Seleção de Modelo de Aprendizado Federado Baseado em Busca e Poda para Detecção de Defeitos Industriais*

Luana Gantert e Miguel Elias M. Campista

[1]Universidade Federal do Rio de Janeiro
PEE-COPPE/DEL-POLI/GTA

{gantert,miguel}@gta.ufrj.br

Abstract. *Federated learning emerges as an alternative to traditional machine learning by decentralizing model training. Client devices communicate with a central server and train the defined model iteratively. The training model definition in advance, however, is a challenge that is rarely discussed. This work proposes selecting the best neural network based on a search procedure involving multiple networks and subsequent pruning of those that appear less promising during training. To do this, in a given evaluation round, the nodes send the performance results to a comparator node using the AUC-ROC metric (Area Under The Receiver Operating Characteristics Curve) for best-model selection. Our experiments use an application for detecting malfunctions in industrial equipment using the emitted sounds. The results demonstrate that the models found when performing pruning in the* 5 *and* 10 *round of training reach final AUC-ROC values of approximately* 0.95 *and* 0.91 *in the best-case scenario, respectively. These values represent an increase of up to* 6.25% *over federated learning without pruning and* 18.75% *over traditional centralized learning.*

Resumo. *O aprendizado federado surge como uma alternativa ao aprendizado de máquina tradicional ao descentralizar o treinamento dos modelos. Os dispositivos clientes se comunicam com um servidor central e treinam o modelo definido de maneira iterativa. A definição antecipada do modelo a ser treinado, porém, é um desafio pouco discutido. Este trabalho propõe a seleção da melhor rede neural a partir de um procedimento de busca envolvendo múltiplas redes, e posterior poda daquelas que se mostrarem menos promissoras durante o treinamento. Para isso, em uma dada rodada de avaliação, os nós enviam os resultados de desempenho a um nó comparador que utiliza a métrica AUC-ROC (Area Under The Receiver Operating Characteristics Curve) para seleção do melhor modelo. Os experimentos utilizam uma aplicação de detecção de mal funcionamento de equipamentos industriais através dos sons emitidos. Os resultados demonstram que os modelos encontrados ao executar a poda na rodada* 5 *e* 10 *do treinamento atingem valores finais de AUC-ROC de aproximadamente* $0,95$ *e* $0,91$ *no melhor cenário, respectivamente. Esses valores representam um aumento de até* $6,25\%$ *em relação ao aprendizado federado sem poda e* $18,75\%$ *em relação ao aprendizado centralizado tradicional.*

*O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nıvel Superior - Brasil (CAPES) - Código de Financiamento 001. O trabalho também foi financiado pelo CNPq, pela FAPERJ com os auxílios E-26/211.144/2019 and E-26/200.380/2023.

1. Introdução

O aprendizado federado proposto por McMahan et al. baseia-se em treinamento distribuído de modelos de aprendizado de máquina, sem que dados sensíveis dos clientes sejam enviados ao servidor central [McMahan et al. 2017]. A ideia é manter a privacidade dos clientes [Neto et al. 2020] ao treinar os modelos localmente e apenas enviar ao servidor os parâmetros obtidos para agregação e geração de um modelo global de forma iterativa. Dessa forma, o treinamento ocorre atualizando e redistribuindo o modelo global, considerando os parâmetros obtidos pelo treinamento dos clientes em seus dados locais. Com a colaboração dos clientes durante a etapa de treinamento, além das vantagens relacionadas à privacidade, é possível aumentar a quantidade de dados disponíveis para treinamento, melhorando o desempenho dos modelos obtidos. Assim, as técnicas de aprendizado federado contribuem com aplicações já bem conhecidas do aprendizado centralizado, como as baseadas em processamento natural de linguagem, processamento de vídeo e de áudio [Li et al. 2020, Ramos et al. 2021].

Entretanto, um dos desafios do aprendizado federado consiste em treinar de forma eficiente os modelos considerando tanto a heterogeneidade dos clientes quanto a dos dados que estes produzem. Para lidar com esse problema, técnicas de seleção de clientes são aplicadas para que grupos com maior propensão a contribuir com a convergência do modelo global permaneçam no treinamento [Fu et al. 2023, de Souza et al. 2022]. Essa abordagem, porém, considera que o servidor do aprendizado federado detém informações que permitam a seleção de clientes e, mais ainda, do melhor modelo a ser treinado. Essa premissa negligencia a possibilidade da existência de um modelo que apresente um melhor desempenho com arquitetura ou configuração diferente da pré-definida, ao menos para um subconjunto de clientes. A escolha do melhor modelo para uma determinada aplicação (ou conjunto de dados) é, portanto, uma tarefa que tipicamente não inclui a participação dos clientes, sendo estes os principais interessados.

Apesar das aplicações do aprendizado de máquina serem diversas, este trabalho foca no cenário industrial e, em especial, na manutenção reativa de equipamentos. Esse tema é de especial interesse, tendo em vista as inúmeras aplicações existentes com a modernização industrial ou indústria 4.0. Note que trabalhos anteriores já comprovaram a viabilidade da adoção de modelos baseados em aprendizado de máquina centralizado para a detecção de defeitos de equipamentos industriais. As abordagens podem ser baseadas em características espectrais do som emitido pelos equipamentos [Natesha e Guddeti 2021, Gantert et al. 2021, Gantert et al. 2022], que possibilitam a redução da complexidade dos modelos em comparação com abordagens baseadas em Mel-Espectrogramas [Suefusa et al. 2020, Tao et al. 2023]. Assim, a extração das características do som tornam-se atrativas também em aplicações de aprendizado federado ao reduzir o tempo de treinamento dos modelos locais e, consequentemente, do modelo global. Mesmo usando características espectrais, a seleção prévia do modelo de rede neural mais adequado para cada par equipamento/cliente persiste. Em trabalhos da área, os clientes partem de um modelo pré-determinado pelo servidor para treinamento da rede neural com possível seleção de clientes. A abordagem típica é avaliar o desempenho de múltiplos modelos para obter o que apresenta o melhor desempenho. Tal procedimento não se traduz em um procedimento de busca realizado durante o treinamento federado.

