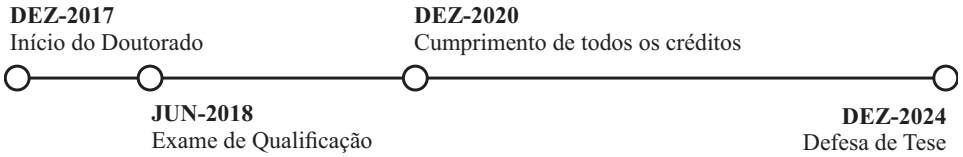
Repense, Recupere e Reranqueie: Um Pipeline de Recuperação e Reranqueamento para Classificação de Texto Multi-rótulo Extrema

Celso França[1], **Berthier Ribeiro-Neto**[1], **Marcos André Gonçalves**[1]

[1]Departamento de Ciência da Computação – Universidade Federal de Minas Gerais (UFMG) Belo Horizonte – MG – Brasil.

{celsofranca,berthier,mgoncalv}@dcc.ufmg.br

Resumo. *A classificação de texto multi-rótulo extrema (XMTC) envolve a atribuição de rótulos relevantes a um texto a partir de um enorme espaço de rótulos. Abordando os desafios centrais da XMTC (**volume**, **desbalanceamento** e **qualidade**), propomos o xCoRetriev, um pipeline de dois estágios migrando de uma perspectiva de classificação para uma abordagem de recuperação de informações (IR). Tratamos o desafio de **volume** combinando de forma eficiente métodos de IR; enfrentamos o desafio do **desbalanceamento** capturando melhor a relação texto-rótulo e; aprimoramos a **qualidade** enriquecendo os nomes dos rótulos com pseudo-rótulos. Nossos resultados demonstram os pontos fortes do xCoRetriev quando comparado a linhas de base em termos de: (i) escalabilidade para grandes espaços de rótulos e quantidade de textos; (ii) eficácia diante do alto desbalanceamento, especialmente para predição de rótulos infrequentes – com ganhos de até 40% em MRR e NDCG –; e (iii) capacidade de lidar com textos e rótulos anotados de baixa qualidade.*



Artigos Publicados Recentemente:

2023 - Celso França, et al. "A Comparative Survey of Instance Selection Methods applied to NonNeural and Transformer-Based Text Classification."ACM Computing Surveys (2023).

2023 - Celso França, et al. "On the class separability of contextual embeddings representations."Information Processing & Management 60.4 (2023).

2023 - Celso França, et al. "An Effective, Efficient, and Scalable Confidence-Based Instance Selection Framework for Transformer-Based Text Classification."Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. 2023.

2022 - Celso França, et al. "Reforço e Delimitaçao Contextual para Reconhecimento de Entidades e Relaçoes em Documentos Oficiais."Anais do XXXVII Simpósio Brasileiro de Bancos de Dados. SBC, 2022.

1. Introdução

Vários algoritmos têm sido propostos para abordar a classificação de texto multi-rótulo, um paradigma de aprendizado de máquina destinado a atribuir rótulos relevantes a um texto. Embora relativamente bem-sucedidas, essas abordagens muitas vezes assumem que o espaço de rótulos é pequeno, o que pode ser muito restritivo em muitas aplicações do mundo real. Por exemplo, milhares de rótulos estão disponíveis para descrever os artigos da Wikipédia e milhões de rótulos foram usadas para descrever produtos de comércio eletrônico. Esses cenários, envolvendo muitos rótulos, representam um caso extremo de classificação de texto conhecido como *Classificação de Texto Multi-rótulo Extrema (**XMTC**)*. XMTC tem recebido atenção crescente nos últimos anos, pois apresenta desafios técnicos e computacionais únicos.

Os principais desafios do XMTC estão associados a três questões principais [Wei et al. 2022]: (i) **volume**, referente ao crescimento explosivo na quantidade de características de entrada e do espaço de rótulos de saída, causando problemas de escalabilidade; (ii) **desbalanceamento** da informação, que se refere à distribuição altamente desequilibrada de rótulos, tipicamente seguindo distribuições de cauda longa; e (iii) **qualidade** relacionada à disparidade de significados entre textos contendo muitas palavras e as rótulos atribuídas a eles, geralmente consistindo apenas de alguns termos. Essa discrepância entre a complexidade dos textos e a simplicidade dos rótulos gera dificuldades ao tentar corresponder os textos aos seus rótulos correspondentes com precisão.

