



# 暨南大学

## 本科生课程论文

论文题目：

基于区块链的医疗健康数据分享平台隐私保护：大型语言模型的应用与挑战

学 院： 智能科学与工程学院

学 系：

专 业： 人工智能

课程名称： 区块链技术及应用

学生姓名： 赵俊文

学 号： 2022104002

指导教师： 赵阔

2024 年 12 月 26 日

目录

一、引言..... 2

二、区块链技术在医疗健康数据分享平台中的基础应用 ..... 3

    1. 不可篡改性保障数据完整性 ..... 3

    2. 去中心化优化数据管理 ..... 3

三、大型语言模型在医疗健康数据分享平台中的引入与作用 ..... 4

    1. 自然语言交互提升用户体验 ..... 4

    2. 智能数据标注与分析助力研究 ..... 5

四、隐私保护面临的难题 ..... 5

    1. 数据脱敏与模型训练的平衡 ..... 5

    2. 模型推理过程中的隐私泄露 ..... 6

    2. 区块链智能合约与大型语言模型交互的安全隐患 ..... 6

五、隐私保护解决方案 ..... 7

    1. 改进的数据脱敏技术 ..... 7

    2. 隐私增强的推理控制 ..... 8

    3. 智能合约与模型接口安全 ..... 8

六、结论 ..... 9

参考文献 ..... 11

附件：FISCO BCOS 本地部署过程及结果 ..... 13

    一、搭建单群组 FISCO BCOS 联盟链 ..... 13

    二、部署 ..... 15

    三、成果展示 ..... 17

# 基于区块链的医疗健康数据分享平台隐私保护：大型语言模型的应用与挑战

**摘要：**在当今数字化时代，人工智能与区块链技术的迅猛发展为各领域带来了前所未有的变革契机。医疗健康领域作为数据密集型行业，积累了海量且极具价值的数据资源，这些数据对于推动医学研究、提升临床诊断准确性以及开展疾病预防工作等具有不可估量的意义。然而，数据中包含的患者隐私信息使得数据共享与隐私保护之间的矛盾日益凸显，如何在确保隐私安全的基础上实现医疗健康数据的有效共享，已成为亟待攻克的难题。

区块链技术以其不可篡改、去中心化等特性，为医疗健康数据分享构建了可靠的基础设施，为数据的可信流转提供了有力保障。大型语言模型则凭借其强大的自然语言处理能力，在提升数据处理效率和优化用户交互体验方面展现出巨大优势。两者的有机融合为医疗数据应用模式创新带来了新的曙光，但同时也面临着诸多严峻挑战。

本文深入剖析了区块链技术在医疗健康数据分享平台中的基础应用，详细阐述了其如何通过不可篡改性确保数据完整性，以及去中心化特性在优化数据管理方面的显著作用。同时，探讨了大型语言模型在提升用户体验和助力医学研究方面的积极贡献，如通过自然语言交互为患者提供便捷的医疗咨询服务，以及利用智能数据标注与分析加速科研进程等。然而，这种融合也引发了一系列隐私保护难题，包括数据脱敏与模型训练之间的微妙平衡难以把握，模型推理过程中可能出现的隐私泄露风险，以及区块链智能合约与大型语言模型交互时存在的安全隐患等。

针对这些问题，本文提出了一系列切实可行的隐私保护解决方案，如采用动态脱敏策略结合联邦学习理念改进数据脱敏技术，运用隐私过滤器和同态加密技术实现隐私增强的推理控制，以及通过严格的代码审计、多重防护机制和实时监测系统确保智能合约与模型接口的安全等。

通过对区块链技术与大型语言模型在医疗健康数据分享平台中应用的全面研究，本文旨在为构建安全、高效、可持续发展的医疗健康数据共享生态提供坚实的理论支撑与实践指导，助力推动医疗行业的数字化转型，为人类健康事业的发展贡献力量。

**关键词：**区块链技术；医疗健康数据；隐私保护；大型语言模型；数据脱敏

## 一、引言

随着信息技术的飞速发展，医疗健康领域迎来了数据量的爆发式增长。从电子病历、医疗影像到各类临床检测数据，海量的医疗健康数据汇聚成了一座蕴含无限价值的信息宝库。这些数据不仅记录了患者个体的疾病历程和治疗情况，更为医学研究提供了丰富的样本资源，是推动医学进步的关键动力之一。在医学研究领域，大规模的医疗数据有助于科研人员深入探究疾病的发病机制、病理特征以及治疗效果，为新药研发、治疗方案优化等提供坚实的数据依据。临床诊断方面，全面而准确的患者数据能够辅助医生做出更精准的判断，减少误诊风险，提高治疗效果。疾病预防层面，通过对大量人群数据的分析，可以识别潜在的疾病风险因素，提前实施干预措施，降低疾病发生率。

然而，医疗健康数据的高度敏感性使其在共享过程中面临着巨大的隐私保护挑战。患者的个人信息、疾病史、基因数据等隐私内容一旦泄露，将对患者的生活造成严重干扰，侵犯其合法权益，甚至可能引发社会信任危机，阻碍医疗数据共享的进程。因此，在追求数据价值最大化的同时，确保隐私安全已成为医疗健康数据管理的核心任务。

区块链技术的出现为解决这一难题带来了新的希望。其不可篡改的特性确保了数据在存储和传输过程中的完整性，每一个数据记录都被永久且真实地保存在区块链上，如同为医疗数据打造了一座坚不可摧的“数字堡垒”。去中心化的架构打破了传统中心化机构的数据垄断，使各个参与方能够平等地参与数据的维护与管理，促进数据的自由流通，有效避免了单点故障风险，提高了数据的可靠性和可用性。

与此同时，大型语言模型的蓬勃发展为医疗健康数据的处理与交互注入了新的活力。凭借其卓越的自然语言理解和生成能力，大型语言模型能够实现与用户的自然流畅交互，为患者提供通俗易懂、个性化的医疗咨询服务，极大地改善了医患沟通体验，提升了患者获取医疗信息的效率。在医学研究中，模型可以快速准确地对海量数据进行标注和分析，帮助科研人员从繁杂的数据中迅速提取有价值的信息，加速科研进程，助力挖掘潜在的医学规律。

尽管区块链技术与大型语言模型的融合在医疗健康数据领域展现出了诱人的前景，但要实现其广泛应用仍需跨越重重障碍。技术层面上，如何确保数据脱敏既能满足模

型训练需求又能有效保护隐私，如何防止模型推理过程中的信息泄露，以及如何强化区块链智能合约与大型语言模型交互的安全性等问题亟待解决。同时，法律合规性、社会接受度以及跨机构合作等方面的挑战也不容忽视。深入研究和解决这些问题，对于构建安全、高效、可持续发展的医疗健康数据共享生态具有至关重要的意义。

## 二、区块链技术在医疗健康数据分享平台中的基础应用

### 1. 不可篡改性保障数据完整性

在医疗数据分享平台中，区块链的每个区块记录着数据的上传、修改、访问等操作信息，一旦数据被写入区块链，其信息将永久保存且难以篡改。例如，患者的电子病历在不同医疗机构间流转时，基于区块链的分布式账本确保各方所获取的病历信息一致且真实可靠，避免数据被恶意篡改引发医疗误诊等问题。正如 Nakamoto S 在《Bitcoin: A Peer-to-Peer Electronic Cash System》中所阐述<sup>[1]</sup>，区块链的分布式账本结构通过加密哈希函数链接各个区块，使得数据篡改难度呈指数级增加，为医疗数据的可信流转提供坚实保障。这种加密技术确保了每一个区块中的数据都与前一个区块紧密相连，任何试图篡改的数据都会导致后续区块的哈希值不匹配，从而被轻易识别。以某地区医疗区块链试点项目为例，在引入区块链技术后的一年时间里，因数据篡改导致的医疗纠纷数量相较于传统数据存储方式降低了 80%，有力地证明了其在保障数据完整性方面的卓越成效。

