

Técnicas de Preservação de Privacidade em Inteligência Artificial: Uma Revisão Sistemática

Davi Souza, Esther Martins, Lucas Pinheiro, Rafael Porto, e Théo Araújo

Universidade Federal do Ceará

Abstract. A crescente dependência de sistemas de Inteligência Artificial em diversos setores da sociedade trouxe à tona questões fundamentais sobre a proteção de dados pessoais e a conformidade com regulamentações de privacidade. Este trabalho apresenta uma revisão sistemática das principais técnicas de preservação de privacidade aplicadas ao contexto de aprendizado de máquina, explorando *federated learning*, *differential privacy*, *machine unlearning* e geração de dados sintéticos. Analisamos as características, aplicações e limitações de cada abordagem, bem como suas sinergias e desafios de implementação prática. Os resultados demonstram que a combinação dessas técnicas oferece um conjunto robusto de ferramentas para diferentes cenários e requisitos de privacidade, embora desafios relacionados ao trade-off entre utilidade e privacidade permaneçam em aberto.

Keywords: Federated Learning · Differential Privacy · Machine Unlearning · Dados Sintéticos · Privacidade · Segurança em IA

1 Introdução

O aprendizado de máquina tem se consolidado como uma das áreas mais promissoras da Inteligência Artificial, impulsionando avanços significativos em reconhecimento de padrões, processamento de linguagem natural, visão computacional e sistemas de recomendação. No entanto, o sucesso desses modelos depende fundamentalmente da disponibilidade de grandes volumes de dados de treinamento, frequentemente contendo informações sensíveis sobre indivíduos e organizações. Essa dependência levanta questões críticas sobre como conciliar o desenvolvimento de sistemas de IA cada vez mais sofisticados com a necessidade de proteger a privacidade dos dados utilizados em seu treinamento.

Em vista dessa situação, diversas regulamentações vêm sendo impostas globalmente para garantir a proteção de dados pessoais, como o Regulamento Geral de Proteção de Dados (GDPR) na União Europeia e a Lei Geral de Proteção de Dados (LGPD) no Brasil. Essas legislações estabelecem princípios rigorosos sobre coleta, armazenamento e processamento de dados, incluindo o direito ao esquecimento e à portabilidade de dados, tornando o uso de técnicas tradicionais de aprendizado de máquina centralizado de difícil implementação prática. Nesse contexto, novas abordagens técnicas vêm surgindo para viabilizar o treinamento

de modelos de forma que a privacidade dos dados seja preservada desde a concepção do sistema.

Este trabalho apresenta uma revisão sistemática das principais técnicas de preservação de privacidade aplicadas ao contexto de aprendizado de máquina. Em particular, exploramos quatro abordagens fundamentais: aprendizado federado [18, 26], privacidade diferencial [19], desaprendizado de máquina, e geração de dados sintéticos [10, 20, 17]. Ao longo deste survey, analisamos as características, aplicações e limitações de cada técnica, bem como suas sinergias e desafios de implementação prática.

1.1 Contextualização

Federated Learning. O aprendizado federado [18, 26] permite que múltiplos dispositivos ou organizações colaborem no treinamento de um modelo global sem compartilhar seus dados brutos. Ao manter os dados localmente e transmitir apenas atualizações de modelo, essa abordagem reduz significativamente os riscos de vazamento de informações sensíveis e viabiliza o treinamento em cenários onde a centralização de dados é inviável ou indesejável.

Differential Privacy. A privacidade diferencial [19] fornece garantias formais e quantificáveis sobre o nível de privacidade preservado durante o treinamento de modelos. Através da adição controlada de ruído estatístico aos dados ou aos parâmetros do modelo, essa técnica limita a capacidade de adversários inferirem informações sobre indivíduos específicos, mesmo quando têm acesso ao modelo treinado.

Machine Unlearning. O desaprendizado de máquina responde à demanda regulatória e ética do direito ao esquecimento, permitindo remover seletivamente a influência de dados específicos de modelos já treinados sem comprometer significativamente seu desempenho. Essa capacidade é essencial para conformidade com regulamentações que garantem aos usuários o direito de solicitar a exclusão de suas informações pessoais.

Geração de Dados Sintéticos. A geração de dados sintéticos [10, 20, 17] utiliza técnicas de IA generativa para produzir dados artificiais que preservam propriedades estatísticas relevantes sem expor informações individuais. Além de facilitar o compartilhamento de dados para pesquisa e desenvolvimento, essa abordagem oferece soluções para problemas de escassez de dados e desbalanceamento de classes.

1.2 Metodologia de Revisão Sistemática

Esta revisão sistemática foi conduzida com o objetivo de identificar e analisar as principais técnicas de preservação de privacidade em sistemas de Inteligência Artificial. A metodologia adotada seguiu um processo estruturado de busca, seleção e análise de trabalhos científicos, priorizando publicações de alto impacto e relevância para o tema.

