# Verteiltes maschinelles Lernen

## Besserer Datenschutz für KI-Anwendungen?

**IN KÜRZE** Verteiltes maschinelles Lernen

- verspricht besseren Datenschutz qua Design bei zugleich hoher Performanz.
- trainiert Modelle maschinellen Lernens (ML) dezentral auf Endgeräten statt zentral auf einem Server, greift damit Edge Computing für KI auf und verteilt so die Rechenlast.
- ermöglicht, dass – häufig personenbezogene – Trainingsdaten auf Endgeräten und somit bei den Nutzenden verbleiben.
- kann über diese Datenhoheit den Schutz persönlicher Daten und die Gewährleistung der informationellen Selbstbestimmung erhöhen.
- ist in der Variante des Federated Learning bereits in Anwendung; andere Ansätze befinden sich noch im Forschungsstadium bzw. an der Schwelle zum Markteintritt.
- lässt sich vielfältig einsetzen, etwa für Mobilitäts- oder Gesundheitsanwendungen.

**Allerdings:** Verteiltes maschinelles Lernen schafft neue Einfallstore für Angreifer und erzeugt möglicherweise ein trügerisches Sicherheitsgefühl. Einige Expertinnen und Experten warnen daher vor überzogenen Erwartungen in puncto Datenschutz.

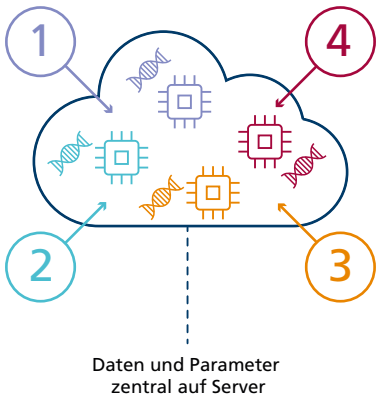
## Ausgangslage

KI-Systeme basieren auf dem Training mit großen Mengen an – teils sensiblen – Daten. Deren Nutzung steht bisweilen in einem Spannungsverhältnis zum Datenschutz und dem persönlichen Recht, selbst über die Preisgabe und Verwendung personenbezogener Daten zu bestimmen (informationelle Selbstbestimmung). Dies ist etwa der Fall, wenn ein KI-System Nutzenden, basierend auf ihrer Suchhistorie, nur bestimmte Vorschläge macht und andere, die möglicherweise passender sind, ausblendet. Gleichzeitig bestehen beim Training von KI-Systemen rechtliche Unsicherheiten für Unternehmen: Personenbezogene Daten dürfen gemäß Datenschutz-Grundverordnung (DSGVO) grundsätzlich nur zweckgebunden genutzt werden; für andere Zwecke kann es nötig sein, das Einverständnis dieser Personen nachträglich einzuholen oder die individuellen Interessen abzuwägen. Letzteres ist aufwändig und interpretationsoffen.

Es gibt jedoch technische Lösungsansätze, die Datennutzung und -schutz wirksam verbinden – und möglicherweise neue Marktchancen für datenschutzwahrende KI-Lösungen schaffen. Dazu zählt die Methode des verteilten maschinellen Lernens.

# Klassischer Ansatz: Zentralisiertes maschinelles Lernen

Aktuell geläufig in der KI-Entwicklung ist die Methode des zentralisierten maschinellen Lernens (ML). Dabei wird ein statistisches Modell über einen Lernalgorithmus zentral auf einem Server (bei der Anwenderin/beim Anwender oder in der Cloud) trainiert. Dazu sammelt der Server für das Training Daten von Endgeräten wie Smartphones oder Sensoren (sog. Clients) ein und bündelt sie zentral. Das trainierte Modell kann dann wieder an die Endgeräte verteilt bzw. auf ihnen angewandt werden. Diese Form des maschinellen Lernens wird beispielsweise in der industriellen Produktion zur vorausschauenden Überwachung und Wartung von Anlagen (Predictive Maintenance) eingesetzt.