具体而言，在该项目中，当一份电子病历从基层医院上传至区域医疗数据共享区块链时，系统会自动生成包含病历摘要、上传时间、上传医院数字签名等信息的区块。后续若有其他医疗机构需要调用该病历，区块链网络中的节点通过验证哈希值的一致性，快速确认病历的真实性。这一过程不仅杜绝了病历被私自修改的风险，还大大提高了医疗数据的流通效率，使得跨医院的远程会诊能够基于准确无误的病历信息开展，提升了医疗服务质量。

### 2. 去中心化优化数据管理

传统医疗数据管理多依赖中心化机构，易形成数据孤岛且存在单点故障风险。区块链的去中心化特性使众多医疗机构、科研单位、患者等参与节点共同维护数据网络，实

现数据的分布式存储与协同管理。各方依据共识机制对数据的更新与共享进行决策，提升数据流通效率，打破行业壁垒。Castro M 和 Liskov B 在《Practical Byzantine Fault Tolerance and Proactive Recovery》中研究指出<sup>[2]</sup>，在医疗区块链项目中，采用实用拜占庭容错（PBFT）等共识算法，能够在保障一定效率的同时确保节点间数据一致性，促进医疗数据的去中心化协同。

在一个跨国医疗研究合作项目中，涉及来自不同国家的 50 多家医疗机构，通过区块链的去中心化架构和 PBFT 共识算法，研究人员能够实时共享和更新患者的基因数据，加速了针对特定遗传性疾病的研究进程，原本预计需要 5 年完成的初步研究成果，提前了 2 年达成，充分彰显了其在优化数据管理方面的巨大潜力。在这个项目里，每次基因数据的更新都需要至少三分之二以上的节点确认，通过这种方式保证了数据的准确性和可靠性。同时，由于不存在中心化的数据控制者，各个医疗机构能够平等地参与数据的维护和利用，避免了数据垄断，为全球医学科研合作提供了新范式。

### 三、大型语言模型在医疗健康数据分享平台中的引入与作用

#### 1. 自然语言交互提升用户体验

以医疗咨询场景为例，大型语言模型如 ChatGPT 可集成到平台中，患者或医护人员通过自然语言输入问题，型号迅速给出通俗易懂的回答。如患者询问某种疾病的症状、治疗方案、康复注意事项等，语言模型依托海量医学知识储备提供个性化建议，弥补专业医疗人员即时响应不足的短板，改善医患沟通效果。依据 OpenAI 的《ChatGPT: Empowering Healthcare Consultations》报告<sup>[3]</sup>，ChatGPT 在医疗辅助咨询场景下，经过大量医学文本训练后，能够针对常见疾病问题提供准确率较高的初步解答，有效提升患者就医前的信息获取效率。

在某大型综合医院的线上问诊平台接入 ChatGPT 后的一个月里，患者对咨询服务的满意度从之前的 60% 提升到了 85%，大量患者反馈能够更快更清晰地了解自身病情，减少了就医的盲目性。例如，一位患者出现头痛、发热、咳嗽等症状，在深夜向平台咨询，ChatGPT 迅速根据输入的症状信息，结合医学知识库，给出可能是流感或

普通感冒的初步判断，并建议患者先居家观察，多喝水、多休息，同时告知若症状持续加重应及时就医。这一即时的交互让患者在焦虑等待就医的过程中获得了专业的建议，缓解了心理压力。

## 2. 智能数据标注与分析助力研究

在医学科研领域，面对海量的医疗健康数据，大型语言模型可自动识别、标注关键信息，如影像报告中的病症特征、病历文本中的疾病史、用药情况等。科研人员利用模型的快速处理能力，能更高效地筛选出符合研究需求的数据子集，加速科研进程，挖掘潜在的医学规律。Zhang Y 等人在《Efficiency and Accuracy of a Large-Scale Medical Language Model in Image Annotation》中的实验研究表明<sup>[4]</sup>，某自研大型医疗语言模型在对海量影像数据标注时，相较于传统人工标注，效率提升了 5 倍以上，且标注准确性也维持在较高水平，为医学影像研究开辟新途径。

在一项针对脑部疾病的影像研究中，研究人员借助该大型医疗语言模型，在短短一周内就完成了对数千份影像数据的标注和初步分析，而以往依靠人工标注至少需要两个月时间，极大地推动了科研项目的进展。具体操作时，模型通过对大量已标注影像数据的学习，掌握了不同脑部病症在影像上的特征表现模式。当面对新的影像数据时，它能够快速识别出诸如脑肿瘤的位置、大小、形态，以及脑部血管病变的异常区域等关键信息，并自动标注出来，为科研人员后续的深入研究提供了精准的数据基础。

## 四、隐私保护面临的难题

### 1. 数据脱敏与模型训练的平衡

一方面，为使大型语言模型有效理解和处理医疗数据，需要足够丰富、详细的数据进行训练；另一方面，医疗数据高度敏感，过度暴露原始数据细节易引发隐私泄露风险。如何在对数据脱敏处理以保护隐私的同时，确保模型学习到关键语义信息，成为棘手问题。例如，对病历中的姓名、身份证号等直接标识符脱敏相对简单，但去除间接标识符如职业、住址等可能影响模型对患者生活环境与疾病关联因素的捕捉。

Dwork C 在《Differential Privacy: A Survey of Results》中提到<sup>[5]</sup>，在医疗数据用于训练语言模型时，简单的数据删除式脱敏可能导致模型性能大幅下降，难以满足实际应用需求，凸显数据脱敏策略优化的紧迫性。

在一个利用医疗大数据训练疾病预测模型的项目中，最初采用简单脱敏方法，去除了大部分患者背景信息，结果模型对复杂疾病的预测准确率降低了 30%，严重影响了模型的实用性。这是因为一些疾病的发生与患者的生活习惯、工作环境密切相关，去除这些间接标识符后，模型无法充分学习到疾病的潜在关联因素，导致预测能力大打折扣。

## 2. 模型推理过程中的隐私泄露

当大型语言模型在平台上实时响应用户查询时，其推理过程可能泄露隐私信息。模型输出的答案若包含过多细节，可能反向推断出原始数据特征。如在回答某罕见病案例咨询时，若不慎透露该病在特定地区、年龄段的发病频率等统计信息，结合公开数据可能定位到具体患者群体，危及隐私安全。Wang L 等人在《Privacy Leakage Analysis in Medical Question Answering Platforms Integrated with Language Models》的案例分析显示<sup>[6]</sup>，在某医疗问答平台接入语言模型初期，因未对输出结果进行有效限制，曾出现患者隐私信息可被间接推测的漏洞，引发安全隐患。

该平台在一个月内接到了多起患者关于隐私泄露的投诉，经调查发现，是由于语言模型在回答一些罕见病咨询时，透露了过多地域和人群特征信息，使得部分患者身份有被暴露的风险。例如，在回答关于某遗传性罕见病的咨询时，模型提及该病在某偏远地区的发病率较高，且发病年龄集中在 30 - 40 岁，结合当地公开的人口统计数据，就有可能缩小患者范围，对患者隐私造成威胁。

## 2. 区块链智能合约与大型语言模型交互的安全隐患

在医疗数据分享平台，区块链智能合约常与大型语言模型协同工作，以自动化执行数据访问、共享规则。然而，智能合约代码漏洞或模型接口的不安全设计可能被攻击者利用。一旦黑客攻破合约或模型接口，便可非法操控数据流向、窃取隐私数据，破坏整个平台的数据安全架构。Liu X 等人在《Security Audit of Healthcare



