开题报告草稿

文献综述与调研报告

研究背景以及研究意义

随着AlphaGo[5]在围棋领域击败了顶尖的人类玩家，我们真正见证了人工智能（AI）的巨大潜力，并期待在许多领域中使用更复杂、更尖端的AI技术，包括无人驾驶汽车、医疗保健、金融等。然而，现实世界的情况有些令人失望：除少数行业外，大多数领域只有有限的数据或低质量的数据，这使得人工智能技术的实现比我们想象的更加困难。通过跨组织传输数据，是否有可能将数据融合到一个公共节点上？事实上，在许多情况下，要打破数据源之间的壁垒是非常困难的。一般来说，任何人工智能项目所需的数据都涉及多种类型。例如，在人工智能驱动的产品推荐服务中，产品销售者拥有关于产品的信息、用户购买的数据，但没有描述用户购买能力和支付习惯的数据。在大多数行业中，数据以孤立的岛屿形式存在。由于行业竞争、隐私安全和复杂的管理程序，即使是同一公司不同部门之间的数据集成也面临巨大阻力。并且几乎不可能整合分散在国家和机构中的数据。

与此同时，随着越来越多的大公司意识到数据安全和用户隐私受到损害，对数据隐私和安全的重视已成为一个世界性的重大问题。关于公共数据泄露的消息引起了公共媒体和政府的极大关注。例如，Facebook的数据泄露事件引起了广泛的抗议[6]。作为回应，世界各国正在加强数据安全和隐私保护方面的法律。例如，欧盟于2018年5月25日实施的《通用数据保护条例》（GDPR）[7]。GDPR旨在保护用户的个人隐私和数据安全。它要求企业在用户协议中使用清晰明了的语言，并授予用户“被遗忘的权利”，即用户可以删除或撤回其个人数据。违反该法案的公司将面临严厉的罚款。美国和中国正在制定类似的隐私和安全法案。例如，2017年颁布的中国《网络安全法》和《民法通则》要求互联网企业不得泄露或篡改其收集的个人信息，并且在与第三方进行数据交易时，他们需要确保拟定合同遵守法律数据保护义务。这些法规的制定显然将有助于建立一个更为公民化的社会，但也将对当今人工智能中常用的数据处理程序提出新的挑战。

传统的训练方法是集中式地训练，即将客户的数据从边缘端上传至中央服务器进行集中式地训练。这一传统程序面临上述新数据法规和法律的挑战。此外，由于用户可能不清楚模型的未来用途，这些交易违反了GDPR等法律。因此各公司需要对传统的数据采集方式、数据训练方式等重新进行考虑。因此，我们面临一个两难境地，即我们的数据是孤立的岛屿，但在许多情况下，我们被禁止收集、融合和使用数据到不同的地方进行人工智能处理。如何合法地解决数据碎片和隔离问题是当今人工智能研究者和实践者面临的一个重大挑战。

为了解决上述问题，Google提出联邦学习[8, 9, 10]，后在2017年提出了联邦学习领域中最经典的FedAvg算法[2]，即一种新的模型训练框架：客户端无需将本地数据上传到中央服务器，而是将本地训练的模型的更新梯度上传，中央服务器聚合这些梯度以更新全局模型，后将全局模型参数下发给客户端，进行新一轮的训练直到收敛或者达到最大训练轮数。因为上传的并非本地数据，而是模型更新的梯度，因此在一定程度上能降低数据泄露的风险。由于上传的并不是客户的数据，因此模型的精度会有所降低，同时因为隐私保护，服务器无法了解客户端的数据分布情况，因此在计算全局模型的参数时无法利用数据的分布情况来调整对应的权重，存在数据非独立同分布的挑战[3]。

但是，联邦学习中的服务器既不能访问客户端的数据，也不能完全控制客户端的行为，因此在联邦学习的过程中，客户端可能会偏离正常行为，上传的参数可能会扰乱全局模型的训练，降低全局模型的精度。造成客户端异常行为的原因可能是客户端本身的设备出现了故障，或者该客户端本身是伪装的恶意攻击者。对于这些异常行为的客户端，及时将他们检测出来非常重要：一是可以降低异常客户端对全局模型的影响，因为异常客户端上传的异常参数将参与到全局模型的计算中，因此可能会扰乱其他正常客户端的训练，那么及时地检测将异常客户端的影响降至最低；二是可以避免全局模型泄露给非预期的客户，客户端的异常行为可能不是其本身的故障造成的，而是本身就是伪装的恶意攻击者，因为在联邦学习的过程中，中央服务器需要计算全局模型并且下发该参数作为客户端新一轮训练的初始参数，因此恶意攻击者很容易获得全局模型的信息；三是可以防止向异常客户分配激励回报[4]。

