movies based on user preferences, past ratings, or relevant contextual signals such as genre affinity, viewing history, or similar user behavior. To help you write a stronger instruction, we provide examples of previous instructions along with quality scores. These are ordered from worst to best in terms of recommendation relevance, clarity, and alignment with the dataset. Use them as inspiration, but do not copy them.
{instructions}
Your instruction must explicitly include the following constraints:
- The output must be a numbered list in this exact format:
  1. title (release year)
  2. title (release year)
  3. title (release year)
  ...
- The model must output **only** the list — no additional text, explanations, or commentary.
To maximize recommendation quality:
- Encourage the model to consider patterns in user preferences, ratings, and genres.
- Make it clear that the recommendations should be personalized and relevant to the user context.
- Ensure that the instruction clearly connects the task to the MovieLens 1M dataset (which includes user ratings, genres, and timestamps).
- Encourage the model to suggest items that are both relevant and meaningfully ordered according to the user's preferences.
- Balance recommendations to not overly favor the most popular items, promoting a more diverse and personalized experience.
- Ensure that the instruction leads the model to capture deeper patterns in user behavior, including preferences across popularity levels and genre variety.
- The goal is to produce recommendations that are not only accurate, but also calibrated, diverse, and well-aligned with each user's unique profile.
Focus on crafting an instruction that is practical, specific, and optimized for generating relevant, high-quality recommendations in the correct format.
Focus on crafting an instruction that is practical, specific, and optimized for generating high-quality recommendations in the correct format.

User Prompt

Create a new instruction that is concise and highly effective for the recommendation task using the MovieLens 1M dataset.
The instruction must explicitly state that the model's response should consist only of a numbered list of movie titles in the format:
1. title (release year)
2. title (release year)
3. title (release year)
...
It must also make clear that no additional text, explanations, or extra information should be included in the output.
Ensure that the instruction is presented directly, without mentioning that it is being generated or evaluated, and do not include any reference to performance metrics or scoring.

Figura 2. *Meta-prompt* utilizado no processo de otimização. O `System Prompt` define a tarefa do otimizador e orienta a criação de novas instruções candidatas, enquanto o `User Prompt` solicita a geração direta de uma nova instrução para recomendação. Os trechos específicos para as métricas *MAP* (em azul) e *MAUT* (em laranja) são destacados. O campo `{instructions}` é substituído, em cada iteração, pelas melhores instruções geradas até o momento e suas respectivas pontuações.

com o *prompt* inicial. Na outra, denominada *below_10*, selecionamos 100 usuários que não atingiam esse mínimo. Essa diferenciação permitiu investigar como a métrica adotada e o perfil dos usuários influenciam o resultado final. Ao todo, realizamos quatro execuções combinando métricas e perfis distintos, resultando nas variantes *MAP_random*, *MAP_below_10*, *MAUT_random* e *MAUT_below_10*, posteriormente analisadas na Seção 5.

4 Ambiente Experimental

4.1 Conjunto de Dados

Utilizamos o MovieLens 1M [Harper and Konstan, 2015], originalmente composto por 1.000.209 avaliações de 6.040 usuários sobre cerca de 3.900 filmes. Conforme práticas da literatura [de Souza and Manzato, 2024], removemos usuários com menos de 30 avaliações, assegurando volume

adequado de dados por perfil. Para cada usuário restante, consideramos as 30 interações mais recentes, sendo 20 destinadas ao treino e 10 ao teste. Além disso, avaliações de teste associadas a filmes ausentes no treino foram removidas para manter a coerência experimental. Ao final, o conjunto reúne 158.446 avaliações (105.780 em treino e 52.666 em teste), distribuídas entre 5.289 usuários e 3.432 filmes.