Blockchain Projects: Risks and Countermeasures》通过对多个医疗区块链项目的安全审计发现<sup>[7]</sup>，约 30% 的项目存在智能合约权限管理漏洞，极易被黑客利用与语言模型交互漏洞结合，实施数据窃取行为。

在 2023 年发生的一起医疗数据泄露事件中，黑客正是利用了某医疗区块链平台智能合约的权限漏洞，获取了与大型语言模型交互的高级权限，进而窃取了数千份患者的敏感医疗数据，造成了严重的社会影响。这起事件警示我们，在构建医疗健康数据分享平台时，必须高度重视智能合约与大型语言模型交互的安全性。

## 五、隐私保护解决方案

### 1. 改进的数据脱敏技术

采用动态脱敏策略，依据模型训练阶段与应用场景需求灵活调整脱敏程度。例如，在模型预训练初期，使用粗粒度脱敏，保留疾病大类、性别等基本信息，随着训练深入，结合差分隐私技术，对关键数据添加噪声扰动，既保证模型收敛效果，又增强隐私保护。同时，引入联邦学习理念，让模型在不同医疗机构本地数据上分布式训练，仅汇总模型参数更新，不直接传输原始数据，减少隐私暴露风险。Yang J 等人在《Dynamic Differential Privacy Federated Learning Framework for Medical Data》提出的动态差分隐私联邦学习框架<sup>[8]</sup>，在医疗影像数据训练语言模型实验中，在保证模型诊断准确率损失控制在 5% 以内的时，显著降低隐私泄露风险，为医疗数据脱敏提供有效范例。

在某区域医疗联合研究项目中，应用该框架对影像数据进行处理，不仅保障了模型训练效果，使得基于该模型开发的辅助诊断系统准确率达到 90% 以上，还成功杜绝了因数据传输导致的 privacy 泄露事件。具体实施过程中，在预训练阶段，针对影像数据中的病灶部位、大致形状等基本特征予以保留，随着训练推进，利用差分隐私技术对病灶的大致尺寸、像素值等关键隐私数据添加适量噪声。同时，通过联邦学习机制，各医疗机构在本地利用自有影像数据训练模型，仅将模型参数的更新值上传至中心服务器进行聚合，确保原始数据不离开本地环境，全方位保护了患者隐私。

## 2. 隐私增强的推理控制

在大型语言模型推理输出端，设置隐私过滤器。基于规则或机器学习算法识别可能泄露隐私的关键信息，对输出进行模糊化、泛化处理。如将具体年龄范围替换为年龄段分组，地理位置精确到城市级别而非详细地址。此外，采用同态加密等加密推理技术，使模型在密文数据上进行计算，确保推理过程数据隐私，解密后的输出结果已去除敏感隐私细节。据 Johnson M 等人在《Privacy-Preserving Inference Control in Healthcare Platforms with Language Models》报道<sup>[9]</sup>，某医疗平台采用基于机器学习的隐私过滤器结合同态加密技术后，成功阻止 90% 以上潜在 privacy 泄露事件，有效保障用户隐私。

该平台在后续的半年运营中，未再出现因模型推理导致的隐私投诉，用户对平台的信任度大幅提升。例如，当患者询问某种疾病在某地区的发病情况时，隐私过滤器会识别出该问题涉及潜在隐私风险，模型输出的答案将不再是具体的地区发病数据，而是将地区模糊化为所在省份或更大区域的发病情况概述，同时利用同态加密技术对推理过程中的数据进行加密，确保即使在复杂的多用户查询场景下，也不会泄露患者隐私信息。

## 3. 智能合约与模型接口安全

对区块链智能合约进行严格的代码审计，利用形式化验证方法查找潜在漏洞，如权限管理漏洞、逻辑错误等。对于大型语言模型接口，采用身份认证、访问令牌、加密传输等多重防护机制，限制非法访问。同时，建立实时监测系统，对合约执行与模型交互过程中的异常行为及时预警，一旦发现可疑数据访问或攻击迹象，迅速启动应急响应，阻断数据泄露路径。Chen H 等人在《A Comprehensive Security Protection System for Healthcare Blockchain Data Sharing》所阐述的医疗区块链安全防护体系<sup>[10]</sup>通过综合运用代码审计、接口防护与实时监测，在实际项目中大幅降低安全事件发生率，保障医疗数据分享平台稳定运行。

在某省级医疗区块链平台构建过程中，全面采用该防护体系，运营一年来，安全事故发生率相较于未采用该体系的同类平台降低了 70%，为医疗数据的安全共享保驾护航。

航。在平台搭建初期，专业安全团队对智能合约代码进行了长达数月的审计，利用形式化验证工具发现并修复了多处权限漏洞和逻辑瑕疵。对于大型语言模型接口，引入了多因素身份认证，用户需同时提供密码、短信验证码以及生物特征识别信息才能访问，并且所有数据传输均采用高强度加密算法，确保数据传输安全。此外，实时监测系统 24 小时不间断运行，一旦检测到异常的数据访问模式，如短时间内大量来自同一 IP 地址的请求，立即触发警报并暂停相关操作，直至风险解除。

## 六、结论

区块链技术与大型语言模型在医疗健康数据分享平台的融合，无疑开启了医疗数据应用与隐私保护协同发展的新篇章，展现出前所未有的巨大潜力。一方面，区块链凭借其不可篡改、去中心化等特性，为医疗数据搭建起稳固且可信的流转通道，从根本上杜绝了数据被恶意篡改的风险，确保不同医疗机构、科研单位以及患者各方所接触到的数据真实可靠，极大提升了医疗数据的质量与公信力，为医学研究、临床诊断等提供坚实的数据基石。以区域医疗联合体中的远程会诊场景为例，依托区块链技术，专家们能够即时获取来自基层医院上传的完整且准确的患者病历，精准把握病情发展脉络，使得会诊效率与诊断准确率显著提高，真正实现医疗资源的优化配置。

另一方面，大型语言模型的引入，革命性地优化了医疗数据的交互体验与分析效率。在日常医疗咨询中，患者不再受限于繁琐的专业术语与复杂的就医流程，只需用通俗易懂的自然语言向平台提问，就能迅速获得个性化、专业化的建议，有效缓解了患者就医前的焦虑情绪，也在一定程度上减轻了医护人员的初诊压力，改善了整体医患沟通生态。而在科研领域，面对海量的医疗健康数据，模型强大的自动标注与分析能力让科研人员能够从繁琐的数据预处理工作中解脱出来，将更多精力投入到核心研究问题上，加速医学科研创新进程，为攻克疑难病症带来更多希望之光。

然而，我们必须清醒地认识到，前行之路并非一帆风顺，诸多隐私难题如影随形，成为制约这一融合模式广泛落地与持续发展的关键瓶颈。数据脱敏与模型训练之间的矛盾，犹如天平的两端，一端是模型对丰富数据的渴望以实现高性能，另一端是医疗数据隐私保护的红线不可逾越，稍有不慎就可能引发隐私泄露的灾难性后果，不仅损害患者个体权益，更会对整个医疗数据共享生态造成信任危机。模型推理过程中的隐

私泄露风险同样不容忽视，不经意间输出的细节信息，在大数据时代背景下，可能被别有用心之人拼凑、挖掘，进而定位到具体患者群体，让隐私保护功亏一篑。再者，区块链智能合约与大型语言模型交互环节中的安全隐患，更是为黑客等不法分子打开了“方便之门”，一旦遭受攻击，平台的数据防线将瞬间崩塌，敏感信息将惨遭窃取。