①②③④ Zugriff der Endgeräte auf ML-Modell

Quelle: Eigene Darstellung nach Warnat-Herresthal et al. (2021)

| Vorteile | | Nachteile |
|---|---|---|
| **Ex-ante**: Einhaltung der Datenschutzanforderungen lässt sich durch zentrale Vorgaben gewährleisten | Datenschutz | **Angriffspunkte**: Sensible Daten können über Attacken aus trainiertem Modell entschlüsselt bzw. abgegriffen werden<br>Zentrale Datensammlung ermöglicht direkten Zugriff auf mitunter sensible (Roh-)Daten |
| **Konzentration**: Nur serverseitige Absicherung in Bezug auf Training des ML-Modells nötig | Sicherheit | **Single Point of Attack**: Mögliche Angriffe auf Server bedrohen Sicherheit des Systems |
| **Single Point of Truth**: Zentralisierte Architektur verständlich und einfach zu warten<br>**Skalierung**: Kompatible Geräte lassen sich ohne großen Aufwand hinzufügen | Technologie | **Schnittstellen**: Einbindung von inkompatiblen Endgeräten oder Datenformaten nicht immer möglich |
| **Geschwindigkeit**: Kaum Verzögerung (Latenzzeit) zwischen Datensammlung und Trainingsbeginn des ML-Modells bei einheitlicher Datenquelle<br>**Hohe Datenverfügbarkeit**: Zentrale Instanz für Datenverarbeitung stärkt Effizienz und Genauigkeit | Leistungsfähigkeit | **Limitierte Möglichkeiten des Echtzeit-Lernens**: Hochladen kompletter Datensätze von Endgeräten sowie Verteilung des ML-Modells vom Server an Endgeräte aufwändig |
| **Eindeutigkeit**: Klare Zurechenbarkeit der Verantwortung für Training eines ML-Modells, das immer auf von einem Anbieter betriebenen zentralen Server geschieht | Ethik | |

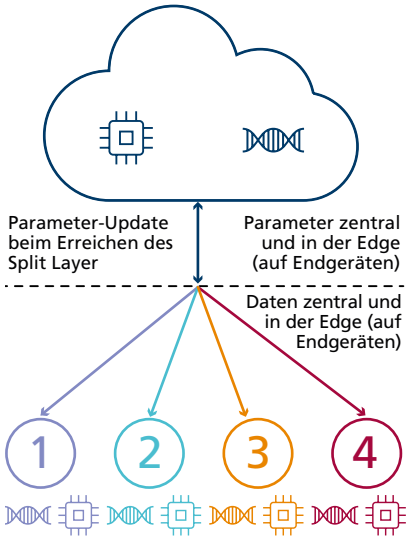
# Neuer Ansatz: Verteiltes maschinelles Lernen

Beim verteilten maschinellen Lernen wird das ML-Modell nicht auf einem zentralen Server trainiert. Stattdessen greift jedes Endgerät (sog. Client) auf das aktuelle ML-Modell zu und trainiert dieses lokal mit dem eigenen Datensatz. Um das ML-Modell zu aktualisieren und zu verbessern, werden nur die Trainingsergebnisse (sog. Weights), nicht aber die Daten mit anderen Endgeräten getauscht (Edge Computing/Edge KI). Drei populäre technische Ansätze des verteilten maschinellen Lernens werden im Folgenden vorgestellt.

## Technische Ansätze

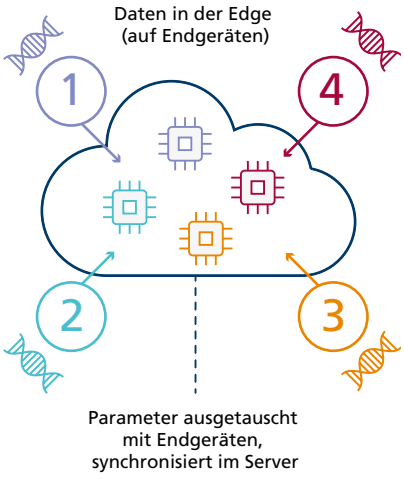
### Split Learning – Lernen sowohl auf Endgeräten als auch auf Server