Os modelos estado-da-arte para XMTC, como XR-Transformer [Yu et al. 2022], AttentionXML [You et al. 2019] e LightXML [Jiang et al. 2021], têm abordado principalmente o desafio de **volume** por meio de um esforço significativo para ajustar eficientemente arquiteturas pré-treinadas baseadas em transformers. Essas abordagens enfrentam consideráveis quedas de eficácia quando avaliadas sob as perspectivas do **desbalanceamento** e da **qualidade**, especialmente em relação aos rótulos menos frequentes, doravante referidos como *rótulos tail*. De fato, os rótulos *tail* correspondem a cerca de 80% do espaço de rótulos em vários cenários do mundo real [Hou et al. 2022, Ge et al. 2022, Huang and Wu 2019].

Argumentamos que, para muitas aplicações do mundo real, os rótulos *tail* são mais informativos e *recompensadores* do que os rótulos *head* (os rótulos mais frequentes)[Wei et al. 2022, Jain et al. 2016]. Por exemplo, no comércio eletrônico, como Amazon ou Alibaba, com milhões de produtos, geralmente é recompensador recomendar uma lista personalizada de produtos com base nos dados do cliente (como histórico de compras). A maioria dos potenciais clientes ficará frustrada com recomendações tendenciosas apenas para os produtos populares[Wei et al. 2022]. A mesma restrição ocorre no domínio da recuperação de informações ao reformular consultas, onde o mecanismo de busca tem como objetivo recomendar consultas mais alinhadas com as necessidades de informação do usuário [Wei et al. 2022, Zeng et al. 2023].

Dado tudo o que foi mencionado acima, propomos **xCoRetriev**, um *pipeline* de *retrieval* em duas etapas complementares que aborda concomitantemente os três principais desafios do XMTC. Fazemos isso migrando de uma perspectiva de classificação pura para uma abordagem de recuperação de informação. Especificamente, em nossa solução, empregamos um pipeline de Recuperação e Reranqueamento em duas etapas: (i) primeiro, recuperamos um conjunto de k rótulos candidatos potencialmente relevantes

e, em seguida, (ii) aplicamos um reranqueador para atribuir uma pontuação de relevância a todos os candidatos em relação à consulta na etapa de reranqueamento.

Nossa hipótese é que podemos lidar melhor com os desafios do XMTC ao reduzi-lo a uma tarefa de recuperação de informação (IR), especialmente em cenários com um grande espaço de rótulos. Em última análise, para sustentar essa tese, nosso objetivo é responder: **(RQ1 - Desbalanceamento)** Quão eficaz é nosso pipeline de Recuperação e Reranqueamento em duas etapas em comparação com abordagens estado-da-arte em rótulos *head* e principalmente na predição de rótulos *tail*?; **(RQ2 - Volume)** O pipeline proposto escala para espaços de rótulos enormes e grande número de características de entrada?; e **(RQ3 - Qualidade)** Os pseudo-rótulos ajudam a enfrentar o desafio da qualidade dos textos anotados?

2. Revisão da Literatura

Abordagens estado-da-arte para XMTC, como **AttentionXML**[You et al. 2019], **XR-Transformer**[Yu et al. 2022] e **LightXML**[Jiang et al. 2021], empregam métodos baseados em árvores buscando diminuir o tempo de inferência e escalonar de forma logarítmica com o número de rótulos. **AttentionXML** primeiro agrupa os rótulos em uma árvore de rótulos probabilística (PLT) e, para cada nível da árvore, treina uma memória de curto prazo de longo alcance bidirecional (BiLSTM) atenta à atenção. **Light XML** segue um caminho semelhante ao **AttentionXML**, agrupando os rótulos na PLT, mas substituindo a BiLSTM por um transformer. **XR-Linear** e **XR-Transformer** seguem um framework recursivo de três estágios: (i) particionamento de rótulos em vários clusters; (ii) correspondência de um texto aos clusters relevantes; e (iii) classificação dos rótulos correspondentes. XR-Linear usa um classificador linear e XR-Transformer emprega um transformer na etapa de correspondência.