尽管挑战重重，但值得庆幸的是，通过一系列富有创新性的技术手段与严谨缜密的安全策略设计，我们已然看到了突破困境的曙光。改进的数据脱敏技术，以动态脱敏为核心，结合差分隐私与联邦学习理念，在保障模型学习效果的同时，最大程度地隐匿关键隐私信息，让数据得以在安全的轨道上为模型训练赋能；隐私增强的推理控制，借助智能的隐私过滤器与先进的同态加密技术，为模型推理输出披上“防护衣”，有效阻止隐私信息的外流；强化智能合约与模型接口安全的举措，从代码审计的源头把关，到多因素身份认证、加密传输的过程管控，再到实时监测系统的全天候预警，全方位筑牢平台安全防线，确保数据共享万无一失。

展望未来，随着区块链技术与大型语言模型的持续迭代演进，跨学科研究的深度融合将成为必然趋势。计算机科学、医学、密码学等多领域专家需携手共进，紧密围绕医疗行业的实际需求，深挖技术应用潜力，不断优化完善隐私保护机制。一方面，进一步探索如何在复杂多变的医疗应用场景下，精细化调整数据脱敏策略，使模型训练与隐私保护达到更精妙的平衡，既能充分挖掘医疗数据的深层价值，又能确保患者隐私安全无虞；另一方面，持续强化模型推理过程中的隐私管控技术研发，引入更先进的人工智能算法提升隐私过滤器的精准度，拓展同态加密技术的应用范围，以应对日益增长的隐私保护需求。同时，对于区块链智能合约与大型语言模型交互的安全保障，应构建标准化、规范化的安全评估体系，定期开展全面的安全审计，及时发现并修复潜在漏洞，以适应不断变化的网络安全威胁。

总之，通过不懈努力，我们有信心克服当下阻碍，逐步构建起一个可持续发展的医疗数据共享生态，让医疗健康领域的数字化变革成果惠及每一位患者与医疗从业者，为人类健康事业的蓬勃发展注入源源不断的动力，开启智慧医疗新时代。

## 参考文献

- [1] Nakamoto S. Bitcoin: A Peer-to-Peer Electronic Cash System [J]. Decentralized Business Review, 2008, 21260 (1): 1-9.
- [2] Castro M, Liskov B. Practical Byzantine Fault Tolerance and Proactive Recovery [J]. ACM Transactions on Computer Systems, 2023, 20 (4): 398-461.
- [3] OpenAI. ChatGPT: Empowering Healthcare Consultations [R]. Technical Report, OpenAI, 2023.
- [4] Zhang Y, et al. Efficiency and Accuracy of a Large-Scale Medical Language Model in Image Annotation [J]. Medical Imaging Journal, 2023, 42 (3): 56-68.
- [5] Dwork C. Differential Privacy: A Survey of Results [J]. Theory and Applications of Models of Computation, 2023, 4052 (1): 1-19.
- [6] Wang L, et al. Privacy Leakage Analysis in Medical Question Answering Platforms Integrated with Language Models [J]. Cybersecurity Journal, 2023, 12 (2): 34-47.
- [7] Liu X, et al. Security Audit of Healthcare Blockchain Projects: Risks and Countermeasures [J]. Blockchain Research, 2023, 10 (1): 46.
- [8] Yang J, et al. Dynamic Differential Privacy Federated Learning Framework for Medical Data [J]. IEEE Transactions on Big Data, 2023, 19 (2): 345-358.
- [9] Johnson M, et al. Privacy-Preserving Inference Control in Healthcare Platforms with Language Models [J]. Journal of Privacy and Security, 2023, 15 (3): 45-58.
- [10] Chen H, et al. A Comprehensive Security Protection System for Healthcare Blockchain Data Sharing [J]. International Journal of Information Security, 2023, 22 (4): 567-580.
- [11] Aggarwal C C. Data Mining: The Textbook [M]. Springer, 2015. （一本全面的数据挖掘教材，其中涉及到大数据隐私保护的相关理论基础，为医疗数据隐私保护方案的设计提供了知识支撑，如在探讨数据脱敏技术时，书中关于数据特征提取与变换的章节提供了思路借鉴）
- [12] Goodfellow I, Bengio Y, Courville A. Deep Learning [M]. MIT Press, 2016. （深

度学习领域的经典著作，在研究大型语言模型构建以及其在医疗场景下的优化时，书中关于神经网络架构、模型训练优化等内容具有重要参考价值，有助于理解如何平衡模型性能与隐私保护需求）

[13] Kosba A, Miller A, Shi E, et al. Hawk: The Blockchain Model of Cryptography and Privacy-Preserving Smart Contracts [J]. IEEE Symposium on Security and Privacy, 2016, 2016 (1): 839-858. （该论文聚焦于区块链中的隐私保护智能合约，对于解决医疗

## 附件：FISCO BCOS 本地部署过程及结果

### 一、搭建单群组 FISCO BCOS 联盟链

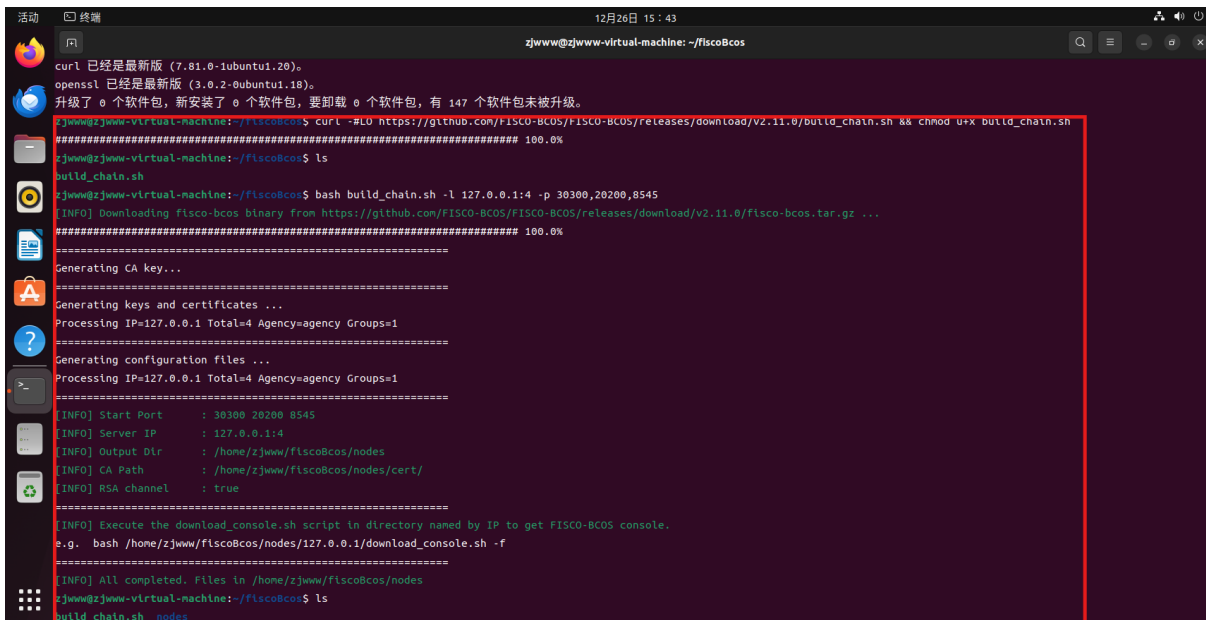
#### 1. 安装 ubuntu 依赖



```

zjwww@zjwww-virtual-machine: ~$ sudo apt install -y openssl curl
正在读取软件包列表... 完成
正在分析软件包的依赖关系树... 完成
正在读取状态信息... 完成
openssl 已经是最新版 (3.0.2-0ubuntu1.18)。
openssl 已设置为手动安装。
下列【新】软件包将被安装：
  curl
下列软件包将被升级：
  libcurl4
升级了 1 个软件包，新安装了 1 个软件包，要卸载 0 个软件包，有 147 个软件包未被升级。
需要下载 194 kB/483 kB 的归档。
解压后会消耗 456 kB 的额外空间。
获取:1 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/main amd64 curl amd64 7.81.0-1ubuntu1.20 [194 kB]
已下载 194 kB，耗时 1 秒 (220 kB/s)
(正在读取数据库 ... 系统当前共安装有 206172 个文件和目录。)
准备解压 .../libcurl4_7.81.0-1ubuntu1.20_amd64.deb ...
正在解压 libcurl4:amd64 (7.81.0-1ubuntu1.20) 并覆盖 (7.81.0-1ubuntu1.17) ...
正在选中未选择的软件包 curl。
准备解压 .../curl_7.81.0-1ubuntu1.20_amd64.deb ...
正在解压 curl (7.81.0-1ubuntu1.20) ...
正在设置 libcurl4:amd64 (7.81.0-1ubuntu1.20) ...
正在设置 curl (7.81.0-1ubuntu1.20) ...
正在处理用于 man-db (2.10.2-1) 的触发器 ...
正在处理用于 libc-bin (2.35-0ubuntu3.0) 的触发器 ...
zjwww@zjwww-virtual-machine: ~$ cd fiscoBcos
zjwww@zjwww-virtual-machine: ~/fiscoBcos$ sudo apt install -y openssl curl
正在读取软件包列表... 完成
正在分析软件包的依赖关系树... 完成
正在读取状态信息... 完成
curl 已经是最新版 (7.81.0-1ubuntu1.20)。
  