4.2 Métodos Tradicionais

Para avaliar nossos modelos, selecionamos *baselines* que abrangem diferentes abordagens de recomendação. Consideramos o **Popularity** [Sacilotti *et al*., 2023], que calibra as recomendações com base na popularidade dos itens e no interesse do usuário por esse aspecto, e o **Personalized** [Sacilotti *et al*., 2023], que ajusta a lista recomendada conforme a preferência individual por popularidade ou gênero. Incluímos também o método clássico de calibração por conteúdo de **Steck** [Steck, 2018], que alinha a distribuição de gêneros do ranking ao perfil do usuário, bem como a abordagem **Two-stage** [Souza and Manzato, 2024], que combina calibração por popularidade e por gênero em duas etapas. Adotamos ainda o método de **Abdollahpouri** [Abdollahpouri *et al*., 2021], voltado à mitigação do viés de popularidade sob uma perspectiva centrada no usuário, empregando uma medida de divergência distinta da utilizada no *Popularity*. Como referência de ranqueamento tradicional, utilizamos o **BPR** [Rendle *et al*., 2012], modelo clássico de *pairwise ranking* para *feedback* implícito, e sua extensão **BPR Calibrado** [de Souza and Manzato, 2024], que incorpora um mecanismo explícito de calibração de popularidade.

4.3 Métricas de Avaliação

Em nossos experimentos, avaliamos os efeitos de diferentes aspectos das recomendações. Em termos de **precisão**, utilizamos o *Normalized Discounted Cumulative Gain* (NDCG) e o *Mean Average Precision* (MAP). O NDCG compara o ganho cumulativo descontado do *ranking* gerado com o de um *ranking* ideal, penalizando itens relevantes posicionados em níveis inferiores da lista. Já o MAP mede a precisão média acumulada ao longo das posições recomendadas, refletindo a qualidade global da ordenação. Ambas variam em $[0, 1]$, sendo valores mais altos indicativos de melhor desempenho.

Para complementar a análise, incorporamos critérios de **justiça**. Adotamos o *Mean Rank Miscalibration* (MRMC) [da Silva *et al*., 2021], que mede a divergência entre a distribuição de atributos no perfil do usuário e na lista recomendada, com base na divergência de Kullback-Leibler normalizada. A métrica agrega o erro de calibração ao longo das posições do *ranking* e entre os usuários, assumindo valores em $[0, 1]$, em que menores valores indicam melhor alinhamento. Consideramos dois atributos associados à miscalibração: gênero e popularidade dos itens (nicho, diverso e *blockbuster*, conforme o princípio de Pareto [Abdollahpouri *et al*., 2021]). Seguindo [de Souza and Manzato, 2024], combinamos $(1 - MRMC_{Genre})$ e $(1 - MRMC_{Pop})$ por meio da média harmônica (F1), de modo que valores mais altos representem melhor calibração global.

Também analisamos a **cobertura**, avaliada por meio da *Long-Tail Coverage* (LTC), que quantifica a proporção de itens de nicho efetivamente expostos nas listas recomendadas, variando em $[0, 1]$, com valores maiores indicando maior exposição da cauda longa. Complementarmente, examinamos o **viés de popularidade**, mensurado pelo *Group Average Popularity* (ΔGAP) [Abdollahpouri *et al*., 2021], que captura o desvio entre a popularidade média dos itens recomendados e aquela observada no perfil do usuário. Como o valor ideal é $\Delta GAP \approx 0$, empregamos o *Root Mean Squared Error* (RMSE) entre os três grupos de popularidade (*Blockbuster*, Nicho e Diverso) para sintetizar o desvio global [Souza and Manzato, 2024], sendo valores menores indicativos de menor viés. Por fim, integramos essas dimensões por meio da **MAUT** [Carvalho and Rocha, 2020]. As métricas MAP, NDCG, RMSE, LTC e F1 são normalizadas via *min-max scaling*, com inversão quando necessário, e combinadas com pesos iguais para compor a utilidade global $U_i \in [0, 1]$. Valores mais altos de U_i indicam melhor capacidade do método em equilibrar simultaneamente os aspectos analisados pelas métricas, refletindo diretamente os *trade-offs* centrais deste trabalho.