- ML-Modell wird in verschiedene Teilmodelle aufgespalten (sog. Links) und sowohl auf Endgeräten (Clients) als auch auf dem Server trainiert, ohne dass Rohdaten geteilt werden (effiziente Verteilung der Rechenlast)
- Iterativer Trainingsprozess: Endgeräte und Server tauschen am Teilungspunkt des ML-Modells (sog. Split Layer) statt Rohdaten nur Ergebnisse des trainierten ML-Modellabschnitts (Weights) aus und trainieren mit diesen auf eigenem Datensatz weiter (geringere Kommunikationskosten)
- Iterationen enden bei erreichter Konvergenz zwischen ML-Modellen der Endgeräte und dem Server



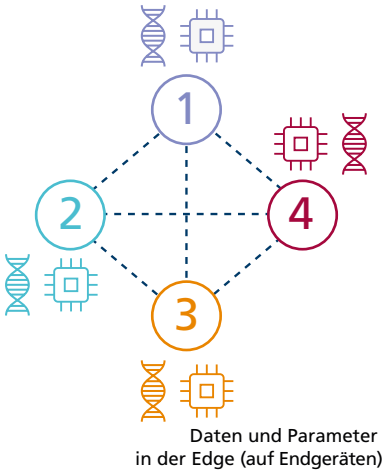

### Federated Learning – Lernen mit zentralem Server als Aggregationsinstanz

- Endgeräte laden Parameter des ML-Modells vom Server herunter
- ML-Modell wird durch Endgeräte mit lokalem Datensatz trainiert
- Endgeräte senden nur Weights an den Server; lokaler Datensatz bleibt beim Endgerät
- Auf dem Server findet kein Training statt, sondern nur die Zusammensetzung der Weights zur zentralen Aktualisierung des ML-Modells (Inferenz)
- Server stellt Parameter des verbesserten, weil synchronisierten ML-Modells an Endgeräte für neuerliches Training bereit
- Beliebig wiederholbarer Prozess, bei dem sich das verteilte ML-Modell stets weiter optimiert




### Swarm Learning – Lernen auf verteilten Geräten ohne Aggregationsinstanz

- Parameter des ML-Modells liegen in zugangsbeschränkter Blockchain statt auf zentralem Server
- Zwar keine Koordinierungsinstanz, aber zentrale Instanz zur Vorautorisierung der Endgeräte für Zugriff auf Blockchain nötig
- Endgeräte laden Parameter des ML-Modells aus Blockchain und können es mit lokalem Datensatz trainieren
- Nach Training werden nur die angepassten Weights in der Blockchain gespeichert
- Angepasste Weights und Parameter des ML-Modells können von Endgeräten ausgelesen und lokal zum Gesamtmodell zusammengesetzt werden




Quelle: Eigene Darstellung

# Anwendungsbeispiele

Verteiltes maschinelles Lernen kommt sukzessive in die wirtschaftliche Anwendung – insbesondere im Mobilitäts- und Gesundheitssektor, aber auch in anderen lebensnahen Bereichen. Im Folgenden wird für jeden der drei vorgestellten technischen Ansätze ein Anwendungsbeispiel skizziert. In der Praxis wären diese Anwendungen auch durch andere Ansätze des verteilten maschinellen Lernens umsetzbar.

**SPLIT LEARNING**

**Mobilität**

## Kontinuierliche Verbesserung von Bilderkennungs-Algorithmen beim autonomen Fahren

**Ausgangslage**

- Automatisierte Erkennung von Hindernissen und Verkehrsschildern zentral für autonomes Fahren
- Sensorbasierte Bilderkennung und -verarbeitung erforderlich

**Relevanz Datenteilung**

- Gesamtes Streckennetz nicht durch ein Auto für Bilderkennung abfahrbar
- Zentrales Modell würde Besonderheiten bestimmter Strecken nicht berücksichtigen

**Relevanz Datenschutz**

- Routenverlauf erlaubt Rückschlüsse auf Mobilitätsverhalten und sensitive personenbezogene Daten

Autonome Fahrzeuge lernen verteilt die Erkennung von Verkehrszeichen. Ein Server übernimmt die Synchronisierung (Integritätswahrung)

1 Auf zentralem Server wird ein Grundmodell aufgesetzt, das anhand frei verfügbarer Bilder aus dem Straßenverkehr trainiert wird.

2 Hat das trainierte ML-Modell eine ausreichende Qualität erreicht, stellt der Server Parameter für den Download durch Endgeräte (Sensoren von autonomen Autos) zur Verfügung.