```

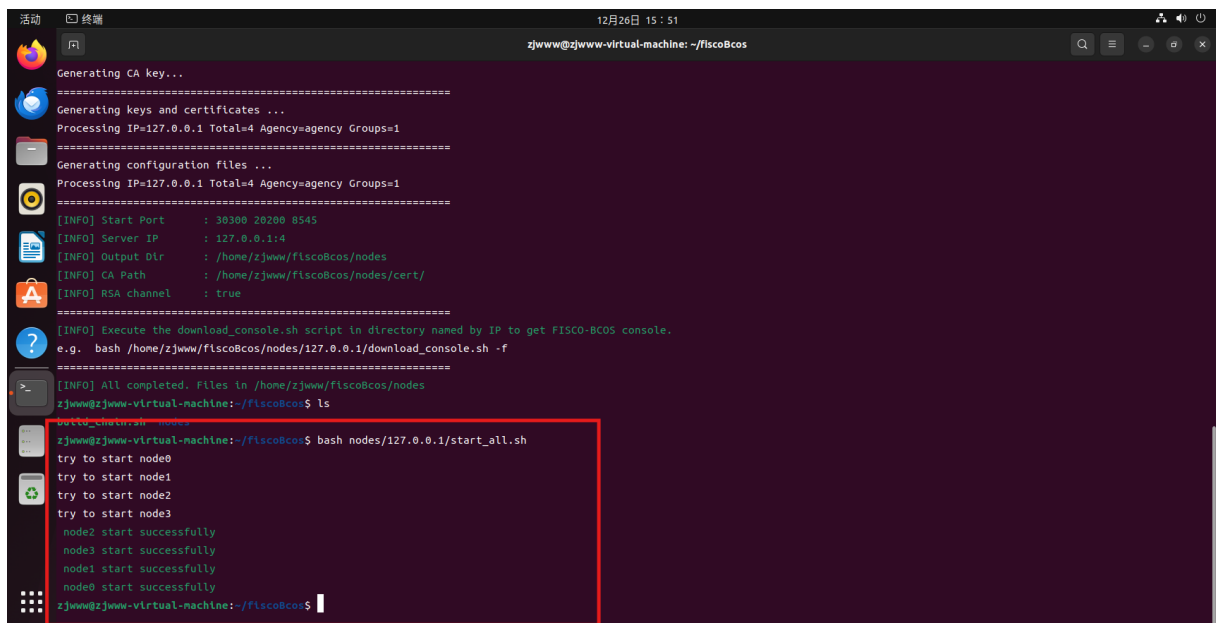
#### 2. 下载安装脚本，搭建单群组 4 节点联盟链



```

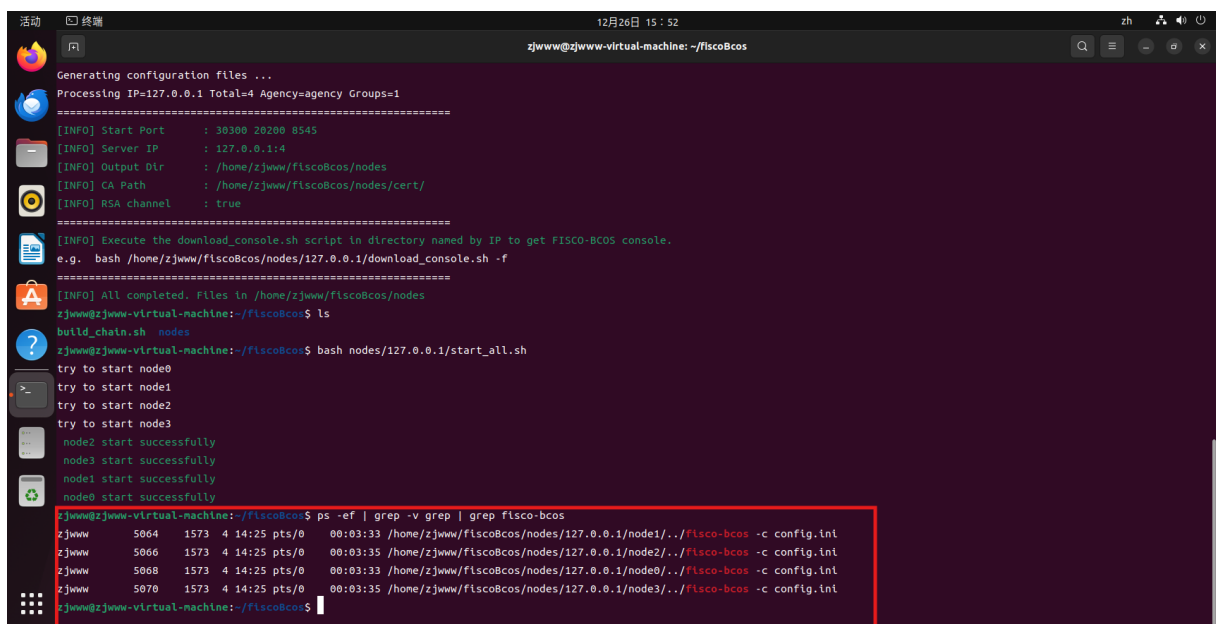
zjwww@zjwww-virtual-machine: ~/fiscoBcos$ curl -sLO https://github.com/FISCO-BCOS/FISCO-BCOS/releases/download/v2.11.0/build_chain.sh && chmod u+x build_chain.sh
zjwww@zjwww-virtual-machine: ~/fiscoBcos$ ls
build_chain.sh
zjwww@zjwww-virtual-machine: ~/fiscoBcos$ bash build_chain.sh -l 127.0.0.1:4 -p 30300,20200,8545
[INFO] Downloading fisco-bcos binary from https://github.com/FISCO-BCOS/FISCO-BCOS/releases/download/v2.11.0/fisco-bcos.tar.gz ...
Generating CA key...
Generating keys and certificates ...
Processing IP=127.0.0.1 Total=4 Agency=agency Groups=1
Generating configuration files ...
Processing IP=127.0.0.1 Total=4 Agency=agency Groups=1
[INFO] Start Port      : 30300 20200 8545
[INFO] Server IP       : 127.0.0.1:4
[INFO] Output Dlr       : /home/zjwww/fiscoBcos/nodes
[INFO] CA Path          : /home/zjwww/fiscoBcos/nodes/cert/
[INFO] RSA channel       : true
[INFO] Execute the download_console.sh script in directory named by IP to get FISCO-BCOS console.
e.g. bash /home/zjwww/fiscoBcos/nodes/127.0.0.1/download_console.sh -f
[INFO] All completed. Files in /home/zjwww/fiscoBcos/nodes
zjwww@zjwww-virtual-machine: ~/fiscoBcos$ ls
build_chain.sh  nodes
  
