

Federated Learning in der Arbeitsplanung

Bestimmung von Fertigungsvorgängen mithilfe von graphenbasiertem Federated Learning

Marco Hussong*,
Matthias Klar und
Jan C. Aurich

Die Aufbereitung von praxisnahen Trainingsdatensätzen für Deep Learning in der Arbeitsplanung ist eine große Herausforderung. Die Datengrundlage aktueller Ansätze basiert auf synthetisch erstellten 3D-Modellen. Eine solche synthetisierte Generierung von Trainingsdaten bildet jedoch nur sehr begrenzt die industrielle Praxis ab. Vor diesem Hintergrund haben Ansätze ein hohes Potenzial, bei denen aus den Daten mehrerer Unternehmen eine ausreichend große Datengrundlage gebildet werden kann, ohne dass diese an eine zentrale Stelle übertragen werden müssen. Eine im beschriebenen Kontext vielversprechende Methode ist das Federated Learning (FL), für dessen Anwendung in der Arbeitsplanung in diesem Beitrag ein Ansatz beschrieben wird.

Motivation

Die Aufbereitung von praxisnahen Trainingsdatensätzen für Deep Learning in der Arbeitsplanung bildet eine Herausforderung für Industrie und Wissenschaft [1]. Auftragsfertigende Unternehmen verfügen oftmals nicht über eine ausreichende Anzahl an Kunden und damit über zu wenige Aufträge, um die Vielzahl an Arbeitsplanungsdaten zur Verfügung stellen zu können. Zusätzlich beschränkt der Datenschutz die Zusammenführung der Daten, da es sich um sensitive Kundendaten handelt. Zwar ad-

ressieren aktuelle wissenschaftliche Ansätze die Nutzung von Deep Learning in der Arbeitsplanung [2]. Bislang basiert die Datengrundlage dieser Ansätze jedoch auf synthetisch erstellten 3D-CAD-Modellen, zum Beispiel durch die automatisierte Generierung von einzelnen Bauteilfeatures auf Quadern oder Zylindern [3]. Eine synthetisierte Generierung von Trainingsdaten bildet aber nur sehr begrenzt die industrielle Praxis ab, in der Produkte auf sehr unterschiedliche Art und Weise konstruiert sind und demnach eine höhere Diversität und Komplexität aufweisen.

Vor diesem Hintergrund werden Ansätze benötigt, die aus den Arbeitsplanungsdaten mehrerer Unternehmen eine ausreichend große Datenbasis bilden, ohne dass die Arbeitsplanungsdaten an eine zentrale Stelle übertragen werden, um das Risiko von Datenlecks und Datenschutzverletzungen zu reduzieren.

Ein im beschriebenen Kontext vielversprechender Ansatz ist das Federated Learning (FL), bei dem Modelle des Deep Learning auf unterschiedlichen Instanzen trainiert und danach zu einem Gesamtmodell zusammengeführt werden [4]. Für die Nutzung von FL – bezogen auf den Anwendungsfall der Fertigungsvorgangsermittlung in der Arbeitsplanung – wird in diesem Beitrag ein Ansatz modelliert.

Stand der Technik

Der Stand der Technik beschäftigt sich mit der Arbeitsplanung und FL. Im ersten Teil wird der Anwendungsfall zur Bestimmung von Fertigungsvorgängen beschrieben und im zweiten Teil auf FL eingegan-

* Korrespondenzautor

Marco Hussong, M.Sc.; Lehrstuhl für Fertigungstechnik und Betriebsorganisation (FBK), RPTU Kaiserslautern-Landau; Postfach 3049, 67663 Kaiserslautern; Tel.: +49 (0) 205-4305, E-Mail: marco.hussong@rptu.de

Weitere Autoren

Dr.-Ing. Matthias Klar; FBK an der RPTU Kaiserslautern-Landau
Prof. Dr.-Ing. Jan C. Aurich; FBK an der RPTU Kaiserslautern-Landau

Hinweis

Bei diesem Beitrag handelt es sich um einen von den Advisory-Board Mitgliedern des ZWF-Sonderheftes wissenschaftlich begutachteten Fachaufsatz (Peer-Review).