Esses estudos autoproclamados estado-da-arte têm se concentrado no desafio do **volume** considerando todos os rótulos igualmente importantes. Além disso, apresentaram uma avaliação tendenciosa em relação aos rótulos *head*. Poucos trabalhos recentes abordam marginalmente o desafio da **desbalanceamento**, como o AttentionXML, ao representar um texto dado de forma diferente para cada rótulo, o que é especialmente útil para muitos rótulos *tail*. **XRR** [Xiong et al. 2023] emprega um framework com baseado em informação mútua alinhada (aPMI) para capturar a coocorrência de termos de texto e rótulos, o que pode ser um esforço em relação ao desafio da qualidade. No entanto, o XRR possui alto custo computacional para calcular o aPMI na etapa de inferência.

Por outro lado, **xCoRetriev** emprega métodos tradicionais de recuperação de informação baseados em índices invertidos, proporcionando uma solução eficiente e escalável para o desafio do **volume**. Respondemos aos desafios de **qualidade** e **desbalanceamento** aumentando os nomes dos rótulos com *pseudo-rótulos*: um conjunto de palavras-chave extraídas do texto que podem representar melhor os rótulos *tail*.

3. Abordagem Proposta

Assumindo um conjunto $\{t_i, y_i\}_{i=1}^{N}$ onde $t_i \in \mathcal{T}$ é o i-ésimo texto de entrada e $y_i \in \{0,1\}^L$ é um vetor binário de dimensão L com $y_{i,\ell} = 1$ indicando que o rótulo ℓ é relevante para o t_i. O objetivo do XMTC é aprender uma função $f : \mathcal{T} \times |L| \rightarrow \mathbb{R}$, tal que $f(t, \ell)$ denota a relevância do rótulo ℓ em relação ao texto t. Formalmente, reduzimos a

tarefa XMTC para IR mapeando o texto t em uma consulta, visualizando os rótulos como documentos e definindo $f(t, \ell)$ como uma função de ranqueamento. Portanto, **reconsiderando** o XMTC como uma tarefa de recuperação de informação, propomos **xCoRetriev**, um pipeline complementar de recuperação e reranqueamento em duas etapas que aborda os três desafios do XMTC. No geral, o primeiro estágio divide o grande espaço de rótulos recuperando um conjunto de k rótulos candidatos potencialmente relevantes, e o segundo atribui uma pontuação de relevância aplicando um *transformer* a todos os pares candidatos (texto, rótulo), como mostrado na Figura 1.

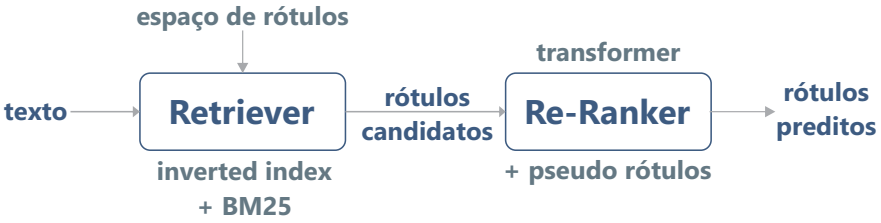


Figura 1. Pipeline de recuperação e reranqueamento.

Adotando um pipeline de Recuperação e Reranqueamento em duas etapas, **xCoRetriev**, combinamos as vantagens das abordagens de recuperação e classificação, o que proporciona uma solução mais escalável, uma vez que o estágio de recuperação ajuda a reduzir o espaço de rótulos para um conjunto menor de candidatos e melhora a eficiência do processamento subsequente. Além disso, o estágio de reranqueamento permite uma avaliação mais precisa da relevância do rótulo, considerando as características específicas do texto da consulta e dos rótulos candidatos.

3.1. Estágio de Recuperação

A primeira etapa no pipeline é um **retriever**, na qual buscamos filtrar o espaço de busca de rótulos selecionando o subconjunto mais promissor de rótulos em relação a um texto dado. Essencialmente, o *retriever* é um índice de busca que retorna rótulos candidatos como resposta a um texto.

Especificamente, primeiro construímos um índice invertido usando textos com rótulos conhecidos. Em seguida, utilizando um texto t como consulta, recuperamos textos com características semelhantes a t no índice e retornamos os rótulos atribuídos. Em seguida, aproveitamos a função de ranqueamento *BM25* para classificar a relevância de todos os rótulos candidatos em relação a t e a cada texto recuperado. Finalmente, selecionamos um subconjunto de k rótulos como resposta. Em resumo, o *retriever* baseia-se em encontrar textos com características contextuais semelhantes para recuperar seus rótulos como candidatos promissores para o estágio de reranqueamento.