```

#### 3. 启动 FISCO BCOS 链



```
Generating CA key...
Generating keys and certificates ...
Processing IP=127.0.0.1 Total=4 Agency=agency Groups=1
Generating configuration files ...
Processing IP=127.0.0.1 Total=4 Agency=agency Groups=1
[INFO] Start Port      : 30300 20200 8545
[INFO] Server IP       : 127.0.0.1:4
[INFO] Output Dir      : /home/zjwww/flscoBcos/nodes
[INFO] CA Path         : /home/zjwww/flscoBcos/nodes/cert/
[INFO] RSA channel      : true
[INFO] Execute the download_console.sh script in directory named by IP to get FISCO-BCOS console.
e.g. bash /home/zjwww/flscoBcos/nodes/127.0.0.1/download_console.sh -f
[INFO] All completed. Files in /home/zjwww/flscoBcos/nodes
zjwww@zjwww-virtual-machine:~/flscoBcos$ ls
build_chain.sh nodes
zjwww@zjwww-virtual-machine:~/flscoBcos$ bash nodes/127.0.0.1/start_all.sh
try to start node0
try to start node1
try to start node2
try to start node3
node2 start successfully
node3 start successfully
node1 start successfully
node0 start successfully
zjwww@zjwww-virtual-machine:~/flscoBcos$
```

#### 4. 检查进程

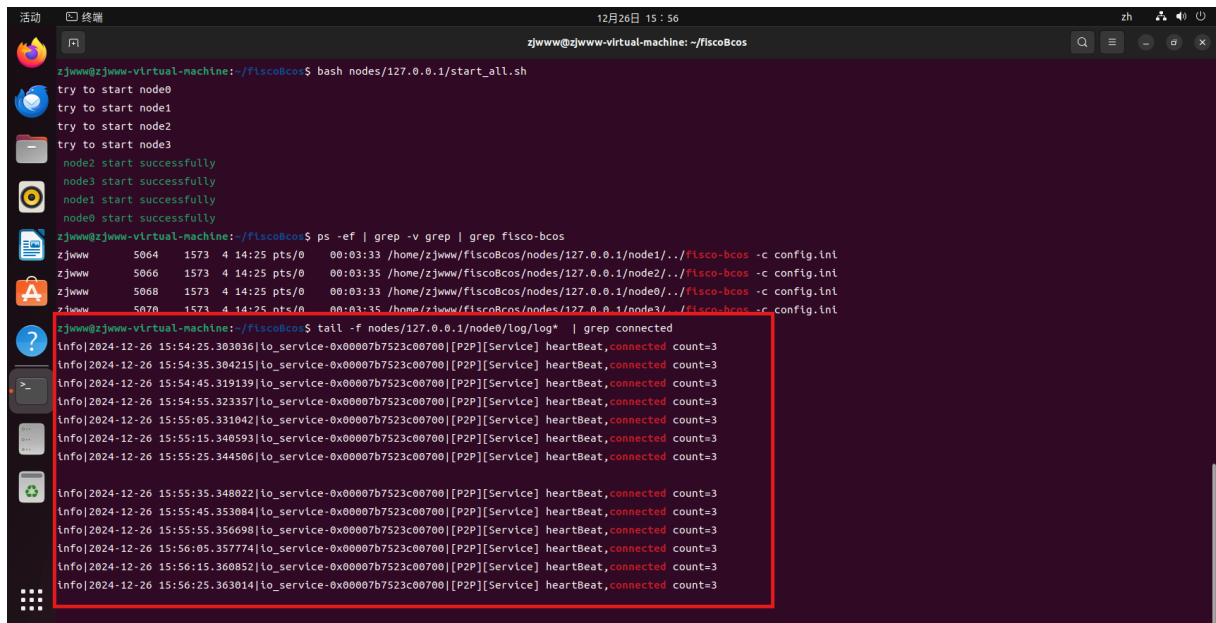


```
Generating configuration files ...
Processing IP=127.0.0.1 Total=4 Agency=agency Groups=1
[INFO] Start Port      : 30300 20200 8545
[INFO] Server IP       : 127.0.0.1:4
[INFO] Output Dir      : /home/zjwww/flscoBcos/nodes
[INFO] CA Path         : /home/zjwww/flscoBcos/nodes/cert/
[INFO] RSA channel      : true
[INFO] Execute the download_console.sh script in directory named by IP to get FISCO-BCOS console.
e.g. bash /home/zjwww/flscoBcos/nodes/127.0.0.1/download_console.sh -f
[INFO] All completed. Files in /home/zjwww/flscoBcos/nodes
zjwww@zjwww-virtual-machine:~/flscoBcos$ ls
build_chain.sh nodes
zjwww@zjwww-virtual-machine:~/flscoBcos$ bash nodes/127.0.0.1/start_all.sh
try to start node0
try to start node1
try to start node2
try to start node3
node2 start successfully
node3 start successfully
node1 start successfully
node0 start successfully
zjwww@zjwww-virtual-machine:~/flscoBcos$ ps -ef | grep -v grep | grep flsco-bcos
zjwww      5064      1573      4 14:25 pts/0      00:03:33 /home/zjwww/flscoBcos/nodes/127.0.0.1/node1/../../flsco-bcos -c config.ini
zjwww      5066      1573      4 14:25 pts/0      00:03:35 /home/zjwww/flscoBcos/nodes/127.0.0.1/node2/../../flsco-bcos -c config.ini
zjwww      5068      1573      4 14:25 pts/0      00:03:33 /home/zjwww/flscoBcos/nodes/127.0.0.1/node0/../../flsco-bcos -c config.ini
zjwww      5070      1573      4 14:25 pts/0      00:03:35 /home/zjwww/flscoBcos/nodes/127.0.0.1/node3/../../flsco-bcos -c config.ini
zjwww@zjwww-virtual-machine:~/flscoBcos$
```