 Dieses Werk ist lizenziert unter der Creative Commons Namensnennung 4.0 International Lizenz.

gen und dabei Anforderungen an das Konzept abgeleitet.

Bestimmung von Fertigungsvorgängen in der Arbeitsplanung

Die Aufgabe der Arbeitsplanung ist die Planung aller einmalig auftretender Fertigungsprozesse für jedes Produkt, um dieses vom Roh- in den Fertigzustand zu überführen. Die Arbeitsplanung bildet damit den Übergang zwischen Entwicklung und Fertigung eines Produkts und wirkt sich maßgeblich auf fertigungstechnische Faktoren, wie z.B. Zeit, Kosten oder Qualität, aus [5].

Ein zentraler Schritt in der Arbeitsplanung ist die Bestimmung der erforderlichen Fertigungsvorgänge, da dieser Einfluss auf die nachfolgenden Schritte, wie z.B. Festlegung der erforderlichen Werkzeuge, hat. Methoden zur Bestimmung von Fertigungsvorgängen analysieren Konstruktionsinformationen aus dem 3D-CAD-Modell [3]. Innerhalb der 3D-CAD-Modelle bestehen die Konstruktionsinformationen aus den Zusammensetzungen geometrischer und topologischer Entitäten des Begrenzungsflächenmodells (engl. Boundary Representation, B-Rep) und Entitäten, die sich zu Produkt- und Fertigungsinformationen (engl. Product and Manufacturing Information, PMI) zusammenfassen lassen. Zu den PMI zählen beispielsweise Angaben zu Toleranzen, wie z.B. Maßtoleranz, oder Ebenheit einer Fläche. Zu den geometrischen und topologischen Darstellungen des 3D-Modells zählen zum Beispiel Flächen, Punkte oder Kanten [6].

Aufgrund der Effektivität im Umgang mit Nichtlinearitäten und Diskontinuitäten im Merkmalsraum der Daten werden zunehmend Ansätze des Deep Learning für die Arbeitsplanung genutzt [3]. Diese basieren auf einer Umwandlung der 3D-CAD-Modelle in eine andere Repräsentation, bspw. Voxel [7], Punktwolke [8] oder projizierte 2D-Ansichten [9], die als Eingabe für Ansätze des Deep Learning genutzt werden. Durch diese Umwandlung werden jedoch Informationen, wie topologische Beziehungen zwischen Flächen, nicht ausreichend berücksichtigt. Weiterhin ist eine konsistente Zuordnung von PMI zu den einzelnen Elementen der neuen Repräsentationen schwierig. Eine Möglichkeit zur Modellierung von topologischen Beziehungen und Berücksichtigung

von Zuordnungen zwischen PMI und Flächen bilden Graphen, weshalb sich deren Einsatz für Deep Learning zur Bestimmung von Fertigungsvorgängen eignet.

Grundlagen, Anwendungen und Anforderungen an das Federated Learning

Federated Learning (FL) ist ein dezentraler Ansatz des maschinellen Lernens, bei dem Modelle auf verteilten Datenquellen trainiert werden, ohne dass die zugrunde liegenden Daten zentral gespeichert oder ausgetauscht werden müssen. Dementsprechend wird FL im Bereich des Internet of Things bzw. Smart Home oder Smart City für Telekommunikation insbesondere auf mobilen Endgeräten oder autonomes Fahren eingesetzt, wo üblicherweise Daten vieler Instanzen zusammengeführt werden müssen. Weitere Anwendungsfelder sind der Gesundheits- oder Finanzsektor aufgrund deren hohen Anforderungen an die Privatsphäre der Daten [10]. Die Herausforderungen der Methode liegen bspw. in der Heterogenität der Daten, die das Training erschwert und spezialisierte Aggregationsmethoden erfordert. Lokale Berechnungen können ressourcenintensiv sein und ein hoher Kommunikationsaufwand kann vorliegen, da Modellparameter regelmäßig zwischen den Geräten und dem Server ausgetauscht werden müssen. Zudem ist FL nicht gänzlich frei von Sicherheitsrisiken, da über die Modellparameter Rückschlüsse auf die Datenstruktur möglich sind [11].