Para assegurar robustez experimental, todos os métodos foram executados com seis repetições independentes, e as diferenças observadas foram avaliadas por meio do teste estatístico não paramétrico de Wilcoxon [Rey and Neuhäuser, 2011], adotando nível de significância de 5%.

5 Resultados

5.1 LLM *versus* Métodos Tradicionais

Para responder à **RQ1**, que investiga em que medida estratégias baseadas em LLM superam métodos tradicionais em termos de precisão, diversidade, cobertura e justiça, comparamos os modelos sob múltiplas métricas, conforme apresentado na Tabela 1. As diferenças observadas entre os métodos são estatisticamente comparáveis, de acordo com o teste estatístico aplicado. Ressaltamos que os resultados do LLM aqui reportados referem-se à versão base do modelo, utilizando o *prompt* inicial, sem aplicação do processo de otimização descrito na Seção 3.2.

Em termos de **precisão** (MAP@10 e NDCG@10), a estratégia baseada em LLM superou consistentemente os métodos tradicionais. O MAP@10 variou entre 0,008 e 0,037 nos métodos clássicos, enquanto o LLM atingiu 0,059. Para o NDCG@10, os valores oscilaram entre 0,003 e 0,013, ao passo que o LLM alcançou 0,021. Esses resultados evidenciam maior capacidade de posicionar itens relevantes nas primeiras posições da lista de recomendação. Quanto à **diversidade** e **cobertura**, o LLM apresentou LTC igual a 0,517. Embora BPR e BPR Calibrado tenham alcançado valores superiores nessa métrica, o LLM demonstrou melhor equilíbrio no *trade-off* entre precisão e diversidade [Zanon *et al*., 2022], reduzindo a concentração em itens populares e ampliando a exposição à cauda longa.

No que se refere ao **viés de popularidade**, sintetizado pelo RMSE entre grupos (quanto menor, melhor), o modelo baseado em LLM superou a maioria dos métodos tradicionais, cujos valores variaram consideravelmente. Embora abordagens como o BPR Calibrado e o Popularity tenham apresentado RMSE menores, esse desempenho ocorreu à custa de perdas relevantes em outras dimensões. O BPR Calibrado, por exemplo, registrou os menores valores de MAP@10 e de NDCG@10, enquanto o Popularity apresentou cobertura reduzida (LTC) e baixa precisão, refletida

Modelo	MAP@10	NDCG@10	LTC	F1 Score	RMSE	MAUT
Métodos Tradicionais						
Popularity [Sacilotti *et al*., 2023]	0,027	0,010	0,048	**0,539**	0,210	0,511
Personalized [Sacilotti *et al*., 2023]	0,035	0,013	0,080	0,257	1,498	0,287
Steck [Steck, 2018]	0,037	0,013	0,075	0,243	1,130	0,320
Two-stage [Souza and Manzato, 2024]	0,034	0,013	0,080	0,266	0,960	0,342
Abdollahpouri [Abdollahpouri *et al*., 2021]	0,026	0,010	0,046	0,537	1,115	0,412
BPR [Rendle *et al*., 2012]	0,008	0,003	**0,542**	0,502	0,623	0,472
BPR Calibrado [de Souza and Manzato, 2024]	0,008	0,003	0,530	0,499	**0,044**	0,527
Modelo baseado em LLM (Llama)						
Sem otimização	**0,059**	**0,021**	0,517	0,296	0,367	**0,741**

Tabela 1. Comparação entre modelos tradicionais e o modelo baseado em LLM (Llama). Os melhores valores por métrica estão em negrito. Todos os resultados apresentaram significância estatística (p-*value* $< 0{,}05$), conforme o teste de Wilcoxon.

em seus valores de MAP@10. Em contraste, o modelo manteve um RMSE competitivo sem comprometer a precisão do ranqueamento (MAP@10 e NDCG@10), a diversidade (LTC) ou o equilíbrio global entre métricas, resultado também evidenciado por seu elevado valor de MAUT.