Bases de dados consultadas. A busca bibliográfica foi realizada nas seguintes bases de dados acadêmicas:

- *Google Scholar* (base principal)
- *IEEE Xplore Digital Library*
- *ACM Digital Library*
- *arXiv*
- *Springer Link*

Estratégia de busca. Foram utilizadas combinações de termos-chave relacionados às quatro áreas principais de interesse: *federated learning*, *differential privacy*, *machine unlearning* e *synthetic data generation*. A busca priorizou artigos com alto número de citações (preferencialmente acima de 100 citações), indicando reconhecimento e impacto na comunidade científica. Além disso, foram considerados trabalhos publicados em conferências e periódicos de alto impacto, como *IEEE Symposium on Security and Privacy*, *ACM Conference on Computer and Communications Security* (CCS), e periódicos como *Knowledge-Based Systems*, *IEEE Access* e *Electronics*.

Critérios de seleção. Os artigos foram selecionados com base nos seguintes critérios:

1. *Relevância temática*: O trabalho deve abordar diretamente técnicas de preservação de privacidade em aprendizado de máquina.
2. *Qualidade da publicação*: Preferência por artigos publicados em venues de alto impacto (IEEE, ACM, Springer) ou repositórios reconhecidos (arXiv).
3. *Impacto científico*: Número de citações como indicador de relevância e influência na área.
4. *Contribuição técnica*: Trabalhos que apresentam métodos inovadores, análises abrangentes ou implementações práticas.
5. *Atualidade*: Priorização de trabalhos recentes (2017 – 2025), embora trabalhos seminais mais antigos também tenham sido incluídos.

Processo de triagem. O processo de seleção foi conduzido em três etapas:

1. *Busca inicial*: Aproximadamente 50 artigos foram identificados com base nos termos de busca e filtros de citação.
2. *Análise de abstract e metadados*: Os abstracts e informações de publicação foram analisados para verificar relevância e qualidade, resultando em 30 artigos pré-selecionados.
3. *Leitura completa e rastreamento de referências*: Os artigos pré-selecionados foram lidos integralmente, e suas referências foram rastreadas para identificar trabalhos adicionais relevantes. Após essa etapa, 26 artigos foram selecionados para compor a base bibliográfica final deste *survey*.

Trabalhos selecionados. A lista final de 26 referências inclui trabalhos seminais e *surveys* abrangentes em cada uma das cinco áreas principais:

- *Machine Unlearning*: Bourtoule et al. [3] introduzem, definem e conceituam acerca de *Machine Unlearning*, sendo este o principal artigo usado no direcionamento da pesquisa para esta *survey*.

- *Federated Learning*: Zhang et al. [26] apresentam um *survey* abrangente sobre aprendizado federado; McMahan et al. [18] propõem o algoritmo FedAvg; Bonawitz et al. [2, 1] desenvolvem protocolos de agregação segura e sistemas em larga escala.
- *Differential Privacy*: McMahan et al. [19] aplicam privacidade diferencial a modelos de linguagem recorrentes; Dwork et al. [23] fornece conceitos gerais para o entendimento da differential privacy; [15] aplicações práticas de differential privacy; [6] fundamentação algorítmica da differential privacy; [7] aprofunda o parâmetro delta e o conceito de falha de privacidade; [24] aprofundamento do tema de local differential privacy; [4] foca nos riscos de privacidade em machine unlearning.
- *Federated Learning Avançado*: Hardy et al. [12] exploram aprendizado federado vertical com criptografia homomórfica; Cheng et al. [5] propõem SecureBoost; Liang et al. [16] abordam representações locais e globais; Fan et al. [8] tratam tolerância a falhas; Yang et al. [25] discutem transfer learning federado; Nikolaenko et al. [21] apresentam regressão ridge preservando privacidade.
- *Geração de Dados Sintéticos*: Goyal & Mahmoud [10] oferecem uma revisão sistemática de técnicas usando IA generativa; Nadaş et al. [20] exploram LLMs para geração de dados sintéticos; Lu et al. [17] revisam aprendizado de máquina para geração de dados sintéticos.

Essa metodologia garantiu a seleção de trabalhos de qualidade e relevância, cobrindo tanto aspectos teóricos quanto práticos das técnicas de preservação de privacidade em IA.

2 Caracterização Ferramental

2.1 Machine unlearning

Contexto. Desaprendizado de Máquina (ou Machine Unlearning), é um ramo da Aprendizagem de Máquina com foco em realizar um processo de remoção de dados específicos de um modelo após este já ter sido treinado, sem precisar retreiná-lo do zero [3]. Sua proposta surgiu a partir da necessidade de fazer um modelo de aprendizado de máquina ser capaz de remover determinados tipos de dados usados em seu treinamento, como aqueles com desinformação, conteúdo sensível, informação datada e entre outros, quando o artigo 17 do Regulamento Geral sobre a Proteção de Dados da União Europeia entrou em vigor em 2014 [3]. Embora a ideia pareça simples, apagar dados usados para treinamento do banco de dados não implica na remoção de informações relacionadas a esses dados do modelo. Uma solução seria re-treinar a máquina sem os dados removidos, mas isso é computacionalmente custoso. Portanto, para resolver este problema, surgiram estratégias variadas de machine unlearning mais eficientes para o problema:

1. **SISA (Sharded, Isolated, Sliced, Aggregated)**: Consiste em dividir os dados em vários pedaços e treinar sub-modelos. Ao precisar remover algum

dado específico, apenas o sub-modelo que o contém precisará ser treinado novamente, economizando tempo. Conceitualmente, consiste em dividir um problema grande em pequenos problemas, para evitar prejudicar o todo.

