Aprendizado Federado Sensível ao Risco em Modelos de Ranqueamento

Gestefane Rabbi Magalhães [1]
Marcos André Gonçalves[1]
Daniel Xavier de Sousa[2]
Celso França[1]

Programa de Pós-graduação em Ciência da Computação da UFMG

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

[2]Instituto Federal de Goiás (IFG)
Anápolis – GO – Brasil

{gestefane,celsofranca,marcos}@dcc.ufmg.br, daniel.sousa@ifg.edu.br

Resumo. *Essa dissertação explora o uso do Aprendizado Federado para Ranqueamento (Federated Learning to Rank - FL2R), uma técnica empregada em sistemas de busca que considera a privacidade dos dados de diversos clientes. O FL2R envolve a construção de um modelo de ranqueamento executado de forma distribuída em vários dispositivos. Após o treino, os parâmetros das redes neurais dos clientes são combinados, resultando em um novo modelo neural que será distribuído aos clientes. Considerado o estado da arte em Federated Learning (FL), o método Federated Averaging (FedAvg) calcula a média de parâmetros para construir o modelo agregado. Contudo, clientes com baixo desempenho podem distorcer a média de forma enviesada, resultando em uma redução na efetividade do modelo global. Para contribuir na solução desse problema, propomos o estudo de técnicas de agregação que superem a simples média aritmética dos pesos, além de aplicar métricas na área de Sensibilidade ao Risco, tentando mitigar a variância dos modelos no lado do cliente. Embora o trabalho esteja em fase inicial, neste artigo foi possível mostrar alguns experimentos fazendo uso de Projeto Fatorial para avaliação de fatores que possam impactar a efetividade dos modelos federados. Os resultados mostram que combinar os valores dos parâmetros não é uma tarefa trivial, mas considerando as perguntas de pesquisa propostas acreditamos que esse trabalho tem forte potencial de contribuição na área de Recuperação de Informação.*

1. Introdução

A tarefa de ranquear documentos relevantes é um dos principais desafios dos sistemas de Recuperação de Informação (RI). Para ajudar nessa tarefa, técnicas de aprendizado de máquina vêm sendo empregadas para treinar modelos considerando informações mais específicas e dispersas em dispositivos mais próximos aos clientes. Nessa linha, alguns modelos atuam em dados restritos a segmentos ou organizações específicas, fazendo uso de múltiplas fontes de dados, como ocorrem em aplicações para governo e indústria. Tais modelos são denominados Federated Search (FS) [Mukut et al. 2012], e a aplicação de Machine Learning (ML) a esses modelos com o objetivo de melhorar a qualidade do ranqueamento é denominada Federated Learning to Rank (FL2R) [Wang and Zuccon 2022].

Federated Learning (FL) é uma configuração de ML na qual vários clientes (dispositivos móveis ou organizações inteiras), orquestrados por um sistema central, treinam de modo colaborativo um modelo de aprendizado de máquina. Por questão de privacidade, os dados são armazenados em cada cliente e não são transferidos para o servidor central, nem intercambiados entre os participantes da rede. Por outro lado, em um servidor central são realizadas atualizações destinadas à agregação de resultados [Kairouz et al. 2021].

Entre os trabalhos na área de FL, FedAvg [McMahan et al. 2023] tem sido bastante citado na literatura. FedAvg atua no servidor central agregando os modelos, ou seja, um modelo global é obtido a partir da média dos valores dos parâmetros nos dispositivos dos clientes, parâmetros estes usados como modelos nas redes neurais profundas. Os parâmetros com valores agregados geram um novo modelo de rede neural que é distribuído na sequência para os clientes treinarem.

2. Definição do problema

Apesar dos bons resultados, as técnicas de ML empregadas em FL2R estão suscetíveis a erros relacionados ao aprendizado enviesado pelos clientes. Considerando que alguns clientes podem ter uma grande variação no espaço de rótulos e peso dos parâmetros, a estratégia de agregação pode produzir modelos menos efetivos.

Outro ponto de falha no aprendizado para ranqueamento, citado por [Kairouz et al. 2021], está no viés introduzido nos dados de treino, devido a fatores como privacidade, recursos dos hardwares e de velocidade de comunicação dos diferentes dispositivos usados para treino, decorrentes da heterogeneidade dos dispositivos e dos dados.

Em sistemas típicos de ML, dados e modelos são contidos em estruturas centralizadas. Em FL, a descentralização destes elementos e a heterogeneidade dos dados leva a um problema de divergência entre os valores de parâmetros obtidos como média dos modelos distribuídos. Além disso, em um modelo centralizado os dados normalmente são IID (Independentes e Identicamente Distribuídos) enquanto que em um modelo federado os dados são não-IID, i.e., não independentes e não identicamente distribuídos.