Este trabalho propõe a seleção do modelo de rede neural mais adequado aos dados

de cada cliente em um ambiente de processamento de áudio em plantas industriais. Na proposta, múltiplos servidores possuem modelos distintos que serão inicialmente treinados por todos os clientes conhecidos. Em dada rodada de avaliação os clientes reportam o desempenho alcançado com cada um dos modelos em seu conjunto de teste a um nó centralizador, chamado de nó comparador. Esse nó, por sua vez, avalia o melhor modelo para cada um dos clientes e poda a participação destes clientes no treinamento de todos os modelos que apresentarem desempenho inferior. O conceito de poda neste trabalho resulta, portanto, na manutenção dos clientes no procedimento de treinamento apenas do modelo com o melhor desempenho encontrado. A ação de poda visa reduzir o consumo computacional de cada cliente, uma vez que este é o compromisso da proposta. Quanto antes a poda for realizada, mais recursos são preservados. O nó comparador promove a poda no processo de busca do melhor modelo usando como métrica de desempenho a AUC-ROC obtida na rodada de avaliação. O conjunto de dados adotado nos experimentos é o MIMII (*Malfunctioning Industrial Machine Investigation and Inspection*) [Purohit et al. 2019], que contém amostras de som de diferentes máquinas industriais. Os resultados obtidos demonstram o aumento da métrica de desempenho nos modelos finais em comparação com o aprendizado federado sem poda de clientes e o aprendizado centralizado tradicional. Este trabalho está organizado da seguinte forma. A Seção 2 apresenta os trabalhos relacionados. A Seção 3 apresenta conceitos básicos de aprendizado federado. A Seção 4 apresenta a proposta. A Seção 5 apresenta as configurações adotadas para a etapa experimental. A Seção 6 apresenta os resultados experimentais obtidos e as análises correspondentes. Por fim, a Seção 7 apresenta as conclusões deste trabalho e os trabalhos futuros.

2. Trabalhos Relacionados

Considerando que os dispositivos clientes do aprendizado federado usualmente apresentam recursos computacionais limitados, a literatura dispõe de trabalhos que adotam técnicas de poda dos modelos de redes neurais para obter modelos mais compactos, reduzindo o tempo de treinamento e FLOPs (*FLoating-point Operations Per Second*), sem comprometer as métricas de desempenho adotadas. Além disso, a descrição do som através da adoção de suas características espectrais em abordagens centralizadas demonstra a possibilidade de realizar a classificação de defeitos usando redes com menor número de parâmetros que as usualmente adotadas com espectrogramas. Dessa forma, a classificação de sons através de algoritmos de aprendizado supervisionado, como o SVM (*Support-Vector Machine*), torna-se também possível.

2.1. Poda de parâmetros e camadas em redes neurais no aprendizado federado

Jiang *et al.* [Jiang et al. 2022] propõem o PruneFL para, de forma adaptativa, podar os parâmetros e adaptar o tamanho dos modelos durante o aprendizado com o objetivo de minimizar o tempo total de treinamento e reduzir os requisitos computacionais necessários. O sistema inicia selecionando um único cliente para podar o modelo com seus dados locais. A poda adaptativa ocorre, pois o modelo original é ajustado iterativamente removendo parâmetros ou os readicionando conforme o procedimento de poda é executado.

Lin *et al.* [Lin et al. 2022] realizam o método de poda de parâmetros redundantes. Além disso, os autores consideram o impacto das diferentes camadas da rede neural

na métrica de avaliação e, dessa forma, adotam uma heurística para determinar o nível de esparsidade das camadas. Os modelos podados alcançam resultados semelhantes aos modelos completos nos experimentos em aplicações de reconhecimento automático.

O FedNets [Alhalabi et al. 2023] gera modelos podados da rede original ResNetV2, adotando valores distintos de poda de hiperparâmetros para garantir a diversidade entre os modelos. Os clientes selecionam aleatoriamente N modelos desse conjunto, os treinam com seus dados locais e os combinam através de métodos *ensemble* para gerar um único modelo por cliente. O modelo global permanece sendo o ResNetV2 que é atualizado e pode ter seus parâmetros podados conforme o desempenho dos clientes.

Este trabalho não realiza procedimentos de poda sobre modelos pré-definidos. A ideia é selecionar um entre múltiplos modelos com potencialmente arquitetura ou configurações diferentes durante o treinamento federado. A poda consiste, portanto, na remoção dos modelos que apresentam desempenho considerado inferior. Uma posterior poda realizada como a dos trabalhos relacionados desta seção pode ser realizada de forma complementar.

2.2. Detecção de defeitos através de características do som em aprendizado centralizado

Os autores em [Natesha e Guddeti 2021] propõem um sistema de monitoramento e classificação de comportamento anômalo através das amostras sonoras adotando o conjunto de dados MIMII. Dessa forma, eles utilizam os coeficientes Mel-Cepstrais e os coeficientes de predição linear como entrada dos modelos centralizados. Redes neurais de múltiplas camadas são adotadas, i.e., os algoritmos de Floresta Aleatória (*Random Forest* – RF), Regressão Logística (*Logistic Regression* – LR), Máquina de Vetor de Suporte (*Support Vector Machine* – SVM) e *AdaBoost* (AB). Como resultado, as redes neurais de múltiplas camadas e o algoritmo AB apresentam resultados superiores nas métricas AUC-ROC e F1-Score.

Em [Gantert et al. 2021] são selecionadas características espectrais distintas, na qual a rede neural de múltiplas camadas destaca-se quando comparada com técnicas baseadas em espectrogramas e *autoencoders*. O trabalho [Gantert et al. 2022] adota o algoritmo *Super Learner* não apenas para classificação de defeitos no cenário industrial, como também para detecção de sons urbanos. A proposta adota como algoritmos a serem combinados o *AdaBoost*, o SVM, o RF, o *Naive Bayes* (NB) o o K-Vizinhos mais Próximo (*k-Nearest Neighbors* – KNN), demonstrando que ao combiná-los é possível obter uma resposta mais robusta nas classificações binárias.