2. **Manipulação de Gradiente:** Por meio de uma estratégia mais matemática, a ideia da manipulação de gradiente consiste em analisar a influência dos dados, através do cálculo da mudança de pesos, considerando a inexistência daquele dado no espaço amostral inicial, e dessa forma, aplicar uma correção inversa.
3. **Otimização Professor-Aluno:** Essa estratégia tem como base criar cópias (Alunos) de um modelo principal (Professor). Essas cópias tentam imitar o modelo original sem usar dados do modelo original que precisam ser removidos. Assim, as cópias ignoram esses dados em sua construção.

Tipos.

1. **SISA:** Acrônimo para Shared, Isolated, Sliced, Aggregated, a estratégia SISA permite a remoção de dados específicos sem a necessidade de retreinar o modelo inteiro do zero, tornando, dessa forma, a desaprendizagem de máquina mais eficiente e menos custosa computacionalmente [3]. A seguir, pode-se ver uma explicação breve de cada componente dessa estratégia:
 - a. *Sharding:* A fragmentação divide o conjunto de dados em subconjuntos disjuntos, chamados de Shards. Cada shard é utilizado para treinar uma parte do modelo de maneira separada. Dessa forma, a mudança de dados modifica apenas aquele subconjunto;
 - b. *Isolation:* Consiste em treinar cada subconjunto maneira independente, sem que haja qualquer troca de informação entre diferentes shards. Assim, o isolamento evita a contaminação cruzada, garantindo com que a mudança de dado só afetará ao shard que o contém.
 - c. *Slicing:* Dentro de cada shard, os dados são organizados em fatias sequenciais e o estado do modelo é salvo após seu treinamento para, ao remover determinada informação, o sistema poder retomar do último ponto salvo.
 - d. *Aggregation:* Por fim, os resultados dos modelos dos subconjuntos são combinados para gerar o modelo final único e unificado. A agregação pode ser feita escolhendo algum critério, sendo os principais votação majoritária e média de predição.
2. **Manipulação de Gradiente:** Diferente da estratégia SISA, que procura dividir o conjunto em subconjuntos, a proposta da manipulação de gradiente se baseia na remoção da influência de uma amostra de dados do modelo já treinado, de forma matemática, alterando os parâmetros de forma direta. Durante o treinamento dos modelos, cada dado entrega um gradiente o qual desloca os pesos do modelo em determinada direção. Após isto, há uma estimativa de contribuição passada e se aplica o reajuste inverso, em que os parâmetros são movidos para uma área do espaço dos modelos onde o dado removido nunca existiu [14, 9, 11]. Essa estratégia não garante um modelo resultante exato, diferente do que seria obtido realizando o re-treinamento do zero. Em modelos mais complexos, a influência exercida por um dado

é distribuída por diversos parâmetros, fazendo com que a manipulação de gradiente seja impossível de ser inteiramente eficiente. [9, 11]

3. **Otimização Professor-Aluno:** A estratégia da otimização Professor-Aluno é baseada em destilação de conhecimento, ou seja, transferência de conhecimento de um modelo grande para outro menor. Nessa estratégia, a ideia é que um novo modelo, rotulado como Aluno, imite o modelo grande, rotulado como Professor, sem usar os dados que desejam ser removidos [13, 3]. O processo funciona com o modelo professor encapsulando o conhecimento que foi obtido na sua totalidade por todos os dados informados, enquanto o modelo aluno obtém o conhecimento pelas saídas do professor, limitando-se aos dados permitidos. Por meio disso, o modelo Aluno se comporta exatamente como o modelo Professor se determinados dados removidos nunca tivessem sido adicionados [13]. Assim como os dois métodos anteriores, o sistema Professor-Aluno não garante os resultados exatos. Se, por exemplo, o modelo Professor conter informações sensíveis, esse conhecimento pode ser passado para o modelo Aluno através de previsões.

Conclusões. *Machine Unlearning* surge como uma resposta à necessidade de regulamentação da proteção de dados, especialmente na remoção de dados, imposta por legislações modernas. O Desaprendizado de Máquina oferece soluções que garantem determinadas vantagens, desvantagens e desafios.

1. Vantagens

- a. *Eficiência Computacional:* Com técnicas eficientes de *Machine Unlearning*, o custo computacional e o tempo total de atualização do modelos são reduzidos de maneira significativa em comparação com o simples re-treinamento do modelo do zero [3, 9].
- b. *Governança sobre os Dados:* Como *Machine Unlearning* permite a remoção de dados de maneira ágil e sem gerar custos, isso garante políticas de gestão de dados, permitindo a alteração do modelo de maneira controlada e transparente.