Für das Training beim FL wird das zentrale Modell für ausgewählte Instanzen bereitgestellt, damit diese das Modell mit ihren jeweils vorliegenden Daten trainieren können. Der Trainingsprozess findet also dezentral und lokal auf den verschiedenen Endgeräten der Instanzen statt. Bei jeder Instanz wird ein lokales Update des Modells (Trainingsschritt) durchgeführt. Anschließend werden lediglich die Modellparameter gesammelt und nicht die eigentlichen Rohdaten. Eine Aggregation der gesammelten Modellparameter bildet die aktualisierte Version des zentralen Modells. Der gesamte Vorgang ist ein iterativer Prozess. Das aktualisierte Modell wird demnach erneut an die Instanzen gesendet, damit diese in weiteren Runden das Training fortsetzen können, bis eine gewünschte Leistung oder Genauigkeit erzielt ist [12].

Grundsätzlich kann zwischen horizontalen und vertikalem bzw. zentralem und dezentralem FL unterschieden werden. Beim zentralen FL existiert ein zentraler Server, der die Koordination des Trainings übernimmt. Dieser Server sammelt Modell-Updates von den Instanzen, aggregiert diese und koordiniert weitere Iterationsschritte. Beim dezentralen FL existiert kein zentraler Koordinator. Stattdessen kommunizieren die Instanzen direkt miteinander, um Modell-Updates auszutauschen und zu aggregieren [13]. Das horizontale Federated Learning bezieht sich daher auf Szenarien, in denen verschiedene Instanzen ähnliche oder identische Datenmerkmale, aber unterschiedliche Datenbeispiele besitzen. Beim vertikalen FL besitzen die Instanzen dagegen unterschiedliche Datenmerkmale bei ähnlichen oder identischen Datenbeispielen. Die Einordnung des Anwendungsfalls in die jeweilige Kategorie hat dabei Auswirkungen auf die Gestaltung des FL. Da alle Instanzen beim horizontalen FL ähnliche Datenmerkmale, aber unterschiedliche Beispiele haben, ist die Modellarchitektur in der Regel auf diese Merkmale abgestimmt. Bei vertikalem FL müssen die Modelle für Merkmale, die bei verschiedenen Instanzen vorhanden sind, kombiniert werden. Dies erfordert spezielle Modellarchitekturen, die Merkmale verschiedener Instanzen integrieren, ohne die Daten direkt auszutauschen [14].

Trotz der Gestaltung des FL als dezentraler Trainingsmethode, bei der die Daten auf den Endgeräten verbleiben können, existieren weiterhin Möglichkeiten, Rückschlüsse auf die zugrunde liegenden Daten zu ziehen [12]. Selbst wenn die Rohdaten nicht übertragen werden, könnten Metadaten oder andere Informationen, die während der Kommunikation zwischen Geräten ausgetauscht werden (wie zum Beispiel die Größe der Daten oder Trainingsgewichte), Rückschlüsse auf die Daten zulassen. Dies gilt es bei der Gestaltung des FL zu berücksichtigen und entsprechende Datenschutzmechanismen anzuwenden. Für eine Zusammenfassung existierender Ansätze wird auf *Chen et al.* [15] verwiesen.

Weiterhin ist beim Einsatz von FL die Heterogenität der Daten zu berücksichtigen. Für verschiedene Instanzen können die Datenverteilungen der Merkmale und