因此，如何快速并准确地检测出异常客户端对模型的精度的提升和避免模型的泄露至关重要。在全局模型收敛之前，客户端上传的参数是动态变化的，为检测算法增加了难度；同时因为联邦学习存在数据异构的问题，数据异构将导致部分客户端上传的参数与全局模型间的偏差增大，而这些正

常的客户端的行为与异常客户端的行为高度重合，容易被检测算法误判，导致这些客户端的利益受到损害。

[1] H Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, pages 1273–1282, 2017 (original version on arxiv Feb. 2016).

[2] McMahan B, Moore E, Ramage D, et al. Communication-efficient learning of deep networks from decentralized data[C]//Artificial intelligence and statistics. PMLR, 2017: 1273-1282.

[3] Zhao Y, Li M, Lai L, et al. Federated learning with non-iid data[J]. arXiv preprint arXiv:1806.00582, 2018.

[4] Zhan Y, Zhang J, Hong Z, et al. A Survey of Incentive Mechanism Design for Federated Learning[J]. IEEE Transactions on Emerging Topics in Computing, 2021.

[5] David Silver, Aja Huang, Christopher J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. 2016. Mastering the game of Go with deep neural networks and tree search. Nature 529 (2016), 484–503. http://www.nature.com/nature/journal/v529/n7587/full/nature16961.html

[6] Wikipedia. 2018. https://en.wikipedia.org/wiki/Facebook-Cambridge_Analytica_data_scandal.

[7] EU. 2016. REGULATION (EU) 2016/679 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). Available at: https://eur-lex.europa. eu/legal-content/EN/TXT (2016)

[8] Jakub Konecný, H. Brendan McMahan, Daniel Ramage, and Peter Richtárik. 2016. Federated Optimization: Distributed Machine Learning for On-Device Intelligence. CoRR abs/1610.02527 (2016). arXiv:1610.02527 http://arxiv.org/abs/1610. 02527

[9] Jakub Konecný, H. Brendan McMahan, Felix X. Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. 2016. Federated Learning: Strategies for Improving Communication Efficiency. CoRR abs/1610.05492 (2016). arXiv:1610.05492 http://arxiv.org/abs/1610.05492

[10] H. Brendan McMahan, Eider Moore, Daniel Ramage, and Blaise Agüera y Arcas. 2016. Federated Learning of Deep Networks using Model Averaging. CoRR abs/1602.05629 (2016). arXiv:1602.05629 http://arxiv.org/abs/1602.05629

研究综述

在联邦学习的环境中，客户端是自治的，中央服务器无法访问客户端的本地数据，也无法完全控制客户端的行为，因此在训练的过程中，客户端可能会偏离正常行为而上传异常数据。造成该异常的原因可能是因为客户端本身发生了故障或客户端本身是恶意客户。就鲁棒性而言，联邦学习系统容易受到数据中毒[24]、[25]和模型中毒攻击[26]、[27]、[28]、[29]。恶意参与者可以通过故意改变其本地数据（数据中毒）或梯度上传（模型中毒），攻击全局模型的收敛或将后门触发器植入全局模型。模型中毒攻击可进一步分为（1）拜占庭式攻击，对手旨在破坏全局模型的收敛性和性能[30]，[31]；和（2）后门攻击，其中对手的目标是将后门触发器植入全局模型中，从而欺骗模型在子任务上不断预测对手类别，同时在主任务上保持良好性能[26]，[27]。后门模型中毒攻击通常利用数据中毒来获取中毒参数更新[24]、[26]、[27]。

近几年，大部分的防御算法的研究都是围绕聚合或者异常检测展开的。

基于聚合的鲁棒算法核心思想在于使用一些鲁棒聚合的方式来聚合各个客户机的梯度，使用聚合结果来更新全局模型以降低拜占庭节点对全局模型的影响程度。比较经典的鲁棒聚合算法有：1）几何中位数(Geometric median)[12]：中央服务器计算所有上传的梯度的几何中位数，使用该几何中位数来更新全局模型。2）边际截断均值(Marginal Trimmed Mean)[13,14]：中央服务器按比例去掉梯度的各个维度上的最大值和最小值，然