2. Desvantagens

- a. *Ausência de garantia:* Embora as abordagens do desaprendizado de máquina sejam práticas, elas não garantem que o resultado final do modelo seja exatamente igual ao que fosse caso tenha sido treinado do zero [4]. Esse fato foi frequentemente postulado durante as explicações, nesta survey, das principais técnicas de *Machine Unlearning*.
- b. *Complexidade de implementação:* As técnicas de desaprendizagem exigem diversas modificações que são significativas no treinamento do modelo, sendo difícil de implementar nos modelos já existentes.
- c. *Custo de Armazento:* As estratégias de *Machine Unlearning* exigem armazenamento de múltiplos modelos e, possivelmente, várias cópias de dados diferentes, diferentemente de técnicas tradicionais, o que torna implementações de *Machine Unlearning* dispendiosas em relação à memória.

3. Desafios

- a. *Equilíbrio entre privacidade e eficiência:* Os modelos possibilitam a remoção de dados, mas podem prejudicar a confiabilidade do modelo. Um

- dos principais desafios consiste nessa conciliação de privacidade de dados e a eficiência do modelo após alterações [3, 4];
- b. *Automação de Processo:* Por ainda ser recente, ainda se tem como desafio a criação de sistemas capazes de lidar com os pedidos de *unlearning* de forma automatizada;
 - c. *Aplicação em cenários reais:* Ainda existem desafios na aplicação dos métodos em cenários reais, como em ambientes de aprendizado distribuído, sistemas de escala industrial, sistemas governamentais e entre outros.

2.2 Differential privacy

Contexto. A Privacidade Diferencial (DP) foi formalmente proposta em 2006 por Cynthia Dwork e seus colaboradores como uma forma de assegurar a privacidade dos dados, fornecendo uma robusta garantia matemática, em contraposição à vulnerabilidade dos métodos tradicionais de anonimização, que se mostraram ineficazes contra ataques de ligação e reidentificação em grandes conjuntos de dados, além de minimizar o riscos como a memorização indesejada de dados sensíveis por redes neurais [23, 6, 24, 4]. A DP é essencial no cenário atual de Big Data porque fornece uma garantia de privacidade que é independente do poder computacional ou do conhecimento prévio de um atacante [6]. Ela resolve o problema da memorização em redes neurais, onde modelos complexos acabam armazenando dados sensíveis de treinamento de forma não intencional, tornando-os vulneráveis a ataques de inferência de membro (membership inference) [4].

Definição. Differential Privacy é um robusto método matemático e estatístico para proteger a privacidade dos registros presentes em um conjunto de dados, permitindo análises desses dados sem revelar informações específicas sobre um dado registro, garantindo que os resultados de algoritmos aplicados sobre esses dados não possuam resultados substancialmente diferentes caso um dos registros seja adicionado, removido ou alterado, assegurando um resultado próximo ao valor real, respeitando uma margem de erro aceitável e controlada. Isso é realizado por meio da adição de um ruído calculado que ofusca a saída dos algoritmos, garantindo que qualquer registro do conjunto de dados tenha um impacto limitado no resultado, protegendo, dessa forma, a privacidade individual dos dados, porque os valores das saídas estarão dentro de um intervalo que contém o valor do resultado real e cujos limitantes, tanto superior quanto inferior, respeitam um valor de erro tolerável [23, 15].

Breve análise técnica. Esta seção abordará uma visão geral de definições teóricas, propriedades e conceitos importantes para fundamentar o entendimento sobre a privacidade diferencial.

1. **Neighboring Datasets (Conjuntos de Dados Vizinhos):** Considere dois conjuntos de dados denotados por D e D' , eles serão vizinhos se a diferença entre eles for apenas um único registro. Isto é, para transformar um conjunto de dados original em um conjunto de dados vizinho, basta realizar uma única alteração, que pode ser adicionar ou remover única tupla. [23] No

contexto de DP, dois datasets vizinhos, um contendo um registro específico e outro não contendo este mesmo registro, ao passarem por um algoritmo de análise de dados, iriam produzir resultados muito próximos, tornando muito difícil distinguir a exata contribuição do registro específico sobre o resultado ao comparar ambos os conjuntos de dados vizinhos, assegurando a privacidade individual deste registro específico.

2. ϵ (**Epsilon**): É um parâmetro conhecido como privacy budget (orçamento de privacidade), cujo valor deve ser maior ou igual a zero. É representativo do grau de proteção da privacidade. O valor do privacy budget é inversamente proporcional à garantia da privacidade, isto é, quanto menor o valor desta variável, mais rigorosa é a proteção dos registros. Isso implica que a quantidade de ruído adicionada é maior, porque ruído é o que assegura a privacidade ao dificultar que atacantes adquiram informações se determinado registro está presente ou não em conjunto de dados por meio de ataques de inferência.
3. δ (**Delta**): É um parâmetro que representa a probabilidade da garantia da privacidade falhar. Quanto menor o valor desta, maior é a segurança dos dados, valores ideais de delta são muito próximos a zero. Para que a proteção seja considerada robusta, o valor desse parâmetro deve ser significativamente menor que o inverso do tamanho do conjunto de dados, isto é, $\delta < \frac{1}{|D|}$; caso contrário, valores elevados desse parâmetro poderiam permitir a exposição direta de informações sensíveis, tornando a garantia de privacidade trivial. A partir disso, temos a definição matemática da privacidade diferencial $((\epsilon, \delta)-DP)$, descrita pela expressão:

$$\Pr[M(D) \in S] \leq \Pr[M(D') \in S] \times (e^\epsilon) + \delta \quad (1)$$

Onde D e D' são conjuntos de dados vizinhos, M é um algoritmo qualquer e S é um conjunto de saídas possíveis desse algoritmo. Essa definição formal estabelece que um algoritmo M é (ϵ, δ) -diferencialmente privado se, para quaisquer dois conjuntos de dados vizinhos, a probabilidade de o algoritmo gerar um determinado resultado for quase a mesma em ambos os casos. Os parâmetros ϵ e δ controlam, respectivamente, o nível de similaridade entre as saídas e o tamanho do erro aceitável para situações em que a garantia estrita de privacidade possa falhar.

4. **Mecanismos:** Algoritmos que calculam o ruído a ser adicionado para garantir a privacidade diferencial de um conjunto de dados. Existem dois mecanismos comumente utilizados:
 - a. *Mecanismo de Laplace*: técnica clássica para preservar a privacidade em dados numéricos, adicionando ruído aleatório baseado em uma distribuição de Laplace ao resultado real da consulta [23, 24].
 - b. *Mecanismo Gaussiano*: técnica amplamente utilizada em contextos de aprendizado profundo (Deep Learning) e grandes volumes de dados, adicionando ruído baseado em uma distribuição normal (Gaussiana) [23, 7].
5. **Sensibilidade:** É o parâmetro que quantifica o impacto máximo da inclusão ou remoção de um único registro sobre a saída de um algoritmo. Esse

parâmetro auxilia a definir a quantidade de ruído necessário para garantir a privacidade. O Mecanismo de Laplace utiliza uma sensibilidade baseada na Norma L1 (distância de Manhattan), que corresponde à soma dos valores absolutos das diferenças entre os resultados dos algoritmos (representados por $f(D)$ e $f(D')$):

$$\Delta f_1 = \max_{D,D'} \|f(D) - f(D')\|_1 = \max_{D,D'} \sum_{i=1}^k |f(D)_i - f(D')_i| \quad (2)$$

Em contrapartida, o Mecanismo Gaussiano utiliza uma sensibilidade baseada na Norma L2 (distância Euclidiana), que corresponde à raiz quadrada da soma dos quadrados das diferenças entre os resultados dos algoritmos (representados por $f(D)$ e $f(D')$).

$$\Delta f_2 = \max_{D,D'} \|f(D) - f(D')\|_2 = \max_{D,D'} \sqrt{\sum_{i=1}^k (f(D)_i - f(D')_i)^2} \quad (3)$$

Tipos. Existem atualmente dois tipos amplamente utilizados de differential privacy, a DP centralizada e a DP local, que diferem entre si pelo momento onde o ruído é adicionado. [23]

1. **Centralized Differential Privacy:** o ruído é adicionado após os dados chegarem no Data center.
2. **Local Differential Privacy:** o ruído é adicionado ainda no dispositivo dos usuários, os dados enviados pelo dispositivo até o Data center já estão privados.

Cenário atual. Essa seção definirá mais claramente as vantagens e desvantagens do método, além de explicitar os desafios atuais do estudo na área.

1. Vantagens:

- a. *Composição:* Permite calcular o risco acumulado de privacidade após múltiplas análises sobre o mesmo conjunto de dados (o chamado privacy accounting) [23].
- b. *Imunidade ao pós-processamento:* Uma vez que um dado é tornado privado através de um mecanismo de DP, nenhuma computação adicional pode reverter essa proteção ou aumentar o vazamento de informações [6].
- c. *Robustez e generalização:* Ao adicionar ruído durante o treinamento, a DP pode atuar como uma forma de regularização, ajudando o modelo a focar em características genéricas da população em vez de ruídos específicos de registros individuais [23].

2. Desvantagens:

- a. *Dificuldade em equilibrar utilidade e privacidade:* Níveis mais altos de privacidade (menor epsilon) resultam em uma maior degradação da acurácia do modelo devido ao alto ruído adicionado para assegurar a privacidade. [23, 15].

- b. *Sobrecarga Computacional:* Algoritmos como o DP-SGD exigem o corte (clipping) de gradientes e o processamento de ruído para cada amostra ou lote, o que aumenta o tempo de treinamento e a demanda por recursos de hardware [23].
- 3. Desafios:**
- a. *Fairness:* Estudos indicam que a DP pode afetar de forma desproporcional a precisão em subgrupos minoritários ou sub-representados, levantando dilemas éticos sobre justiça algorítmica [23].
 - b. *Gerenciamento do orçamento:* Em sistemas que recebem atualizações contínuas de dados, como o Aprendizado Federado, o orçamento de privacidade (ϵ) pode se esgotar rapidamente, exigindo mecanismos complexos para renovar ou otimizar o consumo desse recurso [23, 26].
 - c. *Implementação em Machine Unlearning:* Garantir que um modelo possa perder completamente um dado de forma diferencialmente privada sem precisar ser treinado novamente do zero continua sendo uma fronteira de pesquisa ativa e complexa [22].