No aspecto da **justiça** (F1 Score), o modelo baseado em LLM apresentou desempenho inferior aos métodos tradicionais, com valor de 0,296, enquanto os demais variaram entre 0,243 e 0,539. Esse resultado evidencia o conhecido *trade-off* entre diversidade e justiça [Zhao *et al*., 2025]. Ainda assim, ao considerar a métrica agregada MAUT, que integra precisão, diversidade e justiça, o modelo alcançou 0,741, superando os métodos tradicionais (0,287 a 0,527) e indicando um melhor equilíbrio global entre as dimensões avaliadas.

Assim, em resposta à **RQ1**, os resultados indicam que **estratégias baseadas em LLM superam os métodos tradicionais em precisão e diversidade, mantêm desempenho competitivo em justiça e apresentam melhor equilíbrio global ao considerar múltiplos objetivos simultaneamente.**

5.2 Efeitos da Otimização de *Prompts*

5.2.1 Análise dos Prompts Otimizados

Além da avaliação quantitativa, analisamos qualitativamente os `System Prompts` finais gerados pelo processo de otimização, buscando identificar padrões textuais emergentes e diferenças semânticas entre as instruções obtidas a partir das métricas MAP e MAUT.

Dessa forma, foram gerados quatro cenários principais:

- ***MAP_random***: otimização por MAP com usuários aleatórios (Figura 3);
- ***MAP_below_10***: otimização por MAP com usuários de baixo histórico (Figura 4);
- ***MAUT_random***: otimização por MAUT com usuários aleatórios (Figura 5);
- ***MAUT_below_10***: otimização por MAUT com usuários de baixo histórico (Figura 6).

De modo geral, todos os *prompts* otimizados reforçam a padronização da saída, solicitando listas numeradas com título e ano de lançamento e restringindo a presença de explicações adicionais. Esse padrão textual é relevante porque reduz ambiguidades na geração e facilita as etapas posteriores de normalização, filtragem e validação das recomendações. Ao comparar as métricas de otimização, observa-se que os *prompts* gerados por MAP tendem a enfatizar a personalização e a aderência ao formato esperado, mencionando sinais como avaliações, gêneros, preferências e pistas contextuais do conjunto de dados. Já os *prompts* gerados por MAUT apresentam maior proximidade com objetivos multiatributo, especialmente no caso de *MAUT_random*, que explicita termos associados à calibração, à diversidade de gêneros, ao alinhamento ao perfil do usuário e à afinidade de gênero.

Assim, a análise qualitativa sugere que a métrica utilizada na otimização influencia não apenas os resultados quantitativos, mas também o conteúdo semântico das instruções geradas. Enquanto MAP favorece instruções mais voltadas à precisão e personalização, MAUT tende a induzir descrições mais alinhadas ao equilíbrio entre relevância, diversidade e calibração. Como o `User Prompt` permaneceu fixo em todos os experimentos, essas diferenças podem ser atribuídas à otimização do `System Prompt`.

`System Prompt`

Generate Personalized Movie Recommendations Using the MovieLens 1M Dataset

Use the MovieLens 1M dataset to generate a list of personalized movie recommendations based on the user's context (if provided). This list should consider patterns in user ratings, genres, and timestamps to create unique and relevant suggestions tailored to each viewer.

Your response should be a numbered list in the exact format:

1. Title (release year)
2. Title (release year)
3. Title (release year)
...

This list **must consist only of**:

* Numbered movie titles with release years in the specified format
* Exactly one title per entry
* Release years included for each title

Do not include:

* Any additional text, explanations, or comments in your response
* Extra information or unnecessary details about the movies or the users
* Non-movie titles, incomplete, or missing metadata

Provide your output exactly as specified above, following the specified format for each entry.

Figura 3. `System Prompt` gerado pelo processo de otimização de *prompt* com a métrica *MAP*, utilizando 100 usuários aleatórios.

