Survey em IA e Segurança

Davi Iury, Esther Martins, Lucas Pinheiro, Rafael Porto, and Théo Araújo

Universidade Federal do Ceará

Abstract. Nós resume as coisas aqui.

Keywords: IA · Segurança · Machine Learning.

1 Introdução

IA é muito popular. Porém, precisamos de muitos dados para treinar modelos. Como podemos arranjar esses dados? Mais especificamente, como podemos arrumar esses dados de forma que não infrijamos leis de privacidade de dados? Como privacidade de dados vem se tornando um conceito cada vez mais em voga, Novas formas de treinar modelos e obter dados vem surgindo. Nesta survey, falaremos de:
Federated learning (analisar os dados locamente e mandar os resultados de volta de forma criptografada)
Differential privacy (é uma técnica que visa proteger a privacidade dos usuários por meio da adição de ruído nos dados sendo analisados)
Machine Unlearning (esquecer dados de usuários que foram usados para treinar modelos de forma que isso não prejudique o aprendizado do algoritmo).

1.1 Contextualização

Deixei aqui para ser o template inicial. Quando forem escrever, É legal dar enter a cada oração Para que fique dividido direito e fique fácil de ler.

1.2 Critérios de busca

Google scholar com o nome dos assuntos principais escolhendo os artigos com maior número de referências. Verificando o abstract e o local de publicação para ver a validade do artigo. Procurando os artigos que esses artigos citam e aplicando o mesmo critério de citações, local de publicação e abstract. (detalhar essa seção)

2 Caracterização Ferramental

2.1 Machine unlearning

woow

2.2 Differential privacy

woow

2.3 Federated learning

Contexto. Algoritmos clássicos de aprendizado de máquina normalmente assumem que os dados de treinamento estão disponíveis de forma centralizada. Isso se deve ao fato de que, frequentemente, os dados encontram-se dispersos em *data islands*[1]. Nessas situações, é necessário coletar e consolidar os dados em um servidor central para viabilizar o treinamento. No entanto, esse processo pode resultar em riscos de vazamento de informações sensíveis se não for conduzido de forma adequada. Em vista dessa situação, diversas regulações vem sendo impostas com relação à captação e ao uso de dados para treinamento de modelos, tornando o uso de técnicas de aprendizado de máquina centralizado de difícil implementação prática. Nesse contexto, a aplicação do federated learning (aprendizado federado, ou aprendizado colaborativo) possibilita com que o treinamento seja feito de forma local, não-centralizada, de forma que os dados de um usuário específico mantém-se somente no seu dispositivo local.

Definição. Em sua essência, o federated learning é uma técnica de aprendizado distribuído, ou seja, ao invés de consolidarmos dados de usuário em um servidor central para treinar um modelo, haverão focos locais de treinamento. Assim, evitando a captação de dados sensíveis, o servidor envia um modelo de treinamento para cada dispositivo individual, mantendo a fase de treinamento como uma etapa local. Cada dispositivo somente retornará ao servidor central seu modelo treinado localmente e atualizará o seu modelo interno conforme as atualizações no modelo global. Sob essa arquitetura, os dados permanecem nos dispositivos locais e a comunicação é restrita a parâmetros ou gradientes, reduzindo o risco de exposição de dados sensíveis.

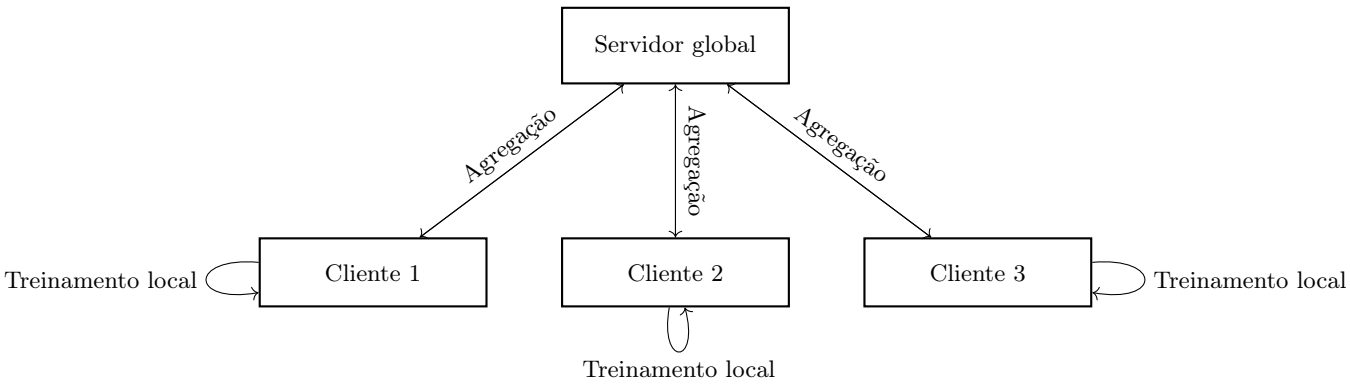

Fig. 1. Fluxo básico do aprendizado federado.