3 Sensoren scannen bei Fahrten die Umgebung (Bilderkennung) und speisen damit einen lokalen Datensatz (Routenverlauf, Bilder der Verkehrszeichen etc.), der für lokales Training des ML-Modells mit vom Server geladenen Parametern verwendet wird.

4 Sensoren stellen dem Server bei Erreichen des Teilungspunkts (Split Layer) nur die lokal trainierten Parameter zur Verfügung. Der Server trainiert das Modell mit diesen zentral weiter, bis er wiederum einen Teilungspunkt erreicht.

**Kontinuierliche Iteration**

FEDERATED LEARNING

## Autovervollständigung und -korrektur

Kontinuierliche Verbesserung der Wortvorschläge von Smartphones

**Ausgangslage**

- Autovervollständigung wichtig für User Experience
- Adaptivität an individuelle (Sprach-)Gewohnheiten erhöht Qualität und Genauigkeit

**Relevanz Datenteilung**

- Qualitätspotenziale durch Skalierungseffekte über Milliarden von Smartphones
- Upload aller Textdaten auf zentralen Server würde Kapazitätsgrenzen sprengen

**Relevanz Datenschutz**

- Geschriebene Texte erlauben Rückschlüsse, z. B. auf Lebenssituationen oder Betriebsgeheimnisse

Für die Synchronisierung und Verbesserung der Autovervollständigung wird das ML-Modell auf den Smartphones der Nutzenden (Endgeräte) trainiert; Weights werden auf den Server hochgeladen.

1. Das Smartphone speichert bei Texterstellung Informationen zum Kontext und ob Nutzende auf einen Suchvorschlag klicken (Erstellung Datensatz).
2. Das ML-Modell wird lokal auf dem Smartphone mit dem Datensatz trainiert.
3. Weights werden auf den Server hochgeladen. Dieser synchronisiert die angepassten Parameter aller Smartphones zur Verbesserung des Vorschlagsmodells.
4. Der Server stellt die synchronisierten Parameter zum Download für Smartphones zur abermaligen Verbesserung der Autovervollständigung bereit.

**Kontinuierliche Iteration über die ersten vier Schritte**

SWARM LEARNING

## Gesundheit

Identifikation von Krankheitsfällen (Leukämie, Tuberkulose, Covid-19)

**Ausgangslage**

- Big Data Analytics basierend auf individuellen Gesundheitsdaten ermöglicht Erkennung von Erkrankungen
- Vorhersage über Blut-Transkriptome (Gesamtheit aller RNA-Moleküle einer Zelle)

**Relevanz Datenteilung**

- Statistische Vorhersagen erst bei vielen Datenpunkten möglich (Large N-Problem)
- Individuelle Parameter erlauben erst im Vergleich mit Gesamtheit Rückschlüsse

**Relevanz Datenschutz**

- Persönliche Gesundheitsdaten äußerst sensibel
- Gefahr der Verletzung von Persönlichkeitsrechten (z. B. missbräuchliche Verwendung zur Einstufung bei Versicherungen)

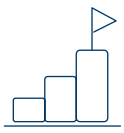
Für die Transkriptom-basierte Vorhersage von Krankheitsfällen wird das ML-Modell dezentral bei Kliniken mit lokal erhobenen Patientendaten trainiert.

1. Das ML-Modell liegt verteilt auf der Blockchain bei verschiedenen Kliniken (sog. Nodes), die über Krankenkassenzulassung Blockchain-Zugang haben.
2. Transkriptome von Patientinnen und Patienten werden in jeder Klinik einzeln erfasst (lokaler Datensatz).
3. Aktuelle Parameter des ML-Modells werden durch einzelne Kliniken von der Blockchain abgerufen und lokal zum Gesamtmodell zusammengesetzt.
4. ML-Modell wird mit jeweiligem lokalen Datensatz trainiert. Weights (Parameter des aktualisierten ML-Modells) werden in der Blockchain gespeichert.