Modelo	MAP@10	NDCG@10	LTC	F1 Score	RMSE	MAUT
Modelos LLM (Llama) com otimização						
MAP_below_10	0,056 ▼	0,019 ▼	**0,592** ▲	0,292 ●	**0,350** ●	0,392 ▼
MAP_random	0,057 ●	**0,021** ●	0,512 ●	0,300 ●	0,490 ▼	0,458 ●
MAUT_below_10	0,055 ▼	0,020 ●	0,583 ▲	0,297 ●	0,406 ●	0,445 ●
MAUT_random	0,055 ●	0,020 ●	0,486 ▼	**0,302** ●	0,433 ▼	0,401 ●
Modelo LLM (Llama) sem otimização						
Sem otimização	**0,059**	**0,021**	0,517	0,296	0,367	**0,518**

Tabela 2. Comparação entre o modelo LLM (Llama) sem otimização e quatro variações com diferentes estratégias de otimização. Os melhores valores em cada métrica estão em negrito. O símbolo ▲ indica que a otimização apresentou melhora estatisticamente significativa em relação ao modelo sem otimização (p-*value* < 0.05, teste de Wilcoxon); ● indica ausência de diferença significativa; e ▼ indica que o modelo sem otimização foi estatisticamente superior.

`System Prompt`

Recommend Movies from the MovieLens 1M Dataset.

Given the MovieLens 1M dataset and a relevant user context, output a **complete** list of personally recommended movies in the following format:

1. title (release year)
2. title (release year)
3. title (release year)
...

Your response should consist **solely** of this numbered list. Please adhere strictly to the specified format.

You may draw upon patterns in user ratings, genres, preferences, and contextual clues from the dataset to generate your recommendations.

Output the recommendation list exactly as instructed; ensure that it includes the title and release year for each movie.

Figura 4. `System Prompt` gerado pelo processo de otimização de *prompt* com a métrica *MAP*, utilizando 100 usuários com menos de 10 recomendações.

5.2.2 Análise de Desempenho do LLM sob Prompts Otimizados

Para responder à **RQ2**, que investiga em que medida a otimização de *prompts* pode aprimorar a qualidade das recomendações em termos de precisão, diversidade, cobertura e justiça, analisamos quatro variações do modelo com diferentes estratégias de otimização, conforme descrito na Seção 3.2, comparando-as ao modelo antes da aplicação do processo de otimização (Tabela 2).

Em termos de **precisão**, tomando o modelo sem otimização como referência (MAP@10 = 0,059; NDCG@10 = 0,021), as estratégias *MAP_random* e *MAUT_random* mantiveram desempenho estatisticamente equivalente, enquanto *MAP_below_10* e *MAUT_below_10* apresentaram redução significativa em MAP@10, sendo que apenas *MAP_below_10* impactou negativamente o NDCG@10. No que se refere ao **viés de popularidade** (RMSE), as variantes *random* aumentaram significativamente o erro entre grupos, ao passo que as estratégias *below_10* mantiveram equivalência estatística com o modelo base, preservando o equilíbrio distributivo.

Quanto à **diversidade** (LTC), observou-se o efeito mais consistente da otimização: *MAP_below_10* e *MAUT_below_10* elevaram significativamente o LTC (0,592 e 0,583), ampliando a exposição à cauda longa. Por outro lado, *MAP_random* manteve a equivalência estatística, enquanto *MAUT_random* apresentou redução. Esse resultado pode

`System Prompt`

Personalized Movie Recommendation

Your task is to generate a list of movie recommendations based on user preferences and past ratings from the MovieLens 1M dataset. Each movie should be accompanied by its release year. Follow these rules exactly:

* Your response should consist only of a numbered list of movie titles, one movie per line, in the exact format:
1. title (release year)
2. title (release year)
3. title (release year)
...

* The list should contain multiple movie entries; there is no minimum or maximum length for the list.
* Leave no blank lines between the list entries.
* Do not include any additional text, headings, or information in your response; the list should be the only thing present.
* Each title should be a string enclosed in parentheses, indicating its release year, or left uncaptioned but present only as part of a title. For example:
1. "Inception (2010)"
2. "The Dark Knight (2008)"

* Provide movie titles that accurately reflect the preferences and habits of the users in the MovieLens 1M dataset.