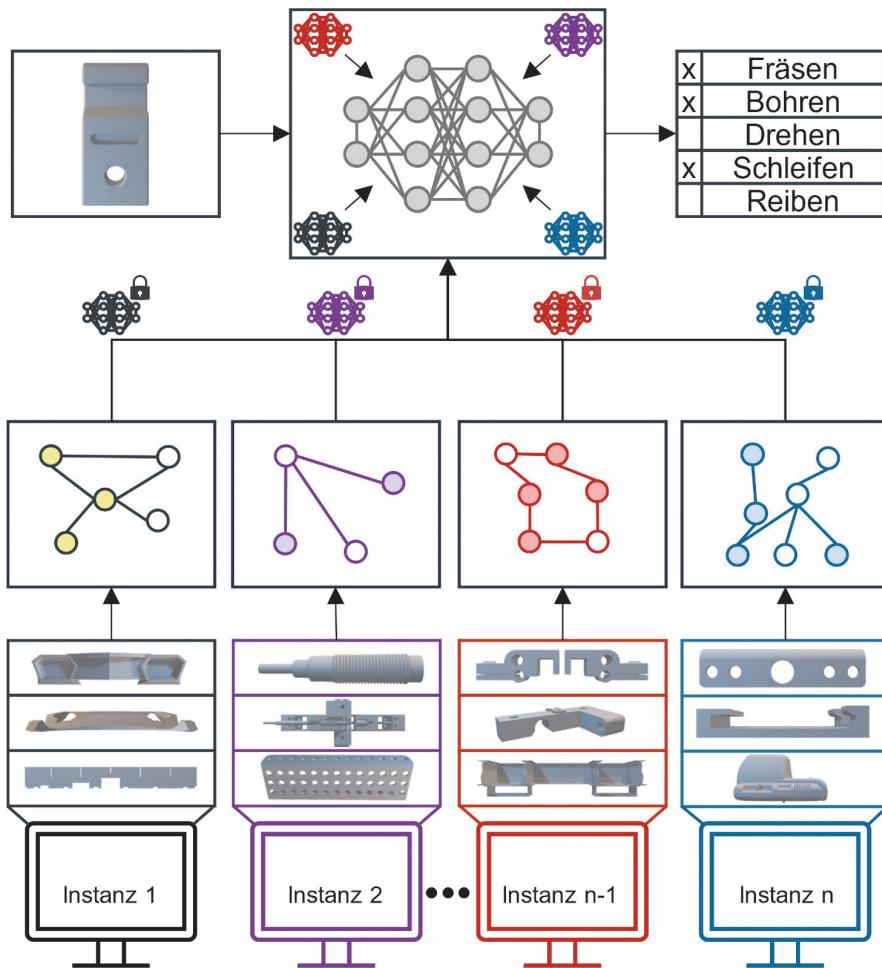


Bild 1. Zusammenfassung des Ansatzes für die Ermittlung von Fertigungsvorgängen durch graphenbasiertes Federated Learning

Labels oder die Anzahl an verfügbaren Datenpunkten stark variieren [12]. Unter Berücksichtigung der Bestimmung von Fertigungsvorgängen in der Arbeitsplanung, bei der der Anwendungsfall der Arbeitsplanung durch graphenbasiertes Deep Learning modelliert werden soll, erweitern sich die Möglichkeiten auftretender Datenheterogenität, da bei Graphen zusätzlich die Verteilung der Topologie unterschiedlich sein kann [13].

Zusammenfassend werden die folgenden Aspekte bei der Gestaltung des FL für den Anwendungsfall in der Arbeitsplanung berücksichtigt:

- Gestaltung als horizontales oder vertikales bzw. zentrales oder dezentrales FL,
- Bestimmung eines Algorithmus für die zentrale Aggregation der Modellparameter,

- Handhabung von Datenheterogenität sowie
- Auswahl von Sicherheitsmechanismen zum Datenschutz.

Anwendung von FL zur Bestimmung von Fertigungsvorgängen in der Arbeitsplanung

Bild 1 verdeutlicht den Aufbau des Systems für graphenbasiertes Federated Learning zur Ermittlung von Fertigungsvorgängen in der Arbeitsplanung. Es werden die Daten verschiedener Unternehmen als Instanzen des FL genutzt. Ein zentraler Server übernimmt die Koordination des Trainings. Dieser wählt aus einer Gesamtmenge an Instanzen eine definierte Anzahl aus und initiiert das jeweilige Training. Durch die Ge-

staltung als zentrales FL bleibt die Struktur des Systems übersichtlich, wodurch die Kommunikation und Synchronisation des Trainings effizienter umgesetzt werden kann. Gleichzeitig ist ein solches System aufgrund der direkten Kommunikation zwischen Server und Partei aber auch anfälliger für Angriffe dieses Kommunikationswegs, sodass durch Zugriff auf die Modellparameter Rückschlüsse auf die Daten gemacht werden können. Insgesamt führen höhere Vertrauensannahmen zu einfacheren Datenschutzmechanismen, was die Systemeffizienz erhöht, durch geringere Rechenzeiten und Netzwerkommunikation, aber gleichzeitig Sicherheitsrisiken erhöht. Bei niedrigem Vertrauen kann auf eine Differential-Privacy-Methode zurückgegriffen werden [13].