[Wang and Zuccon 2022] fortalecem essa argumentação, mostrando que dados não-IID podem afetar a eficácia dos modelos obtidos em ambientes de FL. Os autores comentam que o baixo desempenho pode ser atribuído, principalmente à divergência dos pesos dos modelos locais, decorrente da distribuição não-IID ao longo das rodadas de treinamento. Ainda que iniciados com os mesmos parâmetros, os pesos dos modelos locais finais podem divergir bastante, devido aos dados não-IID dos clientes. A divergência aumenta à medida que as épocas de treino evoluem, de modo que o modelo resultante

se distancie cada vez mais do modelo "ideal´´obtido, por exemplo, em um sistema centralizado. Isso é ilustrado na figura 1, onde vemos do lado esquerdo o comportamento de um modelo com dados IID. Na figura, os pesos do modelo resultante θ_{t+1}^{avg} tendem a convergir para os pesos do modelo de aprendizado centralizado θ_{t+1}. Do lado direito aparece a divergência nos pesos θ_{t+1}^{avg} por causa da natureza não-IID dos dados nos clientes θ_{t+1}^{1} e θ_{t+1}^{2}. É notável a influência de θ_{t+1}^{2} sobre a média θ_{t+1}^{avg}, "empurrando-a" mais para baixo, fazendo-a distanciar de θ_{t+1}, que é o modelo "ideal" quando os dados são IID.

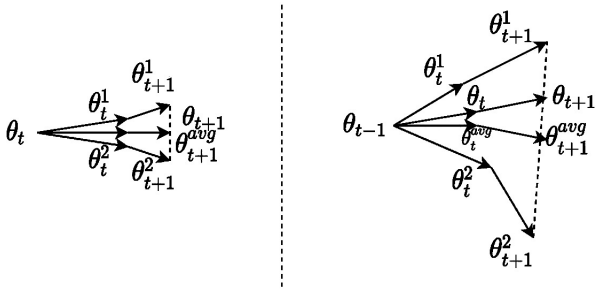

Figure 1. Divergência da média dos pesos ajustados após agregação no servidor central, com base nos pesos coletados nos participantes da rede. Na fase $t+1$ a média calculada θ_{t+1}^{avg} diverge da média do baseline θ_{t+1}

3. Hipótese

Nossa hipótese no contexto de *Federated Learning to Rank*, é que tanto o processo de agregação quanto o modelo de aprendizado local podem ser aperfeiçoados. Ou seja, esperamos minimizar os impactos causados pela variação dos modelos nos clientes. Ao mitigar esses impactos, espera-se melhorar a qualidade do modelo global para ranqueamento obtido a partir do treino coletivo. No intuito de sermos mais claros, seguem nossas principais perguntas de pesquisa:

RQ1: A agregação no servidor pode ser feita de forma mais precisa do que pela média, reduzindo o impacto de clientes com baixa efetividade?

Para responder a esta questão pretendemos investigar fatores que possam servir como ponderadores da média no ato da agregação. Três fatores serão avaliados: o "z-score" que realiza uma padronização nos parâmetros ponderada pelo desvio padrão; a distância ou norma entre os parâmetros de cada cliente e média dos parâmetros; e as métricas que avaliam a sensibilidade ao risco de cada cliente. Dentre esses três fatores será escolhido aquele que resultar em melhor desempenho do modelo global obtido.

RQ2: As redes nos clientes podem ser treinadas, autoavaliadas e produzir um fator de risco que pondere a sua qualidade de ranqueamento?

Para a RQ2 é proposto o cálculo da sensibilidade ao risco de cada cliente. Quanto mais sensível à possibilidade de risco for o modelo, e consequentemente reduzir a chances de erro, melhor é a qualidade do cliente.

RQ3: Um fator de variação na predição ou risco pode ser usado na agregação para ponderar os parâmetros dos clientes trazendo melhorias ao modelo global resultante?

Para a RQ3 é proposta a ponderação dos parâmetros dos clientes pelo fator que mede sua qualidade local, no caso a métrica de sensibilidade ao risco.

4. Revisão da Literatura

O método considerado estado-da-arte em FL é o FedAvg [McMahan et al. 2023], que consiste na distribuição igualitária de parâmetros para todos os modelos locais [Jiang et al. 2020]. Para realizar as agregações, o FedAvg usa técnicas de aprendizado em redes profundas baseadas na média interativa entre os modelos nos clientes [Ye et al. 2020].