Os trabalhos de detecção de anomalias baseados em sons industriais têm como ponto comum o desconhecimento do melhor modelo a ser adotado na prática. Isso faz com que esses trabalhos realizem um procedimento de avaliação de múltiplos modelos para identificação daquele que possui o melhor desempenho. Os resultados são tipicamente semelhantes, o que levanta questões sobre a validade em cenários reais que possivelmente irão apresentar características diferentes. Nesse sentido, o trabalho atual procura identificar o melhor modelo a ser utilizado levando em conta a visão dos clientes federados durante o treinamento.

3. Revisão do Aprendizado Federado

As técnicas de aprendizado de máquinas são amplamente adotadas nos mais distintos cenários como redes móveis, redes sem fio, redes industriais, redes veiculares e Internet das Coisas [Bochie et al. 2021]. No aprendizado centralizado, os modelos são treinados remotamente em um servidor utilizando os dados coletados por múltiplos clientes. Dessa forma, é possível ajustar o modelo para um cenário específico ao custo da falta de privacidade. No aprendizado federado, por outro lado, há a construção de um modelo global que atende os requisitos de privacidade dos clientes participantes, mas com o custo da execução do treinamento em dispositivos diversos. Esta seção revisa o treinamento de modelos no aprendizado federado, tendo como objetivo compará-lo com a abordagem deste trabalho.

A Figura 1 ilustra o funcionamento do treinamento de modelos no aprendizado federado. O processo inicia-se com o servidor central que distribui um modelo inicial para todos os clientes. Esse modelo é enviado na Etapa 1 e representado pela rede neural de cor cinza. Na primeira etapa, o modelo é enviado aos clientes para que seja treinado por cada um individualmente usando os seus dados locais. Essas redes treinadas localmente pelos clientes são representadas pelas redes neurais de cores vermelha, roxa e verde na Etapa 2. Em seguida, na Etapa 3, os clientes enviam ao servidor as atualizações dos parâmetros de seus modelos. Essas informações dos modelos locais são utilizadas para que, através de um algoritmo de agregação, um novo modelo global seja construído. Nos experimentos apresentados a seguir, a estratégia de agregação *Federated Averaging* (*FedAvg*) é adotada. O FedAvg utiliza uma média ponderada para cálculo dos parâmetros da rede neural global, onde o peso é proporcional à quantidade de amostras que cada cliente usa durante o seu treinamento local. Logo, a média ponderada atribui um maior grau de

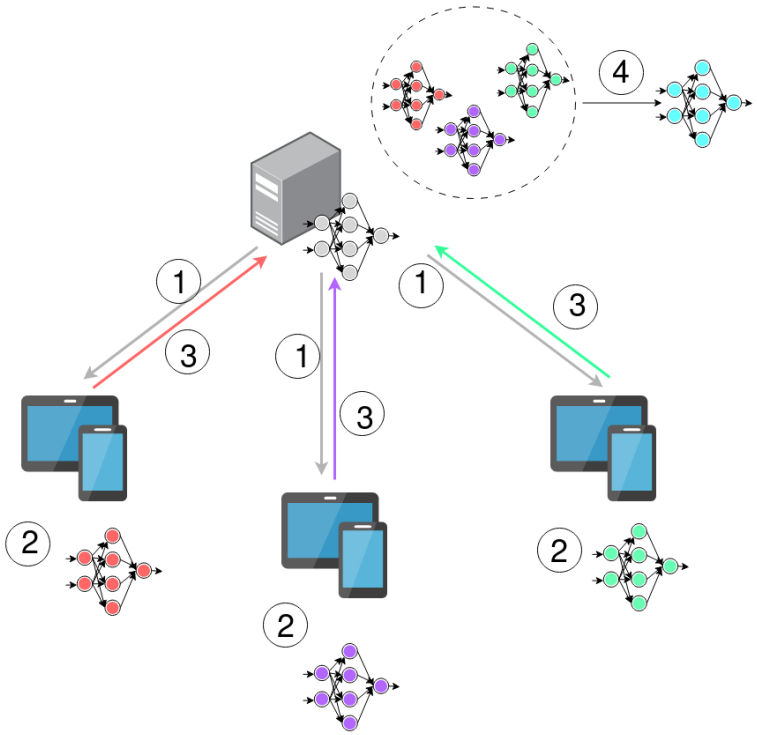


Figura 1: Etapas do procedimento de treinamento no aprendizado federado.

influência na construção do modelo global aos clientes que possuam mais amostras. Essa fase é ilustrada na figura na Etapa 4 e o modelo global é identificado pela rede neural de cor azul. As quatro etapas são repetidas até a convergência do modelo global ou até que se alcance um critério de parada pré-estabelecido.

4. Sistema de Seleção de Modelo Baseado em Busca e Poda

O sistema proposto a princípio realiza o treinamento de mais de um modelo de aprendizado federado em paralelo e permite que cada cliente participe de todos os procedimentos em execução. Sendo assim, o sistema promove inicialmente um procedimento de busca do melhor modelo de rede neural para uma dada aplicação. Após executar o treinamento até a rodada de avaliação, os clientes identificam o modelo mais adequado às suas condições e podam da busca todos os outros modelos. A partir desse momento, o cliente permanece participando do treinamento apenas do modelo que demonstrou o melhor desempenho. O objetivo é economizar recursos dos clientes, poupando-os do treinamento de modelos que possivelmente não serão mais utilizados.