Ensure your recommendations are calibrated and balance diverse film genres, as well as offer unique profiles that account for relevance, alignment with user profiles, and genre affinity.

Include only these necessary items:

1. title (release year)
2. title (release year)
3. title (release year)
...

Your response will take the form of a numbered list with the movie titles presented in the above format, offering a set of personalized and well-calibrated movie suggestions for viewers interested in a varied selection of films from the MovieLens dataset.

Figura 5. `System Prompt` gerado pelo processo de otimização de *prompt* com a métrica *MAUT*, utilizando 100 usuários aleatórios.

estar associado ao fato de que a configuração *below_10* reúne usuários para os quais o *prompt* inicial teve maior dificuldade em produzir uma lista completa de recomendações válidas. Esses casos podem envolver históricos de consumo mais heterogêneos, menos populares ou mais difíceis de capturar por instruções genéricas. Assim, ao otimizar o *prompt* sobre esse subconjunto, o processo pode ter favorecido instruções mais exploratórias, capazes de ampliar o conjunto de itens considerados e, consequentemente, aumentar a exposição a itens de cauda longa.

Para **justiça** (F1), não houve diferenças estatisticamente significativas entre as variantes. Na métrica agregada MAUT,

`System Prompt`

Recommend Movie Titles Based on User Behavior

Respond with a numbered list of recommended movie titles and release years.

The format for the response is:

1. title (release year)
2. title (release year)
3. title (release year)
...

Include the exact movie title and release year for each item.

The response must be a standalone numbered list of movie titles and dates. Provide only this list - do not include any additional text or information.

Figura 6. `System Prompt` gerado pelo processo de otimização de *prompt* com a métrica *MAUT*, utilizando 100 usuários com menos de 10 recomendações.

apenas *MAP_below_10* apresentou queda significativa, enquanto as demais estratégias mantiveram desempenho equivalente ao do modelo não otimizado.

Assim, em resposta à **RQ2**, os resultados indicam que **a otimização de *prompts* não gera melhorias globais automáticas, mas funciona como mecanismo de direcionamento do comportamento do modelo. Estratégias *below_10* favorecem a diversidade e preservam o equilíbrio entre grupos, com leve perda de precisão, enquanto estratégias *random* mantêm precisão, porém tendem a ampliar as discrepâncias distributivas. Assim, sua eficácia depende diretamente das prioridades definidas para o sistema.**

6 Conclusão e Trabalhos Futuros

Este trabalho investigou o desempenho de sistemas de recomendação baseados em LLMs em comparação com métodos tradicionais, bem como os efeitos da otimização de *prompts*. Considerando múltiplas dimensões da qualidade das recomendações, os resultados indicam que estratégias baseadas em LLM superam a maioria das abordagens tradicionais em precisão e desempenho agregado, mantendo equilíbrio competitivo em diversidade, cobertura e controle de discrepâncias entre grupos. Embora alguns métodos tradicionais obtenham melhores resultados em métricas específicas, esses ganhos tendem a ocorrer a custo de perdas em outras dimensões. Quanto a otimização de *prompts*, os resultados indicam que ela não produz benefícios universais, mas atua como mecanismo de calibração, ajustando o equilíbrio entre precisão e diversidade conforme os objetivos da aplicação.

Como direções futuras, propomos a validação em ambientes online por meio de testes A/B para avaliar o impacto no engajamento e na satisfação. Também pretendemos replicar os experimentos em conjuntos de dados adicionais e de diferentes domínios, como Amazon Reviews e Last.FM, a fim de verificar a generalização dos resultados em cenários com características distintas de interação, conteúdo e esparsidade. Além disso, planejamos avaliar outros LLMs abertos, como Mistral e Qwen, bem como comparar a abordagem proposta com baselines recentes baseadas em LLMs, permitindo analisar em que medida os resultados observados dependem do modelo utilizado e se mantêm frente a métodos estado-da-arte. Por fim, propomos investigar novas estratégias de construção e otimização de *prompts*, bem como métricas mais refinadas de justiça e impacto social, ampliando a compreensão sobre o papel de LLMs em sistemas de recomendação multiobjetivo.