3.2. Estágio de Reranqueamento

O *retriever* é eficiente para espaços de rótulos grandes com milhares de entradas. No entanto, ele pode retornar rótulos candidatos irrelevantes. Portanto, um reranqueador pode melhorar substancialmente as classificações finais de rótulos, prevendo a relevância dos rótulos de forma mais precisa para o texto.

Propomos aproveitar modelos *transformers*, como *BERT*, como uma função parametrizada de ranqueamento f_θ, onde realizamos um *forward-pass* do par (texto, rótulo)

em um abordagem *cross-encoders*. A vantagem dos *cross-encoders* é a eficácia, pois o mecanismo de atenção do *transformer* realiza interações finas entre todos os termos do texto e do rótulo. Portanto, nosso pipeline responde ao desafio da **desbalanceamento** selecionando candidatos promissores de rótulos no estágio de recuperação e calculando a pontuação de relevância detalhada no estágio de re-ranqueamento.

3.3. *Pseudo* Rótulos

Textos mais longos são comuns em tarefas de XMTC, enquanto uma ou algumas palavras compõem os rótulos. Isso contribui para que os rótulos sejam ruidosos e ambíguos, e seu significado seja efetivamente entendido apenas pela "compreensão"do texto [Garg et al. 2021, Wang et al. 2023]. Portanto, visando incorporar propriedades semânticas, nossa solução explora uma abordagem de extração de palavras-chave baseada em um *transformer zero-shot*. Para cada instância *(texto, rótulos)*, extraímos um conjunto de *n-gramas* (palavras-chave candidatas), codificando-os em uma representação vetorial densa. Finalmente, selecionamos um subconjunto dos k principais palavras-chave mais relevantes (pseudo-rótulos) para enriquecer um rótulo com base no escore de similaridade de cosseno.

4. Metodologia Experimental

Nós treinamos, validamos e testamos **xCoRetriev** com os dois conjuntos de dados de XMCT mais representativos: Wiki10-31K e Amazon-670K. As estatísticas dos dados são fornecidas na Tabela 1. Contrastamos nossa proposta com as abordagens atuais estado-da-arte de XMTC que foram apresentadas na Seção 2: **XLinear**, **XR-Transformer**, **AttentionXML**, **LightXML** e **XRR**.

Tabela 1. Estatísticas do conjunto de dados (*dataset*) demonstrando o número de instâncias de texto (N); a quantidade de rótulos ($\bar{L}$); o número médio de rótulos *tail* ($\bar{t}$) e rótulos *head* ($\bar{h}$); e o número médio de instâncias por rótulo ($\bar{n}$).

Dataset	N	L	$\bar{t}$	$\bar{h}$	$\bar{n}$
Wiki10-31k	20,762	30,938	3.66	15.10	8.52
Amazon-670K	643,474	670,091	2.56	2.83	5.45

A eficácia é medida como a média entre os cinco *folds* utilizando métricas tradicionais de recuperação de informação: *mean reciprocal rank (MRR)* e *normalized discounted cumulative gain (nDCG)*. Além disso, estratificamos nossa análise em relação aos rótulos *tail* e rótulos *head*. Formalmente, utilizamos o princípio de Pareto para categorizar 80% dos rótulos menos frequentes como *tail* e os 20% restantes como *head*. Finalmente, avaliamos a significância estatística de nossos resultados empregando o *Two-sided Paired Student's t-Test* com 95% de confiança para comparar os resultados médios de nossos experimentos de validação cruzada.

5. Resultados Experimentais

Esta seção apresenta os resultados experimentais com o objetivo de responder às três questões de pesquisa formuladas.

5.1. RQ1 - Desbalanceamento

As Tabelas 2 e 3 apresentam uma avaliação abrangente da eficácia do **xCoRetriev** em comparação com as baselines experimentadas. Valores em negrito indicam melhorias significativas de acordo com o *Two-sided Paired Student's t-Test* com 95% de confiança.

Tabela 2. MRR@k e nDCG@k de todos os modelos testados aplicados no conjunto de dados Amazon-670k para rótulos *tail* e rótulos *head*.