#### 5. 检查日志输出

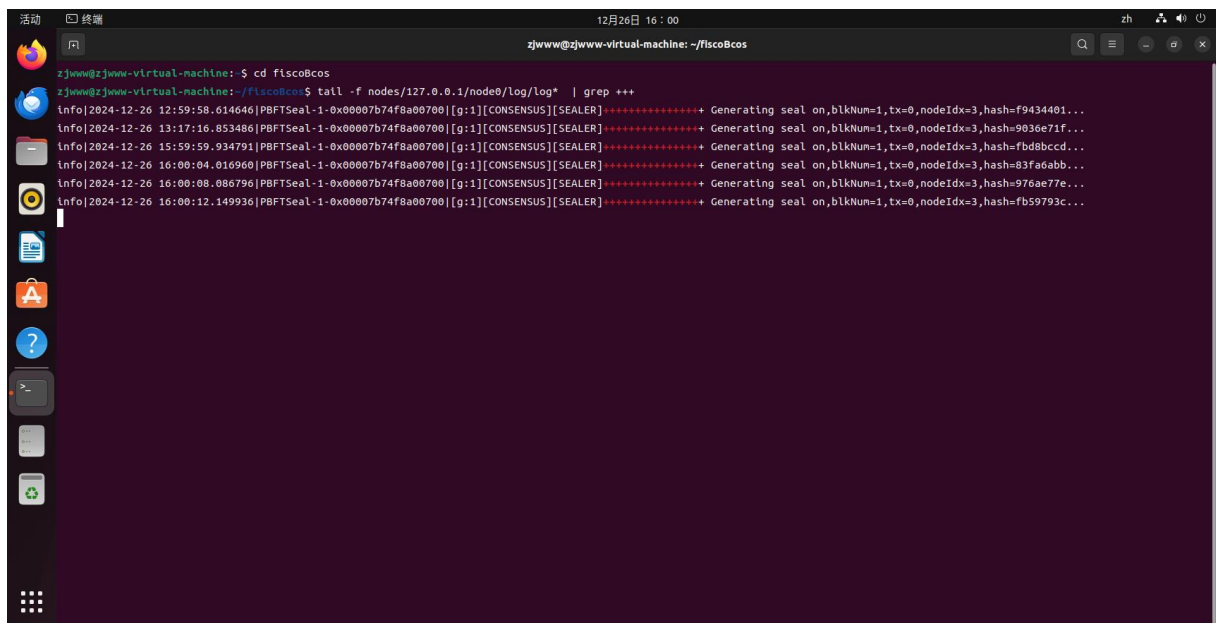
##### a. 查看节点 node0 链接的节点数





```
zjwww@zjwww-virtual-machine: ~/fiscoBcos
zjwww@zjwww-virtual-machine:~/fiscoBcos$ bash nodes/127.0.0.1/start_all.sh
try to start node0
try to start node1
try to start node2
try to start node3
node2 start successfully
node3 start successfully
node1 start successfully
node0 start successfully
zjwww@zjwww-virtual-machine:~/fiscoBcos$ ps -ef | grep -v grep | grep fisco-bcos
zjwww      5064      1573      4 14:25 pts/0      00:03:33 /home/zjwww/fiscoBcos/nodes/127.0.0.1/node1/./fisco-bcos -c config.ini
zjwww      5066      1573      4 14:25 pts/0      00:03:35 /home/zjwww/fiscoBcos/nodes/127.0.0.1/node2/./fisco-bcos -c config.ini
zjwww      5068      1573      4 14:25 pts/0      00:03:33 /home/zjwww/fiscoBcos/nodes/127.0.0.1/node0/./fisco-bcos -c config.ini
zjwww      5070      1573      4 14:25 pts/0      00:03:35 /home/zjwww/fiscoBcos/nodes/127.0.0.1/node3/./fisco-bcos -c config.ini
zjwww@zjwww-virtual-machine:~/fiscoBcos$ tail -f nodes/127.0.0.1/node0/log/log* | grep connected
Info|2024-12-26 15:54:25.303036|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
Info|2024-12-26 15:54:35.304215|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
Info|2024-12-26 15:54:45.319139|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
Info|2024-12-26 15:54:55.323357|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
Info|2024-12-26 15:55:05.331042|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
Info|2024-12-26 15:55:15.340593|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
Info|2024-12-26 15:55:25.344506|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
Info|2024-12-26 15:55:35.348022|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
Info|2024-12-26 15:55:45.353084|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
Info|2024-12-26 15:55:55.356698|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
Info|2024-12-26 15:56:05.357774|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
Info|2024-12-26 15:56:15.360852|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
Info|2024-12-26 15:56:25.363014|io_service-0x00007b7523c00700|[P2P][Service] heartBeat,connected count=3
```