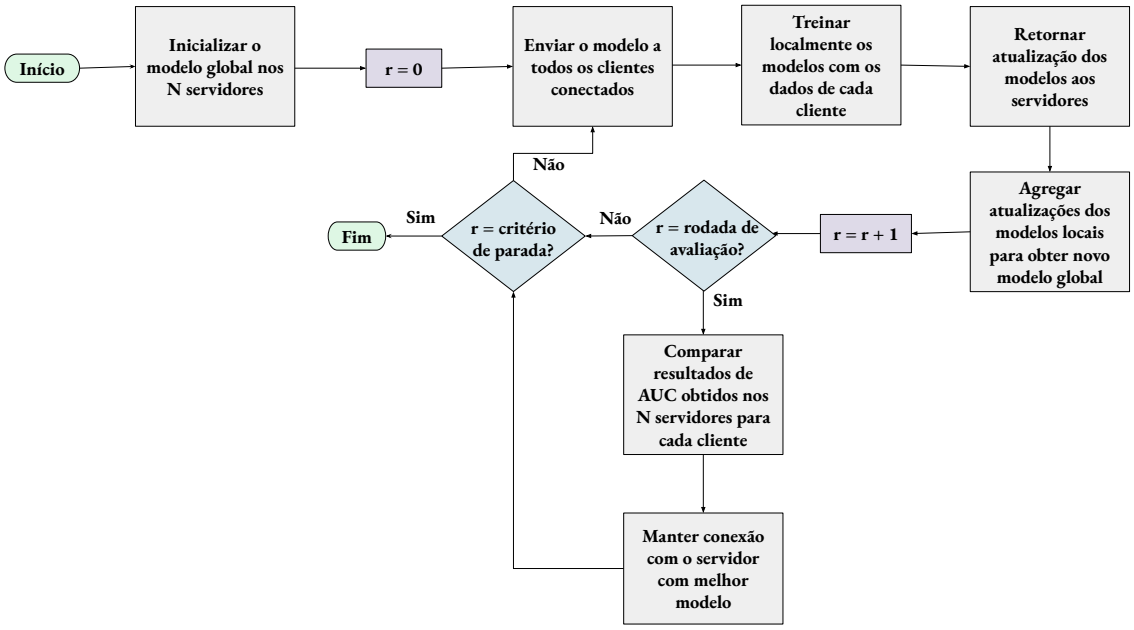


Figura 2: Fluxograma da proposta para procedimento de busca e poda executado para seleção de modelo durante treinamento no aprendizado federado.

A Figura 2 apresenta as etapas realizadas para busca e poda dos modelos conforme a proposta deste trabalho. A proposta considera um cenário com N servidores, cada um treinando um modelo distinto, e M clientes. A proposta também adota um parâmetro r que corresponde ao número da rodada atual do modelo a ser treinado. Os N modelos são inicializados nos M clientes, assim como o parâmetro r. Os M clientes treinam todos os N modelos com seus dados locais e enviam os parâmetros para agregação nos respectivos servidores. Esse processo ocorre até que um dos critérios de parada seja atingido: seja a rodada de avaliação ou o número total de rodadas de treinamento necessárias. Ambos os valores são pré-determinados antes do início da operação do sistema. Note que a proposta define múltiplos modelos e, portanto, não pré-determina qual será utilizado, como tipicamente ocorre na literatura. Ao atingir o valor definido para a rodada de avaliação,

os clientes enviam o valor da métrica AUC-ROC obtido na rodada a um nó centralizador, chamado de nó comparador. Esse nó ranqueia os valores da AUC-ROC em cada servidor por cliente e envia um comando aos clientes que indica qual treinamento deve prosseguir e quais deve encerrar. Caso a decisão do nó comparador seja de encerrar, o cliente termina a conexão com o servidor correspondente, tornando-se então indisponível para procedimentos futuros de seleção de clientes. Após a rodada de avaliação e a decisão de continuar no servidor cujo modelo é o mais promissor, o treinamento prossegue normalmente com os demais clientes que permaneceram.

A Figura 3 sintetiza a proposta do ponto de vista do cliente. As conexões com os servidores são iniciadas paralelamente e o processo de treinamento também ocorre de maneira simultânea. Na figura, esse processo é representado pelas Etapas 1, 2 e 3. Note que essas etapas são igualmente seguidas no procedimento tradicional de aprendizado federado. A diferença é a execução em paralelo de múltiplos procedimentos para o treinamento dos diferentes modelos. No exemplo, dois procedimentos de treinamento ocorrem em paralelo. Ao atingir o critério de avaliação, a métrica AUC-ROC é enviada ao nó comparador (Etapa 4) que ranqueia os resultados da rodada atual em todos os N servidores. Por fim, a decisão de manter ou encerrar o treinamento em determinado servidor é enviada aos clientes, como visto na Etapa 5.

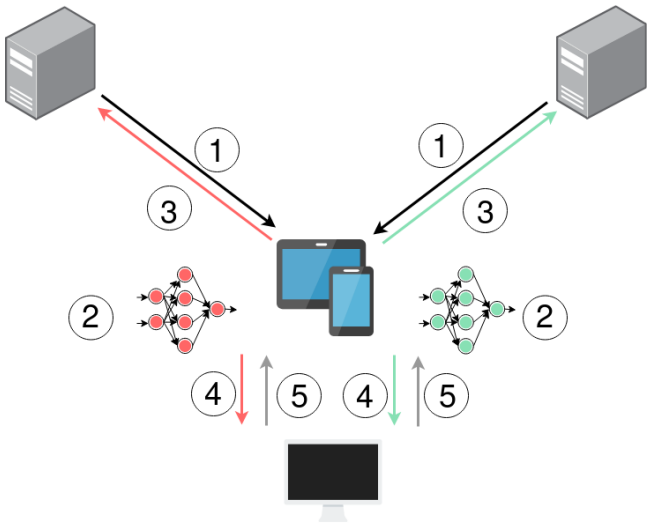


Figura 3: Proposta para seleção do melhor modelo para cada cliente do aprendizado federado.

5. Experimentos

Para etapa experimental foi utilizada uma máquina com CPU Intel Core i7-4790 e 16 GB de RAM. Esta seção apresenta o conjunto de dados e as demais configurações adotadas nos experimentos.