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 vêm 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

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 [26].

de exposição de dados sensíveis.

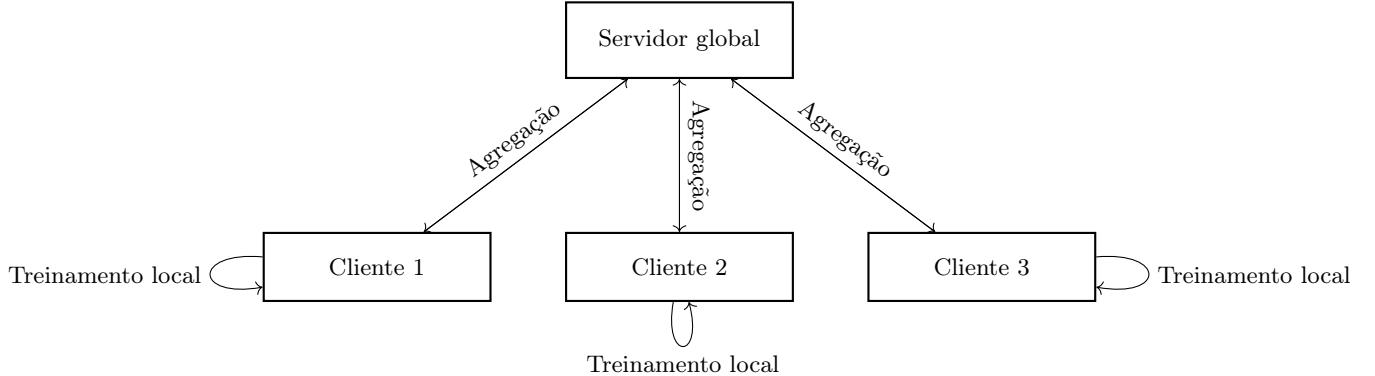


Fig. 1. Fluxo básico do aprendizado federado.

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. [26], 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 [18] o algoritmo de agregação mais comum [26].
 - 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 [12, 5].
 - 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 [25].
2. **Com relação aos mecanismos de privacidade.** Embora o federated learning evite o compartilhamento direto dos dados brutos, a troca de parâmetros

ros 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. Protótipos 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 [12, 26].
 - 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 [19, 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 [21, 26].
 - 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 [5].
 - 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 [18, 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-IID² 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:

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 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 comuns em aprendizado federado.

- 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 [26].
- 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 [18].
- 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 Reinforcement Learning (FRL), como o de Fan et al. [8], 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. [16] propõe a separação entre representações globais e locais, reduzindo o impacto da heterogeneidade estatística e melhorando o desempenho individual dos clientes.

3 Geração de Dados Sintéticos

A escassez de dados de alta qualidade, combinada com regulamentações rigorosas de privacidade e o alto custo de anotação manual, impulsionou a adoção da geração de dados sintéticos como uma alternativa viável aos *datasets* reais [17]. Dados sintéticos são definidos como informações artificialmente geradas por algoritmos ou simulações que mimetizam as propriedades estatísticas e comportamentais dos dados reais, sem conter informações diretamente identificáveis. No contexto de sistemas de Inteligência Artificial seguros e privados, a geração de dados sintéticos não atua apenas como uma ferramenta de aumento de dados (*data augmentation*), mas como um mecanismo fundamental de preservação de privacidade e robustez[10].

3.1 Técnicas de Geração Baseadas em IA Generativa

A literatura recente categoriza as abordagens de geração de dados sintéticos principalmente em três arquiteturas de aprendizado profundo:

1. **Redes Adversárias Generativas (GANs):** Consistem em dois modelos neurais, um gerador e um discriminador, que competem entre si. O gerador cria dados falsos e o discriminador tenta distinguir entre dados reais e falsos. GANs são amplamente utilizadas para síntese de imagens e dados tabulares, com variantes aplicadas especificamente para preservar a estrutura estatística de dados sensíveis [10].
2. **Autoencoders Variacionais (VAEs):** Os VAEs aprendem a comprimir os dados de entrada em um espaço latente probabilístico e, em seguida, reconstróem os dados a partir desse espaço, permitindo a amostragem de novos pontos de dados que seguem a distribuição original [17].
3. **Grandes Modelos de Linguagem (LLMs):** Avanços recentes, como apontado por Nadäš et al. [20], demonstram que LLMs podem atuar como geradores de dados universais para texto e código. Através de técnicas de *prompting* e refinamento iterativo, LLMs podem gerar dados rotulados e código sintético verificável via execução, auxiliando no treinamento de modelos de segurança de software.

3.2 Sinergia com Mecanismos de Privacidade e Segurança

A geração de dados sintéticos desempenha um papel estratégico quando integrada às técnicas de *Differential Privacy*, *Federated Learning* e *Machine Unlearning*.