	MRR x 100						nDCG x 100					
	Rótulos *Tail*			Rótulos *Head*			Rótulos *Tail*			Rótulos *Head*		
Modelo	**@1**	**@5**	**@10**	**@1**	**@5**	**@10**	**@1**	**@5**	**@10**	**@1**	**@5**	**@10**
XR-linear	-	-	-	-	-	-	-	-	-	-	-	-
XR-Transformer	-	-	-	-	-	-	-	-	-	-	-	-
AttentionXML	26.7(0.4)	33.3(0.4)	34.1(0.3)	**44.8(0.2)**	**52.5(0.2)**	**53.3(0.2)**	26.7(0.4)	21.8(0.2)	21.8(0.2)	**44.8(0.2)**	35.7(0.2)	35.5(0.2)
LightXML	10.8(1.9)	18.2(2.5)	19.8(2.5)	43.1(0.1)	50.6(0.6)	51.6(0.6)	10.8(1.9)	15.9(2.5)	21.0(1.9)	43.1(0.1)	**42.1(0.0)**	**46.4(0.6)**
XRR	-	-	-	-	-	-	-	-	-	-	-	-
xCoRetriev	**31.8(2.7)**	**38.9(3.2)**	**39.6(3.3)**	37.9(3.2)	46.4(4.0)	47.3(3.9)	**31.8(2.7)**	**24.7(2.3)**	**25.0(2.6)**	37.9(3.2)	30.3(2.9)	31.2(3.0)

Tabela 3. MRR@k e nDCG@k de todos os modelos testados aplicados no conjunto de dados Wiki10-31k para rótulos *tail* e rótulos *head*.

	MRR x 100						nDCG x 100					
	Rótulos *Tail*			Rótulos *Head*			Rótulos *Tail*			Rótulos *Head*		
Modelo	**@1**	**@5**	**@10**	**@1**	**@5**	**@10**	**@1**	**@5**	**@10**	**@1**	**@5**	**@10**
XR-linear	11.4(0.7)	13.2(0.7)	13.2(0.7)	84.7(0.7)	91.1(0.5)	91.1(0.4)	11.4(0.7)	5.5(0.3)	3.6(0.2)	84.7(0.7)	70.8(0.2)	58.6(0.3)
XR-Transformer	10.0(0.4)	12.4(0.5)	12.5(0.5)	87.4(0.7)	92.7(0.5)	92.8(0.5)	10.0(0.4)	5.4(0.3)	3.7(0.2)	87.4(0.7)	73.1(0.4)	60.3(0.2)
AttentionXML	16.5(1.5)	22.2(1.8)	22.8(1.9)	87.5(1.3)	92.8(0.6)	92.9(0.6)	16.5(1.5)	10.3(0.9)	7.5(0.7)	87.5(1.3)	73.3(0.5)	61.1(0.4)
LightXML	0.0(0.0)	0.0(0.0)	0.1(0.1)	**89.4(0.6)**	**93.8(0.4)**	**93.9(0.3)**	0.0(0.0)	0.0(0.0)	0.1(0.1)	**89.4(0.6)**	**74.5(0.2)**	**63.3(0.2)**
XRR	0.2(0.2)	0.5(0.3)	0.7(0.3)	59.1(1.2)	62.4(1.2)	62.9(1.2)	0.2(0.2)	0.2(0.1)	0.2(0.1)	59.1(1.2)	31.3(1.6)	21.8(1.3)
xCoRetriev	**23.1(2.0)**	**30.1(1.6)**	**30.9(1.6)**	57.2(5.2)	71.8(4.5)	72.5(4.2)	**23.1(2.0)**	**14.2(0.7)**	**10.5(0.5)**	57.2(5.2)	47.4(6.0)	41.3(6.0)

Os rótulos *tail* constituem a maioria do **volume** do vasto espaço de soluções. Essa característica resulta em uma distribuição de rótulos altamente desequilibrada, frequentemente seguindo um padrão *long tail* que contribui para a **desbalanceamento** do conjunto de dados. Além disso, os rótulos *tail* são propensos a exibir maior ambiguidade e ruído, o que impacta sua **qualidade** geral. Consequentemente, abordar os desafios dos rótulos *tail* é crucial na Classificação Extrema de Texto com Múltiplas Etiquetas (XMTC). Vale ressaltar que atribuir um rótulo *tail* relevante a um texto possui importância significativamente maior, com recompensas que podem ser várias vezes maiores do que definir um rótulo *head*.