Dabei wird Rauschen auf die zu sendenden Modell-Updates gegeben, was die Rückschlüsse auf verwendete Trainingsdaten schwer bis unmöglich macht. Jedoch geht diese Methode mit einem Genauigkeitsverlust im zentralen Modell durch das Rauschen einher. Ein beispielhafter Ansatz für eine Differential-Privacy-Methode bietet PRIVATE-FL [16]. Bei höheren Vertrauensannahmen kann zum Schutz der Kommunikationswege und der Privatsphäre der Daten eine homomorphe Verschlüsselung eingesetzt werden. Homomorphe Verschlüsselung ist eine kryptografische Technik, die es ermöglicht, Berechnungen direkt auf verschlüsselten Daten durchzuführen, ohne diese vorher zu entschlüsseln. Dadurch können sensible Daten verarbeitet werden, ohne jemals entschlüsselt werden zu müssen. Ein Beispiel für homomorphe Verschlüsselung bieten Liu *et al.* [17]. Weitere Methoden zur Steigerung der Datensicherheit sind vertrauenswürdige Ausführungsumgebungen oder sichere Mehrparteienberechnung [15]. Zur Aggregation der Modellparameter auf dem zentralen Server wird eine aufmerksame Methode eingesetzt. Dadurch werden den einzelnen lokalen Modellen der Parteien lernbare Gewichte zugeordnet, um den Konvergenzprozess zu beschleunigen. Ein solches Vorgehen wurde schon erfolgreich für nicht-euklidische Graphdaten gezeigt [18]. Der Einfluss der Datenheterogenität kann zusätzlich durch weitere Maßnahmen

abgemildert werden. Dazu zählen die Einführung von Regulierungstermen in der Fehlerfunktion, Wissensdestillation oder Modellinterpolation [19].

In Bezug auf die Bestimmung von Fertigungsvorgängen in der Arbeitsplanung können die Daten in ihrer Rohform (3D-CAD-Modell und Fertigungsvorgänge) nicht direkt für das Training eines Deep-Learning-Modells genutzt werden und werden daher zu Graphen transformiert. Durch die Überführung in Graphen und die damit einhergehende Definition von Merkmalen kann der Ansatz in das horizontale FL eingeordnet werden. Die 3D-CAD-Modelle werden in drei verschiedene Graphen umgewandelt, die als Eingabe für Deep Learning genutzt werden. Die erste Struktur ist ein merkmalsindizierter Adjazenzgraph. Durch diesen werden die topologischen Beziehungen der Flächen zueinander abgebildet. Die Knoten bilden somit die Flächen des 3D-CAD-Modells und die Kanten die Verbindungen der Flächen. Merkmalsinduziert bedeutet in diesem Zusammenhang, dass den Flächen und Kanten des Graphen bestimmte Attribute wie Länge oder Flächeninhalte zugeordnet werden können. Durch eine solche Beschreibung des 3D-CAD-Modells geht die Information zur Form der Fläche weitestgehend verloren. Zwar kann durch die Beschreibung eines Merkmals wie Flächentyp die Form einer Fläche (z.B. zylindrisch) berücksichtigt werden, für komplexere Formen kann diese Beschreibung jedoch nicht ausreichen. Ein weiterer Graph bildet demnach ein Gitternetz (Mesh) einzelner Flächen. Durch die Analyse dieser soll der Deep-Learning-Ansatz befähigt werden, unterschiedliche Flächenformen differenzieren zu können. Als letztes werden die Abhängigkeiten der PMI mithilfe eines Graphen modelliert. Im Wesentlichen können diese in indirekte und direkte PMI unterteilt werden. Direkte PMI, wie z.B. Zylindrizität, kann direkt einer Fläche zugeordnet werden. Indirekte PMI, wie z.B. Parallelität, wird über ein sogenanntes Datum im 3D-CAD-Modell zugeordnet. Dadurch kann ein Graph definiert werden, indem die Flächen des CAD-Modells die Knoten bilden, deren Attribute direkte PMI sind und die indirekten PMI über Kanten mit Attributen im Graph modelliert werden.