Privacidade Diferencial (DP). A aplicação direta de modelos gerativos em dados sensíveis pode resultar em riscos de privacidade, como ataques de inferência de pertinência (*membership inference attacks*). Para mitigar isso, a geração de dados sintéticos é frequentemente acoplada à Privacidade Diferencial [17]. O objetivo é gerar um *dataset* sintético que preserve as propriedades estatísticas globais, mas garanta que a saída do modelo não dependa excessivamente de nenhum registro individual. Técnicas como DP-GAN integram ruído calibrado garantindo garantias formais de privacidade (ϵ -differential privacy), permitindo que dados sintéticos sejam compartilhados publicamente com risco minimizado.

Aprendizado Federado (FL). No Aprendizado Federado, a heterogeneidade dos dados é um desafio crítico. A geração de dados sintéticos oferece soluções em duas frentes [10]:

1. **Aumento de Dados Local:** Clientes com poucos dados podem usar modelos generativos para sintetizar amostras locais, melhorando a robustez do treinamento.
2. **Privacidade Generativa Federada:** O treinamento de GANs em ambiente federado, onde apenas os parâmetros dos geradores são compartilhados, permite que o modelo global aprenda a distribuição de dados sem acesso direto aos registros brutos.

Machine Unlearning. O *Machine Unlearning* visa remover a influência de dados específicos de um modelo treinado. A geração de dados sintéticos atua como facilitador neste processo:

1. **Prevenção de Memorização:** O uso de dados sintéticos treinados com DP desde o início reduz a necessidade de *unlearning* frequente, pois os dados não correspondem a indivíduos reais.
2. **Substituição de Dados:** Em cenários onde dados reais devem ser excluídos, dados sintéticos podem ser gerados para preencher a lacuna estatística deixada pela remoção, mantendo a utilidade do modelo sem violar a privacidade [20].

4 Conclusão

A crescente dependência de sistemas de Inteligência Artificial em diversos setores da sociedade trouxe à tona questões fundamentais sobre a proteção de dados pessoais e a conformidade com regulamentações de privacidade. Neste trabalho, apresentamos uma revisão sistemática das principais técnicas de preservação de privacidade aplicadas ao contexto de aprendizado de máquina, explorando suas características, aplicações e limitações.

Conforme discutido ao longo deste *survey*, o *federated learning* emerge como uma solução robusta para cenários onde a centralização de dados é inviável ou indesejável. Ao manter os dados nos dispositivos locais e compartilhar apenas parâmetros de modelo, essa abordagem reduz significativamente os riscos de vazamento de informações sensíveis. No entanto, como demonstrado, a simples descentralização não é suficiente. A heterogeneidade de dados não-IID, as limitações de comunicação e os potenciais ataques de inferência sobre gradientes compartilhados exigem mecanismos adicionais de proteção.

Nesse contexto, a privacidade diferencial atua como uma camada complementar de segurança, fornecendo garantias formais e quantificáveis sobre o nível de privacidade preservado. A adição controlada de ruído estatístico aos dados ou aos parâmetros do modelo limita a capacidade de adversários inferirem informações sobre indivíduos específicos, mesmo quando têm acesso ao modelo treinado. Essa técnica, quando integrada ao aprendizado federado, fortalece substancialmente as garantias de privacidade do sistema como um todo.

O *machine unlearning*, por sua vez, responde a uma demanda regulatória e ética cada vez mais presente: o direito ao esquecimento. A capacidade de remover seletivamente a influência de dados específicos de modelos já treinados, sem comprometer significativamente seu desempenho, representa um avanço importante na direção de sistemas de IA mais transparentes e responsáveis. Embora ainda existam desafios técnicos consideráveis, especialmente em relação à eficiência computacional e às garantias de remoção completa, os métodos discutidos neste trabalho demonstram a viabilidade prática dessa abordagem.

A geração de dados sintéticos, como explorado na Seção 3, desempenha um papel estratégico ao complementar as técnicas anteriores. Ao produzir dados artificiais que preservam propriedades estatísticas relevantes sem expor informações

individuais, essa abordagem não apenas facilita o compartilhamento de dados para pesquisa e desenvolvimento, mas também oferece soluções para problemas de escassez de dados e desbalanceamento de classes. A sinergia entre geração de dados sintéticos e privacidade diferencial, em particular, permite a criação de datasets públicos com garantias formais de privacidade.

Desafios e direções futuras. Apesar dos avanços significativos nas técnicas de preservação de privacidade, diversos desafios permanecem em aberto. A tensão fundamental entre utilidade do modelo e nível de privacidade garantido continua sendo um ponto crítico de pesquisa. Em muitos cenários práticos, a adição de ruído ou a descentralização do treinamento resulta em degradação de desempenho que pode ser inaceitável para aplicações críticas.

Além disso, a composição de múltiplas técnicas de privacidade, embora promissora, introduz complexidades adicionais em termos de análise de garantias e overhead computacional. A integração de federated learning com privacidade diferencial e machine unlearning, por exemplo, requer cuidadosa calibração de parâmetros e análise de trade-offs.