No intuito de melhorar a efetividade em modelos de FL, os autores em [Ai et al. 2018] propõem o treinamento com uma função de perda baseada em atenção, mostrando ser mais efetiva e eficiente que os métodos já existentes para listas de ranqueamentos. Em [Divi et al. 2021], os autores avaliam doze trabalhos relacionados às métricas de efetividade aplicadas à FL e argumentam que as melhorias observadas reportam apenas a precisão média em diferentes estratégias de divisões das bases de dados. [Wang and Zuccon 2022] aborda a questão dos dados não-IID presentes nos modelos de aprendizado federado, evidenciando os efeitos negativos na performance dos modelos globais, quando clientes com baixa efetividade participam do processo.

As abordagens típicas na literatura usam médias para agregação e avaliação de desempenho, o que pode ser tendencioso devido a clientes de baixo desempenho. Para resolver isso, propomos ponderar os parâmetros dos cliente com base em sua qualidade, dando menos peso a clientes menos assertivos e mais peso a clientes com melhor desempenho local. A sensibilidade ao risco é o fator de ponderação proposto neste trabalho.

Portanto, na tentativa de responder algumas das perguntas de pesquisa levantadas, esperamos que os conceitos e métricas existentes na área de Sensibilidade ao Risco em Recuperação de Informação possam fornecer os fundamentos necessários. Neste contexto, os autores em [Silva Rodrigues et al. 2022] propõem uma família de funções contínuas que permitem construir modelos de redes neurais profundas considerando como objetivo métricas de sensibilidade ao risco. Experimentalmente o modelo proposto pelos autores para aprendizado de ranqueamento, denominado RiskLoss, avança o estado-da-arte em diversas bases de dados conhecidas da literatura, considerando efetividade e sensibilidade ao risco. Os autores se baseiam na análise de variação dos resultados durante o treino – algo bastante associado ao nosso contexto – e na correlação entre as predições dos resultados obtidos para uma consulta e a relevância correta dos documentos.

5. Abordagem Proposta

Para responder nossa primeira pergunta de pesquisa RQ1, analisamos inicialmente a possibilidade de melhorar a agregação baseada na média de parâmetros, introduzindo um passo antes da agregação. Esse passo consiste em aplicar uma padronização dos parâmetros das redes dos clientes de treino antes de calcular a média. Para a padronização foi adotado o fator "z-score", que indica o quão distante cada parâmetro está da média geral dos mesmos, normalizada pelo desvio-padrão da média. Para isso propomos uma variação do algoritmo FedAvg, à qual denominamos FedZscore.

Na tentativa de responder à RQ2, propomos utilizar nos modelos clientes uma otimização menos sucetível à variação na predição. Ou seja, aumentando a sensibilidade ao risco do modelo nos clientes. Para isso, estudaremos os algoritmos apresentados em [Silva Rodrigues et al. 2022], explorando uma otimização do modelo no cliente considerando um objetivo de sensibilidade ao risco no contexto de redes neurais profundas.

Ainda, considerando RQ1 e RQ3, nos modelos convencionais de FL apenas os parâmetros das redes locais (pesos) são trafegados entre clientes e servidores. Propomos um modelo que envie para o servidor, além dos parâmetros calculados localmente no cliente, o resultado das métricas propostas em [Silva Rodrigues et al. 2022]. A ideia é que a métrica atue como um ponderador dos pesos de cada cliente no momento da agregação, reduzindo a tendência que um cliente enviese muito os valores dos parâmetros. Acreditamos que modelos nos clientes que sofram de maior variância, sejam menos confiáveis para compartilhar seus pesos, assim como apresentado nas análises em [Wang and Zuccon 2022]. Da mesma forma, os parâmetros enviesados causam o comportamento de divergência de pesos médios citado anteriormente. Ao mitigar a variância dos modelos nos clientes, espera-se aumentar a efetividade do aprendizado do sistema FL.

6. Resultados Preliminares

O estudo está em estágio inicial e atualmente concentramos nossa análise apenas na RQ1. No entanto, usaremos o modelo FedAvg como baseline, avaliando o aprimoramento dos resultados em termos de acurácia na tarefa experimental ao aplicar nossas abordagens.