Zusammenfassung und Ausblick

In diesem Beitrag wurde ein Ansatz vorgestellt, der die Ermittlung von Fertigungsvorgängen aus 3D-CAD-Modellen für die Arbeitsplanung durch FL ermöglicht. Dazu werden die 3D-CAD-Modelle in Graphen umgewandelt, die anschließend mithilfe von FL analysiert werden. Dabei wurde das FL als horizontal und zentral modelliert. Es wurden Mechanismen zur Reduktion von Datenheterogenität und zur Erhöhung von Datensicherheit aufgezeigt und diskutiert. Des Weiteren wurde eine Methode für die Aggregation der Modellparameter beschrieben. Der dargestellte Ansatz wird zukünftig umgesetzt, in einen Software-Demonstrator integriert und für die Arbeitsplanung unterschiedlicher Unternehmen getestet.

Literatur

1. Klose, G.; Bornemann, H.; Thierstein, J. et al.: Künstliche Intelligenz – Herausforderungen und Chancen für die rheinlandpfälzischen KMU. Prognos AG, 2022
2. Hussong, M.; Varshneya, S.; Rüdiger-Flore, P. et al.: A Process Planning System Using Deep Artificial Neural Networks for the Prediction of Operation Sequences. *Procedia CIRP* 120 (2023), S. 135–140
DOI:10.1016/j.procir.2023.08.025
3. Zhao, C.; Melkote, S. N.: Learning the Manufacturing Capabilities of Machining and Finishing Processes Using a Deep Neural Network Model. *Journal of Intelligent Manufacturing* 35 (2024) 4, S. 1845–1865
DOI:10.1007/s10845-023-02134-z
4. McMahan, B.; Moore, E.; Ramage, D. et al.: Communication-Efficient Learning of Deep Networks from Decentralized Data. *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics* 54 (2017), S. 1273–1282
DOI:10.48550/arXiv.1602.05629
5. Eversheim, W.: Organisation in der Produktionstechnik. Fertigung und Montage. Studium und Praxis, Bd. 4. VDI-Verl., Düsseldorf 1989
DOI:10.1007/978-3-642-61344-9
6. ISO 10303-21:2016. Industrielle Automatisierungssysteme und Integration – Produktionsdatendarstellung und -austausch – Teil 21: Implementierungsmethoden: Klartext-Kodierung der Austauschstruktur. DIN 2016
7. Zhang, Z.; Jaiswal, P.; Rai, R.: FeatureNet: Machining Feature Recognition Based on 3D Convolutional Neural Network. *Computer-Aided Design* 101 (2018), S. 12–22
DOI:10.1016/j.cad.2018.03.006
8. Ma, Y.; Zhang, Y.; Luo, X.: Automatic Recognition of Machining Features Based on Point Cloud Data Using Convolution Neural Networks. In: *Proceedings of the 2019 International Conference on Artificial Intelligence and Computer Science* (2019), S. 229–235
DOI:10.1145/3349341.3349407
9. Shi, P.; Qi, Q.; Qin, Y. et al.: A Novel Learning-based Feature Recognition Method Using Multiple Sectional View Representation. *Journal of Intelligent Manufacturing* 31 (2020) 5, S. 1291–1309
DOI:10.1007/s10845-020-01533-w
10. Li, L.; Fan, Y.; Tse, M.; Lin, K.-Y.: A Review of Applications in Federated Learning. *Computers & Industrial Engineering* 149 (2020) 5
DOI:10.1016/j.cie.2020.106854
11. Ji, S.; Tan, Y.; Saravirta, T. et al.: Emerging Trends in Federated Learning: from Model Fusion to Federated X Learning. *International Journal of Machine Learning and Cybernetics* 15 (2024) 9, S. 3769–3790
DOI:10.1007/s13042-024-02119-1
12. Banabilah, S.; Aloqaily, M.; Alsayed, E. et al.: Federated Learning Review: Fundamentals, Enabling Technologies, and Future Applications. *Information Processing & Management* 59 (2022) 6
DOI:10.1016/j.ipm.2022.103061
13. Liu, R.; Xing, P.; Deng, Z. et al.: Federated Graph Neural Networks: Overview, Techniques, and Challenges. *IEEE Transactions on Neural Networks and Learning Systems* (2024), S. 1–17
DOI:10.1109/TNNLS.2024.3360429
14. Yurdem, B.; Kuzlu, M.; Gullu, M. K. et al.: Federated Learning: Overview, Strategies, Applications, Tools and Future Directions. *Heliyon* 10 (2024) 19
DOI:10.1016/j.heliyon.2024.e38137
15. Chen, J.; Yan, H.; Liu, Z. et al.: When Federated Learning Meets Privacy-Preserving Computation. *ACM Computing Surveys* 56 (2024) 12, S. 1–36
DOI:10.1145/3679013
16. Yang, Y.; Hui, B.; Yuan, H. et al.: PrivateFL: Accurate, Differentially Private Federated Learning via Personalized Data Transformation. *Proceedings of the 32nd USENIX Security Symposium* (2023), S. 1595–1611
17. Liu, X.; Li, H.; Xu, G. et al.: Privacy-Enhanced Federated Learning Against Poisoning Adversaries. *IEEE Transactions on Information Forensics and Security* 16 (2021), S. 4574–4588
DOI:10.1109/TIFS.2021.3108434
18. Hu, K.; Wu, J.; Li, Y. et al.: FedGCN: Federated Learning-Based Graph Convolutional Networks for Non-Euclidean Spatial Data. *Mathematics* 10 (2022) 6
DOI:10.3390/math10061000