5.1. Conjunto de dados

O conjunto de dados MIMII é composto originalmente por amostras sonoras de quatro tipos de máquinas industriais distintas. As amostras possuem duração de 10 segundos e pertencem a duas classes distintas: sons normais e sons anômalos. Os sons

normais representam o funcionamento esperado das máquinas nas fábricas. Cada tipo de máquina é composto por quatro modelos individuais identificados pelos IDs 00, 02, 04 e 06. Os tipos de máquinas são: ventiladores industriais, bombas hidráulicas, trilhos deslizantes e válvulas solenoides. Além disso, as amostras sonoras do conjunto original estão disponíveis em três valores distintos de relação sinal-ruído: $6\,\mathrm{dB}$, $0\,\mathrm{dB}$ e $-6\,\mathrm{dB}$.

Para esta proposta, cada máquina individual do conjunto de dados original é considerada um cliente distinto. São utilizados 8 equipamentos individuais do conjunto de dados para a etapa experimental. Na Tabela 1 são apresentados os clientes adotados. Neste trabalho apenas clientes com amostras de som original com relação sinal-ruído igual a $6\,\mathrm{dB}$ são selecionados. Apenas dois tipos de equipamentos são usados para que se configure dois grupos distintos de clientes. O objetivo é verificar se o desempenho geral medido pelos clientes relacionados a amostras de equipamentos diferentes aumenta com a escolha de modelos mais apropriados.

Tabela 1: Identificação das máquinas do conjunto de dados original e sua correspondência com os clientes de aprendizado federado adotados na etapa experimental.

Máquina	ID original	ID do cliente federado
Ventilador	00	00
	02	01
	04	02
	06	03
Trilho Deslizante	00	04
	02	05
	04	06
	06	07

5.2. Características espectrais

As características do espectro sonoro são extraídas dos sons originais através da biblioteca librosa [McFee et al. 2015] para descrever de forma concisa as amostras do conjunto original. Para isso, as seguintes características foram adotadas: Croma (*Chroma*), Coeficientes Mel-cepstrais (*Mel-frequency Cepstral Coefficients*), Centroide Espectral (*Spectral Centroid*), Largura Espectral (*Spectral Band*), Decaimento Espectral (*Spectral roll-off*) e Taxa de Cruzamento por Zero (*Zero Crossing Rate*). A proposta foi implementada utilizando a biblioteca Flower [Beutel et al. 2020] para treinamento do aprendizado federado. O conjunto de dados original possui um número de amostras de estado normal maior que a de estados anômalos, dificultando o ajuste dos modelos de classificação. Para contornar este desafio, a técnica SMOTE (*Synthetic Minority Over-Sampling Technique* – Técnica de Sobreamostragem Minoritária Sintética) [Chawla et al. 2002] é adotada na etapa de treinamento para equilibrar o número de amostras.

5.3. Parâmetros e hiperparâmetros das redes neurais

Na etapa experimental, o treinamento é realizado por 50 rodadas totais, contabilizadas toda vez que ocorre a agregação de parâmetros dos modelos locais para gerar um novo modelo global. Os valores adotados separadamente para a rodada de avaliação foram $r = 5$ e $r = 10$, correspondendo à 10 e 20% do destinado ao treinamento total. As configurações das redes neurais adotadas são apresentadas na Tabela 2.

Tabela 2: Hiperparâmetros das redes neurais adotadas.

Hiperparâmetros	Rede 1 (Servidor 1)	Rede 2 (Servidor 2)
Nós por camada	(64,32,16,1)	(64,32,1)
Função de Ativação	(-, LeakyReLu, LeakyReLu, Sigmóide)	(-, ReLu, Sigmóide)
Tipo das camadas	Densas	Densas
Função de Custo	Entropia Cruzada	Entropia Cruzada
Otimizador	Adam	Adam
Total de parâmetros treináveis	4.225	3.713

6. Resultados

Para avaliação da proposta da seleção de modelos, as etapas de treinamento e teste são analisadas. Na etapa de treinamento, o percentual de memória RAM utilizada e o tempo de treinamento dos modelos federado sem e com a poda de modelos menos promissores é avaliado. Na etapa de teste, a função de custo durante as rodadas de treinamento é a métrica de desempenho AUC. Além disso, o resultado dos modelos de aprendizado federado é comparado com as respectivas redes treinadas de forma centralizada.

6.1. Etapa de treinamento dos modelos

A Figura 4 apresenta o percentual de memória RAM consumido pela máquina durante o treinamento dos modelos. Note que todos os clientes e servidores são executados em uma única máquina. Os resultados mostram que as alternativas avaliadas seguem o mesmo padrão de consumo de RAM ao utilizar entre 51 e 53% da capacidade máxima nas 5 primeiras rodadas. Ao manter todos os clientes no treinamento, a linha laranja segue uma crescente durante as 50 rodadas de comunicação. A linha roxa indica que ao realizar a poda em $r = 5$ o consumo de RAM decai nas rodadas seguintes até alcançar aproximadamente 37% da capacidade máxima. O mesmo padrão pode ser observado quando a poda é realizada em $r = 10$, como visto no resultado da linha verde. Ao selecionar apenas o melhor servidor, metade das conexões são encerradas. Isso ocorre porque cada cliente mantém a conexão apenas com um servidor dos dois inicialmente utilizados.