6.1. Metodologia dos Experimentos

Focando só na avaliação do processo de **agregação** no FL (RQ1), configuramos uma estratégia experimental que constrói um modelo de aprendizado global, utilizando inicialmente uma tarefa de classificar imagens, em especial a base de dados MNIST [Deng 2012]. Embora classificação não seja o foco principal desta pesquisa, optamos por essa estratégia inicial pela facilidade de localizar e de executar trabalhos replicando a estratégia FedAvg na base de dados MNIST. Aqui o modelo global é obtido a partir da média dos parâmetros das redes neurais treinadas para classificar imagens nos clientes.

O dataset MNIST consiste de 60.000 imagens [Deng 2012]. Essas imagens foram usadas como entrada para a tarefa de classificação. O classificador obtido pelo FL foi o modelo global resultante das agregações sucessivas dos modelos locais dos clientes, selecionados aleatoriamente para cada rodada de treino. Seguindo as mesmas estratégias em [McMahan et al. 2023] o dataset foi devidamente particionado de modo aleatório entre 100 clientes, identificados de 0 a 99. A distribuição das imagens foi realizada indexando-as de modo aleatório com os números de 0 a 99, correspondente ao cliente proprietário. Ou seja, as imagens foram não identicamente distribuídas entre os clientes devido à aleatoriedade das associações cliente-imagem. Desta forma foi simulada natureza "não-IID" típica do aprendizado federado.

Estudamos o desempenho do FedZscore através de um projeto fatorial $2^k r$ seguindo o padrão de projeto fatorial descrito em [Jain 1991], sendo k o número de fatores e r o número de replicações para o experimento. Para este estudo tomamos $k = 3$ e $r = 3$. O objetivo era analisar o impacto dos seguintes fatores na efetividade dos modelos: a quantidade de clientes usados em cada rodada de treino, a quantidade de épocas locais em cada cliente e o número de interações realizadas por usuário em cada dispositivo local antes que seus parâmetros sejam enviados para agregação no servidor.

6.2. Resultados Experimentais

Para analisar o impacto do uso do fator "z-score" antes do processo de agregação, comparamos a eficiência média em 10 execuções dos dois algoritmos, descrito na tabela 1. Foi avaliada a variação nos resultados do modelo FedZscore com respeito aos três fatores:

quantidade de clientes, quantidade de épocas e o número de interações. Aplicamos o teste-t com 95% de confiança para calcular os intervalos de confiança (IC) para as acurácias médias (última coluna da tabela 1). A tabela 2 mostra os resultados da avaliação destes fatores e das interações entre eles. Ainda avaliamos um dos fatores isoladamente, o número de clientes em cada rodada de treino, para entender o quanto esta variação afeta o desempenho e o quão significativo é este fator para os dois algoritmos, FedAvg e FedZscore.

Execuções	1	2	3	4	5	6	7	8	9	10	IC Média
FedAvg	92.5	91.6	94.6	93.5	92.6	92.4	27.9	27.8	93.5	92.8	(60.3 , 99.6)
FedZscore	33.6	31.8	57.1	22.1	41.4	52.4	36.9	33.2	57.3	51.4	(39.0 , 44.4)

Table 1. Acurácias nos dados de teste em 10 execuções de FedAvg e FedZscore na base MNIST.

Os resultados apresentados na tabela 1 comprovaram que efetuar ajustes nos pesos das redes aplicando a padronização definida pelo "z-score" não melhora o desempenho do modelo global obtido pela agregação. De fato a acurácia média do modelo proposto, FedZscore, ficou no intervalo (39.0 , 44.4) enquanto que a do FedAvg ficou entre 60.3 e 99.6. Nossa intuição para esse resultado é que não é trivial fazer alterações de forma conjunta entre os pesos da rede, principalmente considerando que no processo de *back-propagation* cada parâmetro é ajustado independentemente. Ou seja, ao longo do treinamento os parâmetros vão se ajustando, alguns mais próximos de 1 e outros mais próximos de 0, consequentemente uma manipulação de forma conjunta em todos os parâmetros pode reduzir padrões aprendidos.

A tabela 2 mostra os resultados da análise de impacto de cada fator citado no início desta seção bem como da interação entre eles. Ao que se percebe, as interações locais possuem maior impacto sobre os resultados. Algo que faz sentido, pois a cada passo de interação o modelo compartilha e recebe novas atualizações dos parâmetros. Confirmando a premissa do impacto do modelo central sobre os modelos clientes. Ao mesmo tempo nos motivando sobre como melhorar o tratamento dos parâmetros no modelo central.

Fator	A (nº de clientes)	B (épocas locais)	C (interações locais)	AB	AC	BC
Explicação	0.1%	9.3%	27.2%	2.1%	5.9%	0.1%

Table 2. Variação de fatores para FedZscore.