**Kontinuierliche Iteration**

# Potenziale und Herausforderungen: Kompakt analysiert

Die praktische Anwendung von Ansätzen des verteilten maschinellen Lernens scheint vielversprechend. Dennoch bestehen Herausforderungen, die nicht zu vernachlässigen sind und bereits angegangen werden. Im Folgenden werden Potenziale und Herausforderungen des verteilten maschinellen Lernens gegenüber gestellt.

| Potenziale | | Herausforderungen |
|---|---|---|
| **Datenhoheit**: Daten verbleiben für das Training des ML-Modells auf Endgerät der Nutzenden<br>**Kein Datenpooling**: Personenbezogene Daten müssen nicht ausgetauscht werden<br>**Kooperation**: Verschiedene Organisationen können ohne Austausch kritischer Daten gemeinsam ML-Modell nutzen | Datenschutz  | **Privacy**: Modell-Updates erlauben Rückschlüsse auf personenbezogene Daten<br>**De-Anonymisierung**: Weiterhin lässt sich herausfinden, ob Daten bestimmter Personen im Trainingsdatensatz enthalten sind |
| **Verteiltes Risiko**: Datensatzbasierte Attacken sind aufgrund der Verteilung von Daten über Endgeräte schwieriger und reichweitenärmer<br>**Auflösung des Single Point of Attack**: Modell und Datensatz sind voneinander getrennt und erschweren so umfangreichen Angriff | Sicherheit  | **Geringer lokaler Schutz**: Durch Edge Computing weniger Rechenleistung für Angriffserkennung auf Endgeräten verfügbar<br>**Neue Angriffsziele**: Zusätzlich sind Datensätze auf Endgeräten (z. B. vor Data Poisoning) sowie indirekter Zugriff auf ML-Modell über lokale Datensätze (z. B. vor Model Poisoning) zu schützen |
| **Schwarmintelligenz**: Qualität des Gesamtmodells steigt mit Anzahl beteiligter Endgeräte<br>**Toleranz**: Lernen mit diversifizierten Datensätzen und heterogenen Endgeräten möglich, da nur Weights getauscht werden<br>**Hardware-Effizienz**: Verteilung von ML-Training über Endgeräte reduziert Voraussetzungen für Server-Hardware | Technologie  | **Abhängigkeit**: Training von ML-Modell mit zu geringer Anzahl von Endgeräten kann Modellqualität schwächen<br>**Interoperabilität**: Kompatibilität heterogener Endgeräte und Beherrschung statistischer Heterogenität ist sicherzustellen |
| **Geringe Latenz**: Verteilung des Rechenaufwands über Endgeräte (Edge Computing) erlaubt schnelleres Training mit größeren Datenmengen<br>**Echtzeitvorhersagen**: Können direkt und auch ohne Internetverbindung auf Endgeräten vollzogen werden, wenn aktuelle Modellparameter lokal vorliegen | Leistungsfähigkeit  | **Risikofaktor Internetverbindung**: Nötig für Parameteraustausch zwischen Server und Endgeräten; und erzeugt bei Instabilität Latenzen<br>**Kommunikation als Bottleneck**: Effiziente Methoden nötig, um Kommunikationaufwand zum Parameteraustausch gering zu halten |
| **Datenhoheit**: Gestärkte informationelle Selbstbestimmung durch dezentrales Paradigma<br>**Souveränität**: Nutzende haben stärkere Position gegenüber Datenprozessierenden | Ethik  | **Diskriminierung**: Bilden die am Training beteiligten Endgeräte die Grundgesamtheit nicht adäquat ab, berücksichtigt das verteilte ML-Modell Minderheiten nicht adäquat |

# **Bewertung:** Stimmen aus der Plattform Lernende Systeme

*Verteiltes maschinelles Lernen eröffnet neue Möglichkeiten zur effektiven und skalierbaren Nutzung von Daten, ohne diese teilen zu müssen. Dadurch werden viele hilfreiche Anwendungen mit sensitiven Daten erst möglich.*

**Prof. Dr. Ahmad-Reza Sadeghi**, Leiter System Security Lab, Technische Universität Darmstadt

*Das verteilte maschinelle Lernen kommt ohne das Zusammenführen sensibler Daten aus. So lassen sich Risiken zentraler Datensammlungen vermeiden. Dies bringt Vorteile beim Datenschutz. Jetzt gilt es, die offenen rechtlichen, technischen und organisatorischen Fragen für den rechtskonformen Einsatz zu klären.*