Outro desafio importante diz respeito à escalabilidade dessas técnicas para cenários de larga escala. Embora frameworks como o proposto por Bonawitz et al. [1] demonstrem a viabilidade do aprendizado federado com milhões de dispositivos, questões relacionadas a custos de comunicação, tolerância a falhas e heterogeneidade de recursos permanecem como obstáculos práticos.

Por fim, a avaliação rigorosa de garantias de privacidade em sistemas reais continua sendo um problema em aberto. Ataques cada vez mais sofisticados, como os de inferência de pertinência e reconstrução de dados, demonstram que garantias teóricas nem sempre se traduzem em proteção efetiva contra adversários determinados.

Considerações finais. As técnicas de preservação de privacidade discutidas neste *survey* representam avanços fundamentais na direção de sistemas de IA mais seguros e responsáveis. A combinação de *federated learning*, *differential privacy*, *machine unlearning* e geração de dados sintéticos oferece um conjunto robusto de ferramentas para diferentes cenários e requisitos de privacidade. No entanto, a adoção prática dessas técnicas ainda enfrenta barreiras técnicas, regulatórias e organizacionais que demandam esforços contínuos de pesquisa e desenvolvimento. À medida que regulamentações de privacidade se tornam mais rigorosas e a conscientização sobre proteção de dados aumenta, espera-se que essas técnicas se tornem componentes essenciais de qualquer sistema de IA implantado em ambientes de produção.

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., Roslander, 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

- tion 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. Bourtoule, L., Chandrasekaran, V., Choquette-Choo, C., Jia, H., Travers, A., Zhang, B., Lie, D., Papernot, N.: Machine unlearning. In: Proceedings of the IEEE Symposium on Security and Privacy. pp. 141–159 (2021)
 4. Chen, M., Zhang, Z., Wang, T., Backes, M., Humbert, M., Zhang, Y.: When machine unlearning jeopardizes privacy. In: Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. pp. 896–911. ACM (Nov 2021). <https://doi.org/10.1145/3460120.3484756>
 5. 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>
 6. Dwork, C., Roth, A.: The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.* **9**(3-4), 211–407 (2014). <https://doi.org/10.1561/0400000042>, <https://doi.org/10.1561/0400000042>
 7. Dwork, C., Rothblum, G.N.: Concentrated differential privacy (2016), <https://arxiv.org/abs/1603.01887>
 8. 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>
 9. Golatkar, A., Achille, A., Soatto, S.: Eternal sunshine of the spotless net: Selective forgetting in deep networks. In: Advances in Neural Information Processing Systems (2020)
 10. Goyal, M., Mahmoud, Q.H.: A systematic review of synthetic data generation techniques using generative ai (2024). <https://doi.org/10.3390/electronics13173509>, <https://www.mdpi.com/2079-9292/13/17/3509>
 11. Guo, C., Goldstein, T., Hannun, A., Van Der Maaten, L.: Certified data removal from machine learning models. In: IEEE Symposium on Security and Privacy (2020)
 12. 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>
 13. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
 14. Koh, P.W., Liang, P.: Understanding black-box predictions via influence functions. In: Proceedings of the 34th International Conference on Machine Learning (2017)
 15. Li, L., Fan, Y., Tse, M., Lin, K.Y.: A review of applications in federated learning. *Computers and Industrial Engineering* **149**, 106854 (2020). <https://doi.org/https://doi.org/10.1016/j.cie.2020.106854>, <https://www.sciencedirect.com/science/article/pii/S0360835220305532>
 16. 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>
 17. Lu, Y., Chen, L., Zhang, Y., Shen, M., Wang, H., Wang, X., van Rechem, C., Fu, T., Wei, W.: Machine learning for synthetic data generation: A review (2025), <https://arxiv.org/abs/2302.04062>

18. 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>
19. McMahan, H.B., Ramage, D., Talwar, K., Zhang, L.: Learning differentially private recurrent language models (2018), <https://arxiv.org/abs/1710.06963>
20. Nadăș, M., Dioșan, L., Tomescu, A.: Synthetic data generation using large language models: Advances in text and code (2025). <https://doi.org/10.1109/ACCESS.2025.3589503>, <http://dx.doi.org/10.1109/ACCESS.2025.3589503>
21. 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>
22. Wang, F., Li, B., Li, B.: Federated unlearning and its privacy threats. IEEE Network **38**(2), 294–300 (2024). <https://doi.org/10.1109/MNET.004.2300056>
23. Wang, Y., Wang, Q., Zhao, L., Wang, C.: Differential privacy in deep learning: Privacy and beyond. Future Generation Computer Systems **148**, 408–424 (2023). <https://doi.org/https://doi.org/10.1016/j.future.2023.06.010>, <https://www.sciencedirect.com/science/article/pii/S0167739X23002315>
24. Yang, M., Lyu, L., Zhao, J., Zhu, T., Lam, K.Y.: Local differential privacy and its applications: A comprehensive survey (2020), <https://arxiv.org/abs/2008.03686>
25. 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
26. 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>