[1] Em muitos cenários reais, os dados permanecem isolados em repositórios diferentes, frequentemente chamados de *data islands* ou *data silos*, devido a restrições de privacidade, regulatórias ou organizacionais, impedindo a sua centralização para processamento de aprendizado tradicional [11].

Diferenciação.

1. **Com relação à partição dos dados.** A classificação mais tradicional do federated learning diz respeito à forma como os dados estão distribuídos entre as entidades participantes. Conforme sistematizado por Zhang et al. [11], essa categorização pode ser dividida em três cenários principais:
   a. *Aprendizado federado horizontal.* Nesse cenário, os conjuntos de dados possuem funcionalidades semelhantes, porém usuários distintos, isto é, há grande sobreposição no espaço de atributos, mas pouca interseção no espaço de instâncias. A partição ocorre ao longo das linhas da base de dados. Esse modelo é amplamente utilizado em aplicações como teclados inteligentes e sistemas de recomendação distribuídos, sendo o FedAvg [7] o algoritmo de agregação mais comum [11].
   b. *Aprendizado federado vertical.* Aqui ocorre o oposto: os usuários são majoritariamente os mesmos, enquanto as funcionalidades são distintas. A partição é feita ao longo das colunas do conjunto de dados. Esse cenário é comum em colaborações interinstitucionais, como entre bancos e empresas de e-commerce, onde diferentes entidades possuem informações complementares sobre os mesmos usuários. Nesse contexto, técnicas baseadas em criptografia homomórfica e alinhamento seguro de entidades são amplamente utilizadas[5, 3].
   c. *Aprendizado federado com transferência.* Quando há pouca sobreposição tanto de usuários quanto de funcionalidades, recorre-se a técnicas de aprendizado por transferência para viabilizar o treinamento colaborativo. Esse modelo é particularmente útil em cenários com dados escassos ou rotulados de forma incompleta, como aplicações de classificação de imagem ou modelos de aprendizado de linguagem natural [10].
2. **Com relação aos mecanismos de privacidade.** Embora o federated learning evite o compartilhamento direto dos dados brutos, a troca de parâmetros de modelo ainda pode levar ao vazamento de informações sensíveis. Dessa forma, diferentes mecanismos de privacidade são empregados:
   a. *Agregação segura de modelos.* Baseia-se na combinação de parâmetros locais sem que o servidor tenha acesso às atualizações individuais. Protocolos de *secure aggregation* garantem que apenas o modelo global agregado seja observável [2].
   b. *Criptografia homomórfica.* Permite que operações matemáticas sejam realizadas diretamente sobre dados criptografados, garantindo que nem o servidor nem outros participantes tenham acesso às informações originais. Esse método é comum em aprendizado federado vertical [5, 11].
   c. *Privacidade diferencial.* Consiste na adição controlada de ruído estatístico aos gradientes ou parâmetros do modelo, limitando a possibilidade de inferência sobre dados individuais. Pode ser aplicada tanto local quanto globalmente e é amplamente utilizada em sistemas reais, como os propostos e estudados pelo Google [8, 1].
3. **Com relação ao modelo de aprendizado de máquina aplicado.** O federated learning não se restringe a um tipo específico de modelo, sendo aplicável a diferentes classes de algoritmos de aprendizado de máquina:

a. *Modelos lineares.* Incluem regressão linear, regressão logística e ridge regression. Modelos lineares, como regressão linear e ridge regression, são frequentemente utilizados como ponto de partida em ambientes federados devido à sua simplicidade algorítmica e eficiência computacional, especialmente em cenários com restrições de privacidade e comunicação [9, 11].
b. *Modelos baseados em árvores.* Árvores de decisão, random forests e gradient boosting têm sido adaptados para o contexto federado, especialmente em ambientes verticais. O SecureBoost é um exemplo representativo desse tipo de abordagem [3].
c. *Redes neurais profundas.* Redes neurais são amplamente utilizadas em aplicações modernas de federated learning, como reconhecimento de voz, visão computacional e sistemas embarcados. Frameworks baseados em FedAvg permitem o treinamento eficiente de redes profundas em larga escala [7, 1].