**Dr. h.c. Marit Hansen**, Landesbeauftragte für Datenschutz Schleswig-Holstein

*KI-Systeme in der Medizin können nur erfolgreich sein, wenn ihnen die zum Erreichen hoher Genauigkeit notwendigen Datenmengen zur Verfügung stehen. Verteiltes maschinelles Lernen stellt eine der wichtigsten technischen Möglichkeiten dar, um dies unter Wahrung der informationellen Selbstbestimmung des Einzelnen zu ermöglichen.*

**Prof. Dr. Björn Eskofier**, Lehrstuhl für Maschinelles Lernen und Datenanalytik, Friedrich-Alexander-Universität Erlangen-Nürnberg

## Welche Fragen sind offen?

- **Kosten-Nutzen-Abwägung:** Überwiegen die erwarteten Vorteile des verteilten maschinellen Lernens in puncto Datenschutz und Leistungsfähigkeit tatsächlich?
- **Praxistest:** Wie bewähren sich die technischen Ansätze in der wirtschaftlichen Anwendung? Wie groß ist der Markt?
- **Zielorientierung:** Welche technischen Ansätze eignen sich für welche Anwendungsdomänen?
- **Sicherheit:** Wie können neu entstehende Angriffsfenster (z. B. auf Austausch der Weights zwischen Endgeräten) geschlossen werden, ohne die Leistungsfähigkeit einzuschränken?

## Glossar

**Adversarial Attack (feindlicher Angriff):** Angriff zur Manipulation des Trainingsdatensatzes eines KI-Systems, etwa durch Fehlklassifikationseingaben; dabei schleusen Angreifer schädliche Inhalte in den Filter eines Machine Learning-Algorithmus ein, damit das System einen bestimmten Datensatz falsch klassifiziert.
**Edge KI:** Verlagerung des Trainierens von ML-Modellen auf Endgeräte und allenfalls Austausch von Metadaten mit zentralem Server.
**Machine Learning-Modell (ML-Modell):** Statistisches Modell, das mit Hilfe von Daten darauf trainiert wurde, bestimmte Arten von Mustern zu erkennen. Das ML-Modell ermöglicht es, neue Daten zu analysieren und Vorhersagen über diese Daten zu treffen.
**Weights:** „Ergebnis" eines (lokal) trainierten ML-Modells, das in ein Gesamtmodell zusammengesetzt werden kann; Weights machen den Austausch von Datensätzen obsolet.

## Quellen

**Beyerer, J., Müller-Quade, J. et al. (2022)**: KI-Systeme schützen, Missbrauch verhindern. Maßnahmen und Szenarien in fünf Anwendungsgebieten. Whitepaper der Plattform Lernende Systeme. Online unter: https://doi.org/10.48669/pls_2022-2 (letzter Zugriff: 20.06.2022)

**Houdeau, D. (2022)**: Wie wir KI-Systeme vor Cyberangriffen schützen. Plattform Lernende Systeme. Online unter: https://www.plattform-lernende-systeme.de/reden-und-beitraege-newsreader/wie-wir-ki-systeme-vor-cyberangriffen-schuetzen.html (letzter Zugriff: 20.06.2022)

**Kaissis, G. A. et al. (2020)**: Secure, privacy-preserving and federated machine learning in medical imaging. Nature Machine Intelligence, 2, 305–311.

**Warnat-Herresthal et al. (2021)**: Swarm Learning for decentralized and confidential clinical machine learning. Nature, 594, 265–270.



**Impressum**
Expertise: Björn Eskofier, Marit Hansen, Ahmad-Reza Sadeghi
Redaktion: Jan Biehler, Birgit Obermeier
Herausgeber: Lernende Systeme – Die Plattform für Künstliche Intelligenz | Geschäftsstelle | c/o acatech | Karolinenplatz 4 | D-80333 München
kontakt@plattform-lernende-systeme.de | www.plattform-lernende-systeme.de
Stand: September 2022 | Bildnachweis: Kurt Fuchs, Markus Hansen, Technische Universität Darmstadt, S. 7
Folgen Sie uns auf Twitter und LinkedIn.

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung

acatech
DEUTSCHE AKADEMIE DER TECHNIKWISSENSCHAFTEN