19. Qi, P.; Chiaro, D.; Guzzo, A. et al.: Model Aggregation Techniques in Federated Learning: A Comprehensive Survey. Future Generation Computer Systems 150 (2024), S. 272–293
DOI:10.1016/j.future.2023.09.008

Die Autoren dieses Beitrags

Marco Hussong, M. Sc., ist wissenschaftlicher Mitarbeiter am Lehrstuhl für Fertigungstechnik und Betriebsorganisation (FBK) der RPTU in Kaiserslautern und forscht im Bereich Produktionssysteme zu den Themen digitale Technologien in der Produktion.

Dr.-Ing. Matthias Klar, ist Oberingenieur am Lehrstuhl für Fertigungstechnik und Betriebsorganisation (FBK) der RPTU in Kaiserslautern und forscht im Bereich Produktionssysteme zu den Themen digitale Technologien in der Produktion, Nachhaltigkeit in der Produktion und Additive Fertigung.

Prof. Dr.-Ing. Jan C. Aurich leitet den Lehrstuhl für Fertigungstechnik und Betriebsorganisation (FBK) an der RPTU in Kaiserslautern. Seine Forschungsinteressen liegen in den Bereichen Ultrapräzisions- und Mikrobearbeitung, Zerspanetechnologie, Additive Fertigung, Digitale Technologien für Produktionssysteme und Nachhaltigkeit in der Produktion.

Abstract

Federated Learning in Process Planning – Selection of Manufacturing Processes Using Federated Graph Learning. Preparing of training datasets for deep learning in process planning presents a significant challenge. The data basis for various approaches has been based on synthetically created 3D models. However,

such synthetic generation of training data only limitedly reflects industrial practice. Considering this background, approaches that can form a sufficiently large data basis from several companies' data without transferring the data to a central location are required. A promising approach in the described context is Federated Learning (FL). Therefore, this paper focuses on modeling an FL approach for process planning.

Danksagung

Die Autoren danken dem Ministerium für Wirtschaft, Verkehr, Landwirtschaft und Weinbau des Landes Rheinland-Pfalz für die finanzielle Unterstützung im Rahmen des Forschungsvorhabens „KI4KMU-RLP – Erforschung und Transfer nachhaltiger KI-Innovationen für produzierende KMU in Rheinland-Pfalz“.

Schlüsselwörter

Arbeitsplanung, Fertigungsvorgangsermittlung, Deep Learning, Federated Learning

Keywords

Process Planning, Manufacturing Process Selection, Deep Learning, Federated Learning

Bibliography

DOI:10.1515/zwf-2024-0131
ZWF 120 (2025) Special Issue; page 269 – 273
Open Access. © 2025 bei den Autoren,
publiziert von De Gruyter. 
Dieses Werk ist lizenziert unter der Creative Commons Namensnennung 4.0 International Lizenz.
ISSN 0947-0085 · e-ISSN 2511-0896