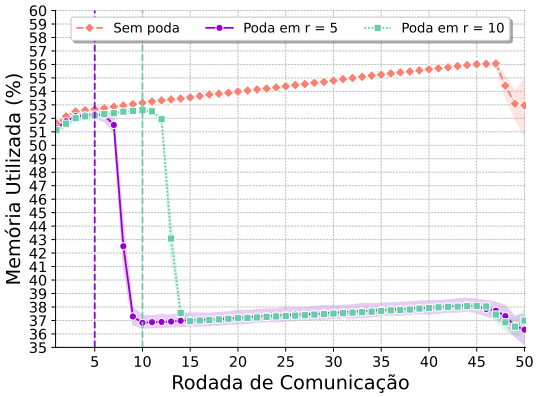
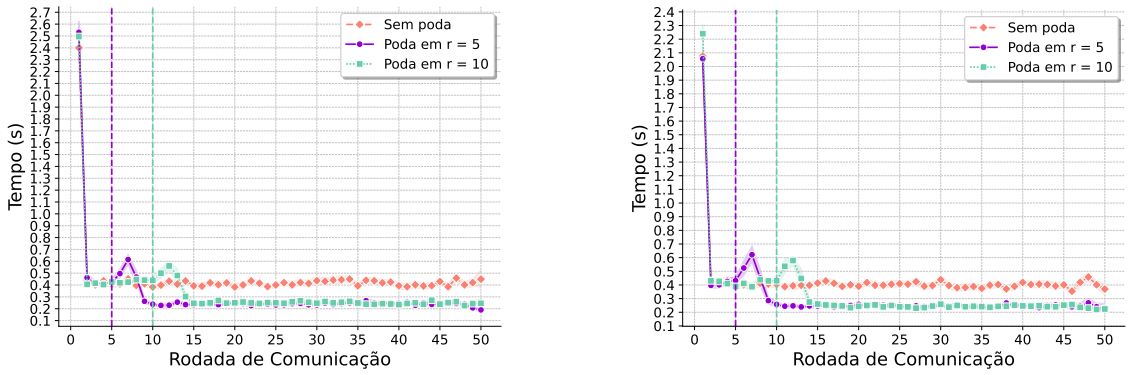


Figura 4: Memória RAM utilizada pelos modelos de aprendizado federado durante a etapa de treinamento.

A Figura 5 mostra o tempo necessário para o treinamento de cada uma das redes por rodada de treinamento. Pode-se observar que a rodada inicial é mais lenta e que há

uma queda significativa no tempo de treinamento nas rodadas subsequentes. Além disso, os resultados mostram que o tempo é proporcional à existência de clientes com dados heterogêneos. Após o procedimento de poda, cada uma das redes prossegue no procedimento de treinamento considerando apenas o subconjunto de clientes mais adequados. Ademais, a poda leva a um processo de transiente logo após a sua execução, como visto nos picos que surgem tanto nas linhas roxas quanto nas laranjas. O aumento no tempo é consequência da mudança de gradiente que ocorre ao limitar o subconjunto de clientes àqueles que possuem dados mais próximos.

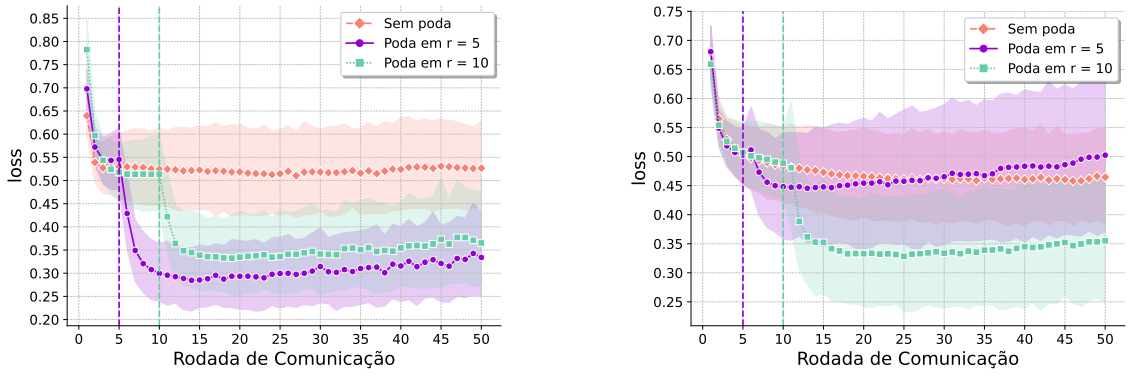


(a) Tempo necessário para treinamento da Rede 1.

(b) Tempo necessário para treinamento da Rede 2.

Figura 5: Tempo de treinamento de cada um dos modelos antes e depois do procedimento de poda.

6.2. Etapa de teste dos modelos



(a) Função de Custo da Rede 1.

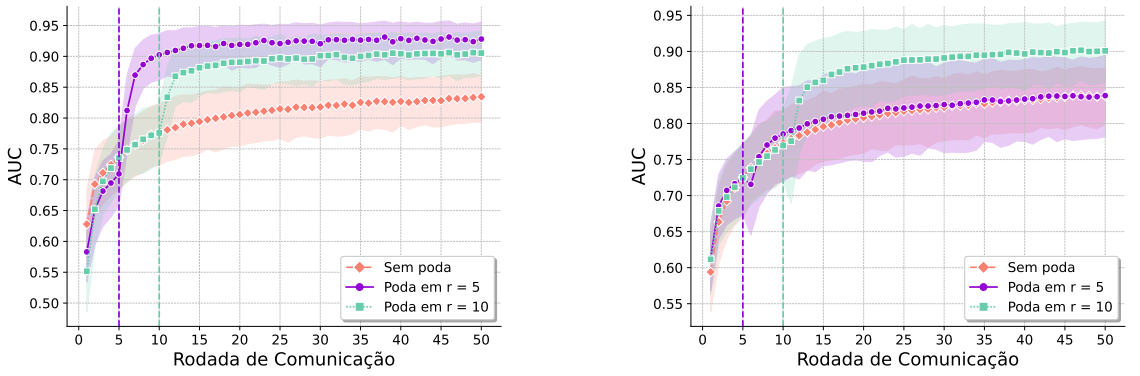
(b) Função de Custo da Rede 2.

Figura 6: Evolução da função de custo no treinamento dos modelos durante as rodadas de comunicação com o servidor considerando o conjunto de teste.