Neste contexto, **xCoRetriev** define um novo estado-da-arte em ambos os conjuntos de dados Wiki10-31k e Amazon-670k, superando estatisticamente todas as baselines. De fato, obtivemos ganhos de até 40% em MRR@1 e NDCG@1 no Wiki10-31k em relação à melhor baseline nos rótulos *tail* - AttentionXML - e entre 13-37% em @5 e @10, para ambas as métricas, em relação a essa mesma (melhor) baseline em ambos os conjuntos de dados. Em nossos experimentos, não pudemos executar XR-Linear, XR-Transformer e XRR (com as configurações relatadas pelos autores) no Amazon-670k devido ao alto requisito de RAM (mais de 128GB).

5.2. RQ2 - Volume

A Tabela 4 compara a complexidade temporal e o tempo de execução para as etapas de treinamento e previsão para descrever como cada modelo aborda o desafio de volume. Uma vez que os conjuntos de dados avaliados diferem no número de textos de treinamento N_T, textos de previsão N_Q e número de rótulos L, modelamos a complexidade

temporal como uma função dessas variáveis, mantendo apenas os termos mais custosos e abstraindo os custos computacionais específicos do modelo.

Tabela 4. Complexidade assintótica de tempo e tempo de execução para as etapas de treinamento e predição.

Modelo	Complexidade de Tempo		Treinameto/Predição (hrs)	
	Treinameto	**Predição**	**Wiki10-31k**	**Amazon-670k**
XR-Linear	$\mathcal{O}(DC_l\ N_T)$	$\mathcal{O}(DC_l\ N_Q)$	0.40/0.21	-
XR-Transformer	$\mathcal{O}(C_f\ N_T + DC_l\ N_T)$	$\mathcal{O}(C_f\ N_Q + DC_l\ N_Q)$	0.64/0.40	-
AttentionXML	$\mathcal{O}(L \log L + D\mathcal{C}_b\ N_T)$	$\mathcal{O}(D\mathcal{C}_b\ N_Q\ k\ )$	2.65/0.66	26.65/7.96
LightXML	$\mathcal{O}(L \log L + D\mathcal{C}_f\ N_T)$	$\mathcal{O}(D\mathcal{C}_f\ N_Q\ k\ )$	6.92/2.30	43.03/18.44
XRR	$\mathcal{O}(N_T M\mathcal{C}_c\ + N_T\mathcal{C}_f\ )$	$\mathcal{O}(N_Q ML) + \mathcal{O}(N_Q\mathcal{C}_f\ \ k\ )$	6.29/0.45	-
xCoRetriev	$\mathcal{O}(\mathcal{C}_f\ N_T\ k\ )$	$\mathcal{O}(\mathcal{C}_f\ N_Q\ k\ )$	2.16/0.13	32.95/9.84

Apesar de exigirem um ambiente de data center, XR-Linear e XR-Transformer surgiram como soluções altamente eficientes para a tarefa de XMTC. O XRR torna-se inviável em conjuntos de dados maiores, como o Amazon-670k. Por outro lado, AttentionXML e LightXML demonstraram eficácia superior em comparação com seus requisitos de tempo de execução para previsão e treinamento, provando assim sua viabilidade para a tarefa de XMTC.

De forma consistente, **xCoRetriev** está entre os métodos mais rápidos durante o treinamento e predição. No conjunto de dados **Wiki10-31k**, **xCoRetriev** perde apenas para XR-Linear e XR-Transformer em termos de tempo de treinamento e é o mais rápido entre todos os métodos para prever. E entre os métodos que puderam ser executados no conjunto de dados **Amazon-670k**, **xCoRetriev** parece ligeiramente mais lento que o AttentionXML tanto no tempo de treinamento quanto no de predição.

5.3. RQ3 - Qualidade

Conduzimos um estudo de ablação para avaliar o impacto dos pseudo-rótulos na eficácia do **xCoRetriev**. Conforme mostrado na Tabela 5, observamos o impacto negativo em MRR e $NDCG$ quando re-treinamos **xCoRetriev** sem enriquecer os rótulos com pseudo-rótulos (linha - pseudo-rótulos) no conjunto de dados Amazon-670k.