Declarações complementares

Agradecimentos

Esta pesquisa foi financiada por: CNPq, Capes, Fapemig, Fapesp, AWS, NVIDIA, CIIA-Saúde e Instituto Nacional de Ciência e Tecnologia em Inteligência Artificial Responsável para Linguística Computacional, Tratamento e Disseminação de Informação (INCT-TILD-IAR; 408490/2024-1).

Contribuições dos autores

H. Sekido, G. Prenassi, L. Rocha e M. Manzato contribuíram para a concepção do estudo (*Conceptualization*) e análise formal dos resultados (*Formal analysis*). H. Sekido foi responsável pelo desenvolvimento dos códigos (*Software*), investigação (*Investigation*) e escrita do rascunho original (*Writing – original draft*). M. Manzato atuou na supervisão da pesquisa (*Supervision*). G. Prenassi e L. Rocha colaboraram na validação da metodologia (*Validation*). Todos os autores participaram da revisão (*Writing – review & editing*) e aprovaram o manuscrito final.

Conflitos de interesse

Os autores declaram que não têm nenhum conflito de interesses.

Disponibilidade de dados e materiais

Os conjuntos de dados (e/ou softwares) gerados e/ou analisados durante o estudo atual serão feitos mediante solicitação.

Outras informações relevantes

Ferramentas de Inteligência Artificial Generativa foram utilizadas como suporte à revisão gramatical do texto. Os autores realizaram uma revisão completa do texto e assumem total responsabilidade pela integridade das informações apresentadas.

Referências

Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B., and Malthouse, E. C. (2021). User-centered evaluation of popularity bias in recommender systems. In *Proceedings of the 29th ACM Conference UMAP 2021, Utrecht*, pages 119–129. ACM. DOI: 10.1145/3450613.3456821.

Atauchi, P. D. F., Zanon, A. L., Rocha, L. C. D. d., and Manzato, M. G. (2025). Do calibrated recommendations affect explanations? a study on post-hoc adjustments. 16:441–460. DOI: 10.5753/jis.2025.5563.

Carvalho, R. and Rocha, L. (2020). Estratégias para aprimorar a diversidade categórica e geográfica de sistemas de recomendação de pois. In *Simpósio Brasileiro de Sistemas Multimídia e Web (WebMedia)*, pages 23–26. SBC.

da Silva, D. C., Manzato, M. G., and Durão, F. A. (2021). Exploiting personalized calibration and metrics for fairness recommendation. *Expert Systems with Applications*, 181:115112. DOI: https://doi.org/10.1016/j.eswa.2021.115112.

de Souza, R. F. and Manzato, M. G. (2024). Uma abordagem em etapa de processamento para redução do viés de popularidade. In *Brazilian Symposium on Multimedia and the Web (WebMedia)*, pages 310–317. SBC.

Fonseca, G., Cunha, W., Prenassi, G., Gonçalves, M. A., and Rocha, L. C. D. D. (2025). Instance-selection-inspired un-

dersampling strategies for bias reduction in small and large language models for binary text classification. In *Proceedings of the 63rd ACL*, pages 9323–9340. Association for Computational Linguistics.

Gao, J., Chen, B., Zhao, X., Liu, W., Li, X., Wang, Y., Wang, W., Guo, H., and Tang, R. (2025). Llm4rerank: Llm-based auto-reranking framework for recommendations. In *Proceedings of the ACM on Web Conference 2025*, WWW '25, page 228–239. DOI: 10.1145/3696410.3714922.

Harper, F. M. and Konstan, J. A. (2015). The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.*, 5(4). DOI: 10.1145/2827872.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. (2021). Lora: Low-rank adaptation of large language models.