Os fatores analisados isoladamente assim como suas interações por pares (AB, AC, BC), têm pouco impacto na variação do modelo, com o fator C sendo o mais influente.

Por fim, avaliamos como o número de clientes nos treinos pode afetar o desempenho dos dois algoritmos, estimando a variação nos resultados explicada pelo fator em ambos os casos. A tabela 3 mostra as acurácias obtidas em 5 replicações e 5 níveis de quantidade de clientes por treino (5, 10, 20, 50 e 100 clientes). A tabela exibe as médias gerais das acurácias para os dois algoritmos, sendo 93.6 para FedAvg e 55.6 para FedZscore. Vemos ainda que o fator número de clientes explica 62.9% da variação no modelo para FedAvg e apenas 22.9% para FedZscore. Os resultados desses últimos experimentos mostraram que, para o modelo FedZscore, fator número de clientes pode não ser a melhor escolha para realizar os experimentos deste trabalho.

7. Conclusões e Futuros Passos

Neste trabalho propomos o uso de métricas de sensibilidade ao risco para melhorar modelos de aprendizado federado. Mesmo sendo estudos iniciais, realizamos experimentos

		FedAvg					FedZscore				
k		5	10	20	50	100	5	10	20	50	100
r	1	78.1	94.4	95.7	97.1	96.7	32.2	40.3	53.5	52.8	52.6
	2	80.7	93.9	95.2	96.6	96.6	58.6	58.0	51.2	74.7	49.2
	3	86.3	90.2	92.7	96.6	97.2	50.6	66.1	50.3	53.5	45.2
	4	91.8	96.1	92.9	96.4	96.6	37.8	57.8	72.1	75.3	61.8
	5	94.3	94.4	95.7	96.5	96.7	62.3	60.5	43.4	65.6	64.3
Média Geral						93.6					55.6
Variação						**62.9%**					**22.9%**

Table 3. Acurácias para FedAvg e FedZscore com *k* clientes (5 a 100) em cada replicação *r (1 a 5)*. FedZscore explica apenas 23% da variação enquanto FedAvg explica quase três vezes mais, 63%.

com diferentes estratégias de agregação no servidor central, incluindo a normalização z-score. Descobrimos que o z-score pode não melhorar o desempenho do modelo global. Hipotetizamos que isso ocorre porque os pesos da rede mudam de maneira não uniforme entre 0 e 1, sem seguir uma tendência clara de máximo e mínimo, e a padronização pode reduzir informações importantes obtidas durante o ajuste da rede.

Nos próximos passos, exploraremos outras abordagens de agregação em FL para responder à RQ1, como ponderação por sensibilidade ao risco e por distância entre os parâmetros a média deles. O foco seguinte será melhorar o processo de aprendizado nos clientes, determinando o fator de qualidade do modelo local, que será aplicado na agregação, ponderando os parâmetros, objetivando responder às RQ2 e RQ3.

References

Ai, Q., Bi, K., Guo, J., and Croft, W. B. (2018). Learning a deep listwise context model for ranking refinement. In *ACM SIGIR conference*, pages 135–144.

Deng, L. (2012). The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6):141–142.

Divi, S., Lin, Y.-S., Farrukh, H., and Celik, Z. B. (2021). New metrics to evaluate the performance and fairness of personalized federated learning.

Jain, R. (1991). *The Art of Systems Performance Analysis: Techniques for experimental design, Measurement, simulation, and modeling*. John Wiley amp; Sons.

Jiang, J. C., Kantarci, B., Oktug, S., and Soyata, T. (2020). Federated learning in smart city sensing: Challenges and opportunities. *Sensors*, 20(21):6230.

Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, et al. (2021). Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210.

McMahan, H. B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. (2023). Communication-efficient learning of deep networks from decentralized data.

Mukut, S., Kakoli, G., and Jyotika, B. (2012). Federated search: An information retrieval strategy for scholarly literature.

Silva Rodrigues, P. H., Xavier Sousa, D., Couto Rosa, T., and Gonçalves, M. A. (2022). Risk-sensitive deep neural learning to rank. In *ACM SIGIR Conference*, SIGIR '22, page 803–813.

Wang, S. and Zuccon, G. (2022). Is non-iid data a threat in federated online learning to rank? In *ACM SIGIR Conference*, SIGIR '22, page 2801–2813.

Ye, Y., Li, S., Liu, F., Tang, Y., and Hu, W. (2020). Edgefed: Optimized federated learning based on edge computing. *IEEE Access*, 8:209191–209198.