后对剩余的值取平均作为最终的更新梯度。3）各维度上的中位数[15]：中央服务器对所有梯度按维取中位数，各维上的中位数最终组合成更新梯度。4）在Krum算法[11]中，中央服务器获取到所有的客户机上传的梯度之后，计算每个梯度到其邻近n-f-2个梯度的距离之和（其中，n表示客户机的总数量，f表示拜占庭客户机数量），该距离之和作为对应梯度的分数，最终选取分数最小的梯度来更新全局模型。5）Zeno算法[32]对梯度进行打分，更新该梯度后的全局模型的精度下降时，那么分数就会减少，同时梯度的量级越大，分数也会越少，因此使得全局模型往有利的方向上训练并且量级小的梯度被认为是良好的梯度，最终对这些分数进行排序，取排在前列的梯度的平均值来更新全局模型。6）DETOX算法[33]使用层级聚合的方式过滤拜占庭梯度，每层的聚合方式可以是目前任何一种鲁邦聚合的算法，如Krum算法[11]、边际截断均值[13, 14]算法等。该文章[33]证明了，通过第一层聚合后，可以将拜占庭节点的数量指数级别地减少，而计算节点的总数只是线性减少。这意味着用计算资源的线性减少换来拜占庭节点的指数级减少。然而，这些算法的缺点就是：它们都假设数据是独立同分布的。然而这对联邦学习是不适用的。在联邦学习的环境下，数据很有可能是非独立同分布的。数据的依赖性和不一致可能源于特定的用户、特定的地理位置或者特定的时间窗口。这些非独立同分布的现象与“数据偏移”紧密相关，文章[16,17]提到了“数据偏移”的现象，并对训练集的分布和测试集的分布之间的差异进行了研究。

基于异常检测的鲁棒算法被认为是一种更主动的防御类型，它可以明确检测恶意更新并防止其对系统的影响。文章[34]提出AUROR算法，

[11] Blanchard, Peva, et al. "Machine learning with adversaries: Byzantine tolerant gradient descent." *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 2017.

[12] Chen Y, Su L, Xu J. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent[J]. Proceedings of the ACM on Measurement and Analysis of Computing Systems, 2017, 1(2): 1-25.

[13] Yin D, Chen Y, Kannan R, et al. Byzantine-robust distributed learning: Towards optimal statistical rates[C]//International Conference on Machine Learning. PMLR, 2018: 5650-5659.

[14] Xie C, Koyejo O, Gupta I. Phocas: dimensional byzantine-resilient stochastic gradient descent[J]. arXiv preprint arXiv:1805.09682, 2018.

[15] Xie C, Koyejo O, Gupta I. Generalized byzantine-tolerant sgd[J]. arXiv preprint arXiv:1802.10116, 2018.

[16] Jose G. Moreno-Torres, Troy Raeder, Roc´ıO Alaiz-Rodr´ıGuez, Nitesh V. Chawla, and Francisco Herrera. A unifying view on dataset shift in classification. Pattern Recogn., 45(1), January 2012.

[17] Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil D. Lawrence. Dataset Shift in Machine Learning. The MIT Press, 2009. ISBN 0262170051, 9780262170055.

[18]Chen L, Wang H, Charles Z, et al. Draco: Byzantine-resilient distributed training via redundant gradients[C]//International Conference on Machine Learning. PMLR, 2018: 903-912.

[24] H.Wang, K. Sreenivasan, S. Rajput, H. Vishwakarma, S. Agarwal, J.-y. Sohn, K. Lee, and D. Papailiopoulos, “Attack of the tails: Yes, you really can backdoor federated learning,” NeurIPS, 2020.

[25] C. Xie, K. Huang, P. Chen, and B. Li, “DBA: distributed backdoor attacks against federated learning,” in 8th International Conference on Learning Representations, 2020.

[26] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, “How to backdoor federated learning,” CoRR, arXiv:1807.00459, 2018.

[27] A. N. Bhagoji, S. Chakraborty, P. Mittal, and S. Calo, “Ana- lyzing federated learning through an adversarial lens,” CoRR, arXiv:1811.12470, 2018.

[28] C. Fung, C. J. Yoon, and I. Beschastnikh, “Mitigating sybils in federated learning poisoning,” CoRR, arXiv:1808.04866, 2018.