Tabela 5. Estudo de ablação de qualidade no conjunto de dados Amazon-670k.

	MRR x 100						nDCG x 100					
	Rótulos *Tail*			Rótulos *Head*			Rótulos *Tail*			Rótulos *Head*		
Modelo	**@1**	**@5**	**@10**	**@1**	**@5**	**@10**	**@1**	**@5**	**@10**	**@1**	**@5**	**@10**
xCoRetriev	**31.8(2.7)**	**38.9(3.2)**	**39.6(3.3)**	**37.9(3.2)**	**46.4(4.0)**	**47.3(3.9)**	**31.8(2.7)**	**24.7(2.3)**	**25.0(2.6)**	**37.9(3.2)**	**30.3(2.9)**	**31.2(3.0)**
- pseudo-labels	11.7(0.6)	17.9(1.3)	19.6(0.6)	15.1(0.6)	23.4(1.4)	24.4(1.2)	11.7(0.6)	9.9(0.5)	12.0(0.6)	15.1(0.6)	13.6(0.6)	16.7(0.6)

Sem os pseudo-rótulos, a perda de eficácia é superior a 50% em todos os *MRRs* e *nDCGs*, alcançando em média mais de 60% de perda em MRR@1 para rótulos *tail* e rótulos *head*.

6. Conclusão e Direções Futuras

Quanto ao desafio de volume (**RQ2**), nosso pipeline proposto requer apenas um ambiente computacional acadêmico padrão, enquanto se equipara às baselines em termos de complexidade temporal. Sob a perspectiva do desafio de desbalanceamento (**RQ1**), **xCoRetriev** estabelece um novo estado-da-arte para todos os cenários avaliados em relação

aos rótulos *tail*. Finalmente, nossas análises revelam que os pseudo-rótulos são essenciais para lidar com o desafio de qualidade (**RQ3**) ao aprimorar o significado dos rótulos com base em características textuais. Atualmente, estamos definindo e escrevendo o projeto de tese enquanto aguardamos a revisão de alguns artigos em revisão. Uma vez concluída esta etapa, dedicaremos nossos esforços à redação da tese considerando o feedback do comitê de avaliação, culminando na defesa final da tese.

Referências

Garg, S. et al. (2021). Towards robustness to label noise in text classification via noise modeling. In *CIKM*, CIKM '21, page 3024–3028, New York, NY, USA. ACM.

Ge, Y. et al. (2022). Explainable fairness in recommendation. In *SIGIR*, SIGIR '22, page 681–691, New York, NY, USA. ACM.

Hou, R. et al. (2022). Contrastive-weighted self-supervised model for long-tailed data classification with vision transformer augmented. *Mechanical Systems and Signal Processing*, 177:109174.

Huang, X. and Wu, F. (2019). A novel topic-based framework for recommending long tail products. *Computers & Industrial Engineering*, 137:106063.

Jain, H. et al. (2016). Extreme multi-label loss functions for recommendation, tagging, ranking and other missing label applications. In *KDD*, page 935–944, New York, NY, USA. ACM.

Jiang, T. et al. (2021). Lightxml: Transformer with dynamic negative sampling for high-performance extreme multi-label text classification. In *AAAI*, volume 35, pages 7987–7994.

Wang, J., Chen, Z., Qin, Y., He, D., and Lin, F. (2023). Multi-aspect co-attentional collaborative filtering for extreme multi-label text classification. *KBS*, 260(2):1–11.

Wei, T., Mao, Z., Shi, J.-X., Li, Y.-F., and Zhang, M.-L. (2022). A survey on extreme multi-label learning. *arXiv*.

Xiong, J., Yu, L., Niu, X., and Leng, Y. (2023). Xrr: Extreme multi-label text classification with candidate retrieving and deep ranking. *Information Sciences*, 622:115–132.

You, R. et al. (2019). Attentionxml: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification. In Wallach, H. et al., editors, *NIPS*, volume 32. Curran Associates, Inc.

Yu, H.-F. et al. (2022). Pecos: Prediction for enormous and correlated output spaces. In *SIGKDD*, KDD '22, page 4848–4849, New York, NY, USA. ACM.

Zeng, J. et al. (2023). Personalized dynamic attention multi-task learning model for document retrieval and query generation. *ESA*, 213:119026.