As Figuras 6a e 6b apresentam o decaimento da função de custo durante as rodadas de comunicação nos servidores 1 e 2, respectivamente. Dessa forma, observa-se que ao adotar a poda em $r = 10$, o desempenho analisado através da função de custo é superior ao cenário em que os modelos são treinados pelas 50 rodadas de comunicação pré-determinadas. Com $r = 5$ o desempenho no servidor 1 é inferior ao obtido com

$r = 10$, sendo que este comportamento não se reproduz no servidor 2. Essa inversão e o desempenho comparável ao procedimento sem poda no servidor 2 estão relacionados à seleção de clientes, como explicado em breve.

As Figuras 7a e 7b apresentam os resultados da métrica AUC-ROC na etapa de teste com os dados locais durante as rodadas de comunicação. Pode-se observar que, após as podas, o valor da métrica sofre um salto em ambos os servidores comparado ao aprendizado federado sem podas. A inversão no desempenho entre as duas opções de poda e a pequena diferença em $r = 5$ seguem o mesmo efeito ocorrido na Figura 6.



(a) AUC-ROC obtida pela Rede 1.

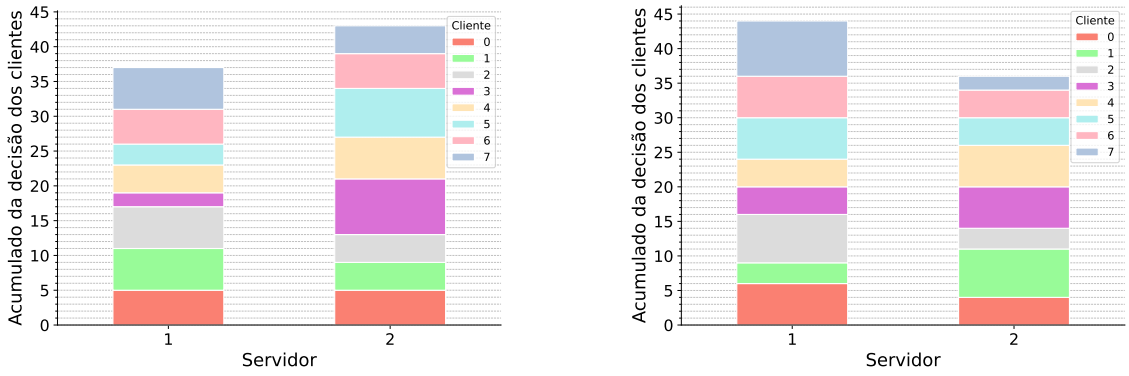
(b) AUC-ROC obtida pela Rede 2.

Figura 7: Evolução da AUC-ROC no treinamento dos modelos durante as rodadas de comunicação com o servidor considerando o conjunto de teste.

A inversão de modelos tanto na métrica AUC-ROC quanto na função de custo pode ser explicada pela necessidade distinta de mais rodadas de treinamento dos parâmetros nas duas redes treinadas. Isso ocorre, pois, através da escolha do melhor servidor, os clientes que dificultam a convergência do modelo podem encontrar o servidor mais adequado ao seu conjunto de dados. Ao comparar a poda em $r = 5$ em ambos os servidores, a rede 1 com maior número de parâmetros treináveis não realizou rodadas de treinamento suficientes para a escolha dos clientes. Em paralelo, os clientes com menor desempenho permanecem na rede 2, comprometendo o restante do treinamento. Por outro lado, ao realizar a poda em $r = 10$ a rede 2 alcança resultado superior aos demais com a métrica AUC-ROC por volta de 0.90 ao final das 50 rodadas. Apesar da rede 1 sofrer um decréscimo em sua métrica em relação à poda em $r = 5$, ao adotar um valor maior para o parâmetro r, ambas as redes superam o desempenho do aprendizado realizado com todos os clientes simultaneamente.

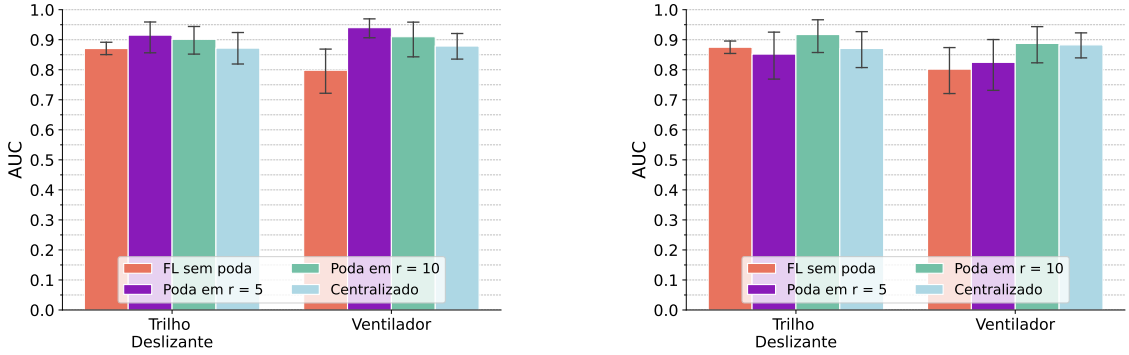
As Figuras 8a e 8b apresentam o agregado das escolhas dos clientes pelas 10 repetições independentes realizadas na etapa experimental. Ou seja, considerando que o parâmetro r é fixado, o somatório de cada cliente deve ser igual a 10. Por exemplo, em $r = 5$, o cliente 3 permaneceu no servidor 1 em duas vezes que o experimento foi realizado, e no servidor 2 nas outras oito vezes. É possível observar que em $r = 5$ a maioria dos clientes permanece o treinamento na rede 2. Por outro lado, ao aumentar o valor de r, a maioria dos clientes permanece na rede 1.

As Figuras 9a e 9b compararam a métrica AUC-ROC obtida pelos modelos finais após as 50 rodadas de comunicação e o desempenho da rede equivalente treinada através



(a) Clientes para parâmetro r = 5. **(b) Clientes para parâmetro r = 10.**

Figura 8: Clientes após as podas considerando as 10 repetições do experimento.