Klimashevskaia, A., Jannach, D., Elahi, M., and Trattner, C. (2024). A survey on popularity bias in recommender systems. *User Modeling and User-Adapted Interaction*, 34(5):1777–1834.

Lichtenberg, J. M., Buchholz, A., and Schwöbel, P. (2024). Large language models as recommender systems: A study of popularity bias. In *Proceedings of the SIGIR 2024 Workshop on Generative Information Retrieval*.

Liu, D., Yang, B., Du, H., Greene, D., Lawlor, A., Dong, R., and Li, I. (2023). Recprompt: A prompt tuning framework for news recommendation using large language models. *CoRR*.

Ortega, G. M., de Souza, R. F., and Manzato, M. G. (2024). Evaluating zero-shot large language models recommenders on popularity bias and unfairness: A comparative approach to traditional algorithms. In *Simpósio Brasileiro de Sistemas Multimídia e Web (WebMedia)*, pages 45–48. SBC.

Prenassi, G., Souza, R. F. d., Sekido, H. Y., Fonseca, G., Manzato, M. G., and Rocha, L. C. D. d. (2025). Calibração de sistemas de recomendação com llms: otimização de prompts para balancear precisão, diversidade e justiça. *Anais*.

Quadrana, M., Cremonesi, P., and Jannach, D. (2018). Sequence-aware recommender systems. *ACM computing surveys (CSUR)*, 51(4):1–36.

Rendle, S., Freudenthaler, C., Gantner, Z., and Schmidt-Thieme, L. (2012). Bpr: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618*.

Rey, D. and Neuhäuser, M. (2011). Wilcoxon-signed-rank test. In *International encyclopedia of statistical science*, pages 1658–1659. Springer.

Sacilotti, A., Souza, R. F. d., and Manzato, M. G. (2023). Counteracting popularity-bias and improving diversity through calibrated recommendations. In *In Proceedings of the 25th International Conference on Enterprise Information Systems*, volume 1, Prague, Czech Republic. Scitepress.

Souza, R. and Manzato, M. (2024). A two-stage calibration approach for mitigating bias and fairness in recommender systems. In *Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing*, pages 1659–1661, New York, NY, USA. ACM.

Steck, H. (2018). Calibrated recommendations. In *Proceedings of the 12th ACM conference on recommender systems*, pages 154–162.

Wang, J., Chen, X., Lee, K.-C., Ghosh, D., Rao, N., and Hu, H. (2025). Automating personalization: Prompt optimization for recommendation reranking.

Werneck, H., Silva, N., Viana, M. C., Mourão, F., Pereira, A. C., and Rocha, L. (2020). A survey on point-of-interest recommendation in location-based social networks. In *Proceedings of the Brazilian Symposium on Multimedia and the Web*, pages 185–192.

Wu, L., Zheng, Z., Qiu, Z., Wang, H., Gu, H., Shen, T., Qin, C., Zhu, C., Zhu, H., Liu, Q., *et al*. (2024). A survey on large language models for recommendation. *World Wide Web*, 27(5):60.

Yang, C., Wang, X., Lu, Y., Liu, H., Le, Q. V., Zhou, D., and Chen, X. (2023). Large language models as optimizers. In *The Twelfth International Conference on Learning Representations*.

Zanon, A. L., da Rocha, L. C. D., and Manzato, M. G. (2022). Balancing the trade-off between accuracy and diversity in recommender systems with personalized explanations based on linked open data. *Knowl. Based Syst.*, 252:109333. DOI: 10.1016/J.KNOSYS.2022.109333.

Zhao, Y., Wang, Y., Liu, Y., Cheng, X., Aggarwal, C. C., and Derr, T. (2025). Fairness and diversity in recommender systems: A survey. *ACM Trans. Intell. Syst. Technol.*, 16(1). DOI: 10.1145/3664928.

Zhao, Z., Wallace, E., Feng, S., Klein, D., and Singh, S. (2021). Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pages 12697–12706. PMLR.