4. **Com relação aos métodos de tratamento da heterogeneidade.** A heterogeneidade dos dados e dos recursos de clientes é um dos principais desafios inerente aos ambientes distribuídos, que pode manifestar-se tanto na forma de dados estatisticamente não-IID[2] quanto em diferenças computacionais, de disponibilidade e de conectividade entre os dispositivos participantes. Diversos mecanismos têm sido propostos na literatura para mitigar esses efeitos adversos:
   a. *Comunicação assíncrona.* Abordagens assíncronas permitem que o servidor agregue atualizações de clientes à medida que elas se tornam disponíveis, sem a necessidade de sincronização global entre todos os participantes. Esse modelo reduz o impacto de *stragglers* e dispositivos com conectividade intermitente, sendo particularmente adequado para cenários em larga escala e ambientes móveis [11].
   b. *Amostragem de clientes.* Em sistemas federados reais, apenas um subconjunto dos clientes disponíveis participa de cada rodada de treinamento. A amostragem de clientes reduz significativamente os custos de comunicação e computação, além de tornar o processo mais robusto à heterogeneidade de recursos. Esse mecanismo é parte fundamental do algoritmo FedAvg e de seus desdobramentos [7].
   c. *Mecanismos tolerantes a falhas.* Falhas de comunicação, desconexões inesperadas e clientes lentos são comuns em ambientes federados. Métodos tolerantes a falhas buscam garantir a continuidade do treinamento e preservar propriedades de convergência mesmo na presença de clientes ausentes ou atualizações perdidas. Embora fora do escopo do aprendizado federado supervisionado tradicional, trabalhos em Federated Re-

[2] dados não-IID violam uma ou ambas dessas seguintes propriedades: independência e distribuição idêntica. Assim, dados não-IID podem ser um sinal de que ou há viés no conjunto de dados, ou de que os dados não seguem a mesma distribuição, ou de que há correlação local entre os dados. Uma vez que cada cliente coleta dados de acordo com comportamentos e contextos específicos, resultando em distribuições distintas entre dispositivos, dados não-IID são um comum em aprendizado federado.

inforcement Learning (FRL), como o de Fan et al. [4], formalizam esse problema e propõem estratégias que mantêm a estabilidade do treinamento sob diferentes modelos de falha.

d. *Heterogeneidade de modelos e aprendizado personalizado.* Para lidar com distribuições de dados altamente não-IID, abordagens de aprendizado federado personalizado permitem a coexistência de modelos locais adaptados a cada cliente juntamente com um modelo global compartilhado. O trabalho de Liang et al. [6] propõe a separação entre representações globais e locais, reduzindo o impacto da heterogeneidade estatística e melhorando o desempenho individual dos clientes.

3 Desafios

References

1. Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konečný, J., Mazzocchi, S., McMahan, H.B., Overveldt, T.V., Petrou, D., Ramage, D., Roselander, J.: Towards federated learning at scale: System design (2019), https://arxiv.org/abs/1902.01046
2. Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H.B., Patel, S., Ramage, D., Segal, A., Seth, K.: Practical secure aggregation for privacy-preserving machine learning. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. pp. 1175–1191. CCS '17, Association for Computing Machinery, New York, NY, USA (2017). https://doi.org/10.1145/3133956.3133982, https://doi.org/10.1145/3133956.3133982
3. Cheng, K., Fan, T., Jin, Y., Liu, Y., Chen, T., Papadopoulos, D., Yang, Q.: Secureboost: A lossless federated learning framework (2021), https://arxiv.org/abs/1901.08755
4. Fan, F.X., Ma, Y., Dai, Z., Jing, W., Tan, C., Low, B.K.H.: Fault-tolerant federated reinforcement learning with theoretical guarantee (2022), https://arxiv.org/abs/2110.14074
5. Hardy, S., Henecka, W., Ivey-Law, H., Nock, R., Patrini, G., Smith, G., Thorne, B.: Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption (2017), https://arxiv.org/abs/1711.10677
6. Liang, P.P., Liu, T., Ziyin, L., Allen, N.B., Auerbach, R.P., Brent, D., Salakhutdinov, R., Morency, L.P.: Think locally, act globally: Federated learning with local and global representations (2020), https://arxiv.org/abs/2001.01523
7. McMahan, H.B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data (2023), https://arxiv.org/abs/1602.05629
8. McMahan, H.B., Ramage, D., Talwar, K., Zhang, L.: Learning differentially private recurrent language models (2018), https://arxiv.org/abs/1710.06963
9. Nikolaenko, V., Weinsberg, U., Ioannidis, S., Joye, M., Boneh, D., Taft, N.: Privacy-preserving ridge regression on hundreds of millions of records. In: Security and Privacy. pp. 334–348 (05 2013). https://doi.org/10.1109/SP.2013.30
10. Yang, Q., Liu, Y., Cheng, Y., Kang, Y., Chen, T., Yu, H.: Federated Transfer Learning, pp. 83–93. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-031-01585-4_6

11. Zhang, C., Xie, Y., Bai, H., Yu, B., Li, W., Gao, Y.: A survey on federated learning. Knowledge-Based Systems **216**, 106775 (2021). https://doi.org/https://doi.org/10.1016/j.knosys.2021.106775, https://www.sciencedirect.com/science/article/pii/S0950705121000381