(a) AUC-ROC para a Rede 1/Servidor 1. **(b) AUC-ROC a Rede 2/Servidor 2.**

Figura 9: AUC-ROC obtida no aprendizado federado e centralizado

de aprendizado centralizado por 50 épocas. Pode-se observar que, para os modelos treinados, o uso da poda em $r = 10$ resultou em um desempenho superior para as duas redes. Em números relativos, após 20% das rodadas de treinamento é possível selecionar o modelo com o melhor desempenho para cada cliente, sem precisar de premissas sobre o modelo a ser treinado no aprendizado federado.

7. Conclusão e Trabalhos Futuros

Este trabalho apresentou uma proposta para seleção de modelos de aprendizado federado considerando um procedimento de busca e poda. Os clientes do aprendizado federado iniciam o procedimento de múltiplos modelos, mas após a poda permanecem participando apenas do treinamento do modelo que apresenta o melhor desempenho. Essa proposta visa complementar a literatura que tipicamente assume que o modelo a ser treinado é pré-determinado. O desempenho da proposta foi medido através da métrica AUC-ROC nos conjuntos de testes e alcançou um aumento de até 6.25% em comparação ao aprendizado federado sem a poda. Em comparação com a mesma hierarquia de rede com aprendizado centralizado, a melhoria alcançou 18.75%.

Além disso, ao analisar o impacto no treinamento dos modelos, a memória RAM percentual utilizada teve uma redução de 52% para aproximadamente 37% após a poda. O

tempo de treinamento também é reduzido, uma vez que o número de clientes participantes diminui na etapa de treinamento dos dados locais.

Como trabalhos futuros, o parâmetro r pode ser adotado de forma adaptativa considerando a quantidade de parâmetros a serem treinados. Além disso, pretende-se implementar a proposta em cenários com um número maior de clientes e servidores.

Referências

Alhalabi, B., Basurra, S., e Gaber, M. M. (2023). Fednets: Federated learning on edge devices using ensembles of pruned deep neural networks. *IEEE Access*, 11:30726–30738.

Beutel, D. J., Topal, T., Mathur, A., Qiu, X., Parcollet, T., de Gusmão, P. P., e Lane, N. D. (2020). Flower: A friendly federated learning research framework. *arXiv preprint arXiv:2007.14390*.

Bochie, K., Gilbert, M. S., Gantert, L., Barbosa, M. S., Medeiros, D. S., e Campista, M. E. M. (2021). A survey on deep learning for challenged networks: Applications and trends. *Journal of Network and Computer Applications*, 194:103213.

Chawla, N. V., Bowyer, K. W., Hall, L. O., e Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

de Souza, L. A. C., Camilo, G. F., Sammarco, M., Campista, M. E. M., e Costa, L. H. M. (2022). Aprendizado federado com agrupamento hierárquico de clientes para aumento da acurácia. In *Anais do XL Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*, pages 545–558. SBC.

Fu, L., Zhang, H., Gao, G., Zhang, M., e Liu, X. (2023). Client selection in federated learning: Principles, challenges, and opportunities. *IEEE Internet of Things Journal*.

Gantert, L., Sammarco, M., Detyniecki, M., e Campista, M. E. M. (2021). A supervised approach for corrective maintenance using spectral features from industrial sounds. In *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)*, pages 723–728. IEEE.

Gantert, L., Sammarco, M., Detyniecki, M., e Campista, M. E. M. (2022). Super learner ensemble for sound classification using spectral features. In *2022 IEEE Latin-American Conference on Communications (LATINCOM)*, pages 1–6. IEEE.

Jiang, Y., Wang, S., Valls, V., Ko, B. J., Lee, W.-H., Leung, K. K., e Tassiulas, L. (2022). Model pruning enables efficient federated learning on edge devices. *IEEE Transactions on Neural Networks and Learning Systems*.

Li, L., Fan, Y., Tse, M., e Lin, K.-Y. (2020). A review of applications in federated learning. *Computers & Industrial Engineering*, 149:106854.

Lin, R., Xiao, Y., Yang, T.-J., Zhao, D., Xiong, L., Motta, G., e Beaufays, F. (2022). Federated pruning: improving neural network efficiency with federated learning. *arXiv preprint arXiv:2209.06359*.

McFee, B., Raffel, C., Liang, D., Ellis, D. P., McVicar, M., Battenberg, E., e Nieto, O. (2015). librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8, pages 18–25.

McMahan, B., Moore, E., Ramage, D., Hampson, S., e y Arcas, B. A. (2017). Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR.

Natesha, B. e Guddeti, R. M. R. (2021). Fog-based intelligent machine malfunction monitoring system for industry 4.0. *IEEE Transactions on Industrial Informatics*, 17(12):7923–7932.

Neto, H. N., Mattos, D. M., e Fernandes, N. C. (2020). Privacidade do usuário em aprendizado colaborativo: Federated learning, da teoria à prática. *Sociedade Brasileira de Computação*.

Purohit, H., Tanabe, R., Ichige, K., Endo, T., Nikaido, Y., Suefusa, K., e Kawaguchi, Y. (2019). Mimii dataset: Sound dataset for malfunctioning industrial machine investigation and inspection. *arXiv preprint arXiv:1909.09347*.

Ramos, H. S., Maia, G., Papa, G. L., Alvim, M. S., Loureiro, A. A., Cardoso-Pereira, I., Campos, D. H., Filipakis, G., Riquetti, G., Chagas, E. T., et al. (2021). Aprendizado federado aplicado à internet das coisas. *Sociedade Brasileira de Computação*.

Suefusa, K., Nishida, T., Purohit, H., Tanabe, R., Endo, T., e Kawaguchi, Y. (2020). Anomalous sound detection based on interpolation deep neural network. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 271–275. IEEE.

Tao, B., Chen, C., e Chen, H. (2023). Communication efficient federated learning via channel-wise dynamic pruning. In *2023 IEEE Wireless Communications and Networking Conference (WCNC)*, pages 1–6. IEEE.