[29] Z. Sun, P. Kairouz, A. T. Suresh, and H. B. McMahan, “Can you really backdoor federated learning?” arXiv preprint arXiv:1911.07963, 2019.

[30] P. Blanchard, R. Guerraoui, J. Stainer et al., “Machine learn- ing with adversaries: Byzantine tolerant gradient descent,” in NeurIPS, 2017, pp. 119–129

[31] J. Bernstein, J. Zhao, K. Azizzadenesheli, and A. Anandkumar, “signSGD with majority vote is communication efficient and byzantine fault tolerant,” in In Seventh International Conference on Learning Representations (ICLR), 2019

[32] Xie C, Koyejo S, Gupta I. Zeno: Distributed stochastic gradient descent with suspicion-based fault-tolerance[C]//International Conference on Machine Learning. PMLR, 2019: 6893-6901.

[33] Rajput S, Wang H, Charles Z, et al. DETOX: A redundancy-based framework for faster and more robust gradient aggregation[J]. arXiv preprint arXiv:1907.12205, 2019.

[34] Shen S, Tople S, Saxena P. Auror: Defending against poisoning attacks in collaborative deep learning systems[C]//Proceedings of the 32nd Annual Conference on Computer Security Applications. 2016: 508-519.

本课题的基本内容，预计解决的难题

研究内容

1、基于深度聚类的异构数据划分。异构数据的划分有利于将数据不平衡因素对异常检测算法的影响尽可能地降低，为后续簇内异常检测做铺垫。在联邦学习的环境下，本地训练的模型有大有小，如果模型的参数数量过大，那么用于异构数据划分的特征向量的维数就会过大，那么传统的聚类方法就无法应对，因此考虑使用基于深度学习的聚类方法对异构数据进行划分。

2、簇信息丰富的异常检测算法。当异构数据划分完成后，如果簇内信息丰富即簇内客户机的数量足够，那么可以利用簇内其他客户机的权重更新信息来判断当前客户机的更新行为是否异常。

3、簇信息缺乏的异常检测算法。当异构数据划分完成后，如果簇内信息缺乏即簇内客户机的数量较少，那么可以考虑预训练一个全局的异常检测模型，簇信息缺乏的情况就交由该模型来诊断。

4、……

> 因为client的本地模型也可能是一个层数多的神经网络，因此上传的参数量可能会非常多，例如神经元有50个，那么就有至少50个参数，如果将他们都视为特征向量来进行异构划分，那么就是50维，高维使得传统的方法处理起来非常困难，因此我这里考虑用基于deep learning的聚类方法来处理。

> 这里有一个难点，因为类不平衡的因素，因此类不平衡的种类会非常多（例如有十个类，某个client上只有两个类的数据，那么可能数就是C_{10}^{2}），因此在构造训练集时肯定不可能覆盖到所有的情况，因此使用机器学习的方式训练出来的模型的精度难免会收到影响，所以这是为什么我优先考虑传统的异常检测算法。当簇信息缺乏的时候（例如一个簇中就只有少数的client），那么传统的异常检测算法就不可行了，那么这种情况又应该如何处理?

研究目标

第一块先讲目前研究的不足：数据不平衡分布在联邦学习中的普遍性和挑战性，目前联邦学习对类的不平衡问题大部分都集中在优化损失函数上，很少对客户端直接进行类信息的划分，然后讲目前联邦学习中的异常检测只是很简单地将客户端上传参数的序列直接作为训练数据，而没有考虑数据不平衡状态下的数据的偏离，这种数据的偏离可能会被检测算法误判。

第二块讲研究目标：

1、在联邦学习初期使得服务器能很快掌握客户端数据分布情况的大致信息

2、无论是簇信息丰富还是缺乏等情况，检测模型都能很好地进行检测

3、……

拟解决的关键问题

1、如何联邦学习的条件限制下使服务器获取客户端数据分布的大致信息并划分成簇

2、当簇信息丰富的时候如何进行异常检测

2、当簇信息缺乏的时候如何进行异常检测

课题的研究方法，技术路线

研究方法

- 文献综合研究法
- 定性分析法
- 实验法

技术路线

这部分应该详细地说明自己的步骤，画流程图。

目前确定下来的是：先使用deep clustering确定异构数据划分，然后根据簇内信息量来决定使用传统的方式进行检测还是使用基于机器学习的方式检测。

进度计划

这一部分主要写进度的安排之类的。