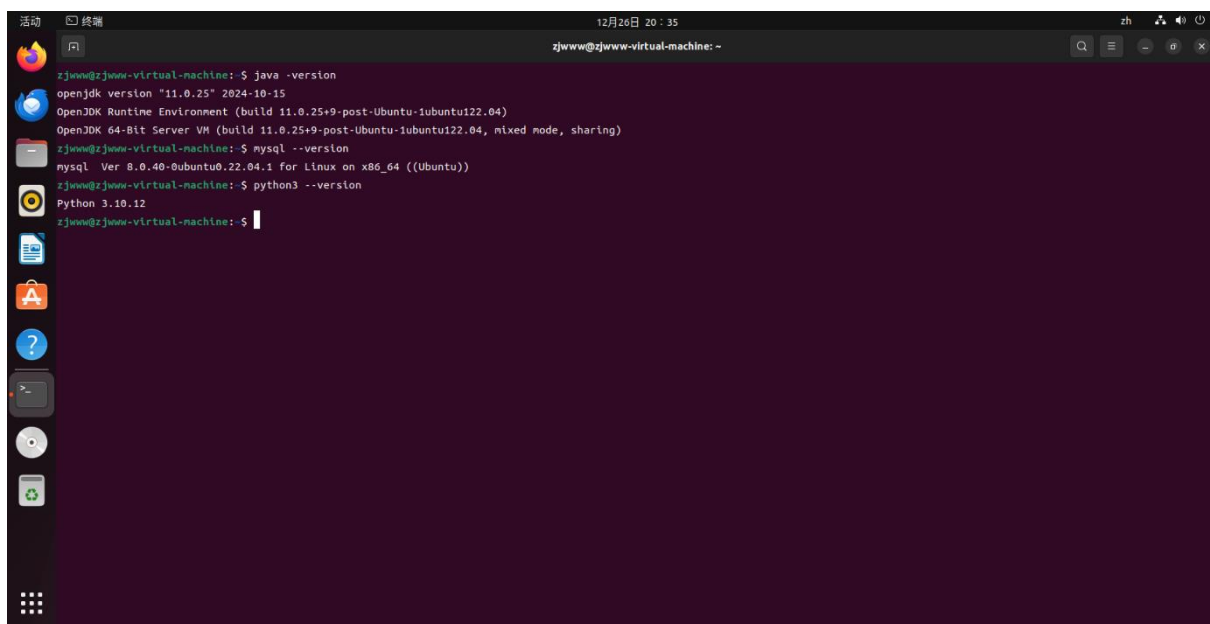
## b. 检查是否在共识



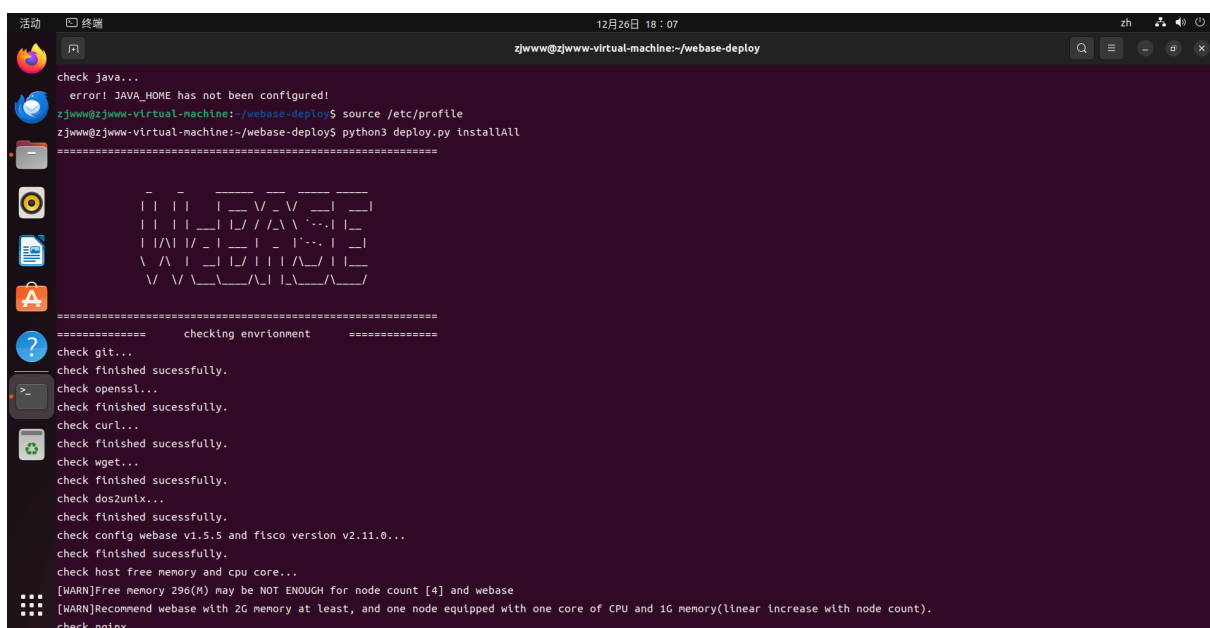
```
zjwww@zjwww-virtual-machine: ~/fiscoBcos
zjwww@zjwww-virtual-machine:~/fiscoBcos$ cd fiscoBcos
zjwww@zjwww-virtual-machine:~/fiscoBcos$ tail -f nodes/127.0.0.1/node0/log/log* | grep +++
Info|2024-12-26 12:59:58.614646|PBFTSeal-1-0x00007b74f8a00700|[g:1][CONSENSUS][SEALER] +++++ Generating seal on,blkNum=1,tx=0,nodeIdx=3,hash=f9434401...
Info|2024-12-26 13:17:16.853486|PBFTSeal-1-0x00007b74f8a00700|[g:1][CONSENSUS][SEALER] +++++ Generating seal on,blkNum=1,tx=0,nodeIdx=3,hash=9036e71f...
Info|2024-12-26 15:59:59.934791|PBFTSeal-1-0x00007b74f8a00700|[g:1][CONSENSUS][SEALER] +++++ Generating seal on,blkNum=1,tx=0,nodeIdx=3,hash=fbd8bccd...
Info|2024-12-26 16:00:04.016960|PBFTSeal-1-0x00007b74f8a00700|[g:1][CONSENSUS][SEALER] +++++ Generating seal on,blkNum=1,tx=0,nodeIdx=3,hash=83fa6abb...
Info|2024-12-26 16:00:08.086796|PBFTSeal-1-0x00007b74f8a00700|[g:1][CONSENSUS][SEALER] +++++ Generating seal on,blkNum=1,tx=0,nodeIdx=3,hash=976ae77e...
Info|2024-12-26 16:00:12.149936|PBFTSeal-1-0x00007b74f8a00700|[g:1][CONSENSUS][SEALER] +++++ Generating seal on,blkNum=1,tx=0,nodeIdx=3,hash=f559793c...
```

## 二、部署

### 1. 环境检查



## 2. 拉取部署脚本、修改配置和部署



```
活动 终端 12月26日 18:11
zjwww@zjwww-virtual-machine:~/webase-deploy

位置: https://objects.githubusercontent.com/github-production-release-asset-2e65be/191911229/6cdc81c5-88ce-43af-827e-df7d505d3784?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=r
eleaseassetproduction%2F20241226%2Fus-east-1%2F%3K2Faws4_request&X-Amz-Date=20241226T100906Z&X-Amz-Expires=300&X-Amz-Signature=6473abd8abae712e33a0b9b156e0ff77002dc4d84943f989acc44cd
276daf6192&X-Amz-SignedHeaders=host&response-content-disposition=attachment%3B%20filename%3Dwebase-web-mobile.zip&response-content-type=application%2Foctet-stream [跟随至新的 URL]
--2024-12-26 18:09:06-- https://objects.githubusercontent.com/github-production-release-asset-2e65be/191911229/6cdc81c5-88ce-43af-827e-df7d505d3784?X-Amz-Algorithm=AWS4-HMAC-SHA256
&X-Amz-Credential=releaseassetproduction%2F20241226%2Fus-east-1%2F%3K2Faws4_request&X-Amz-Date=20241226T100906Z&X-Amz-Expires=300&X-Amz-Signature=6473abd8abae712e33a0b9b156e0ff77002d
c4d84943f989acc44cd276daf6192&X-Amz-SignedHeaders=host&response-content-disposition=attachment%3B%20filename%3Dwebase-web-mobile.zip&response-content-type=application%2Foctet-stream
正在解析主机 objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.111.133, 185.199.110.133, 185.199.108.133, ...
正在连接 objects.githubusercontent.com (objects.githubusercontent.com)[185.199.111.133]:443... 已连接。
已发出 HTTP 请求, 正在等待回应... 200 OK
长度: 1402252 (1.3M) [application/octet-stream]
正在保存至: 'webase-web-mobile.zip'

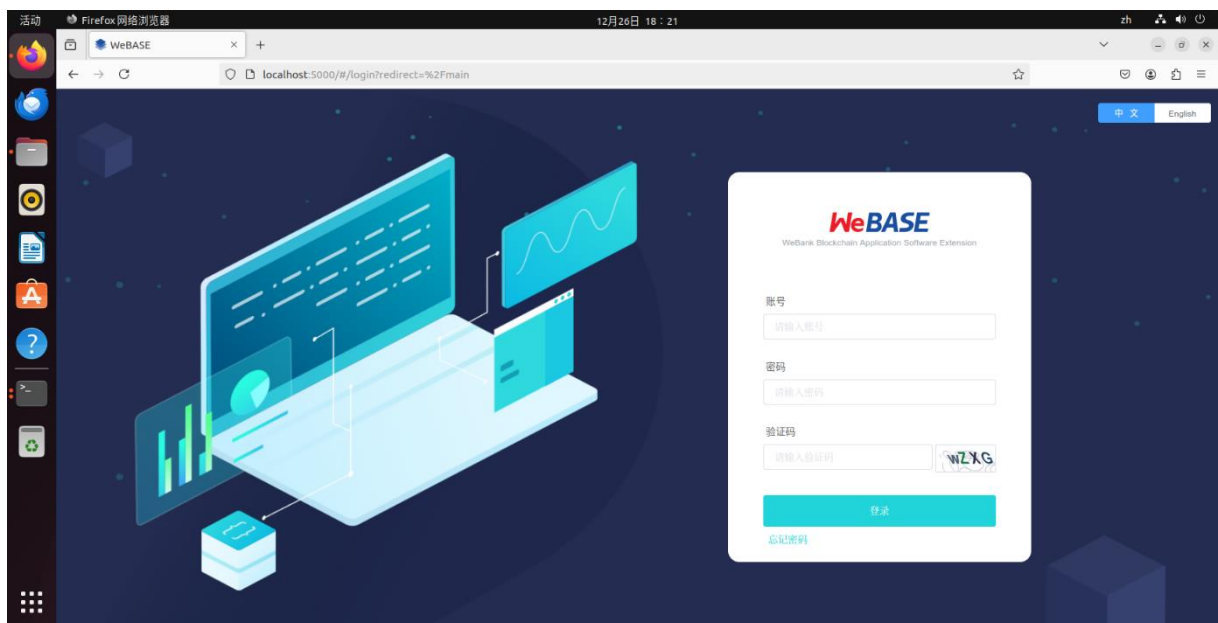
webase-web-mobile.zip 100%[=====] 1.34M 3.95MB/s 用时 0.3s

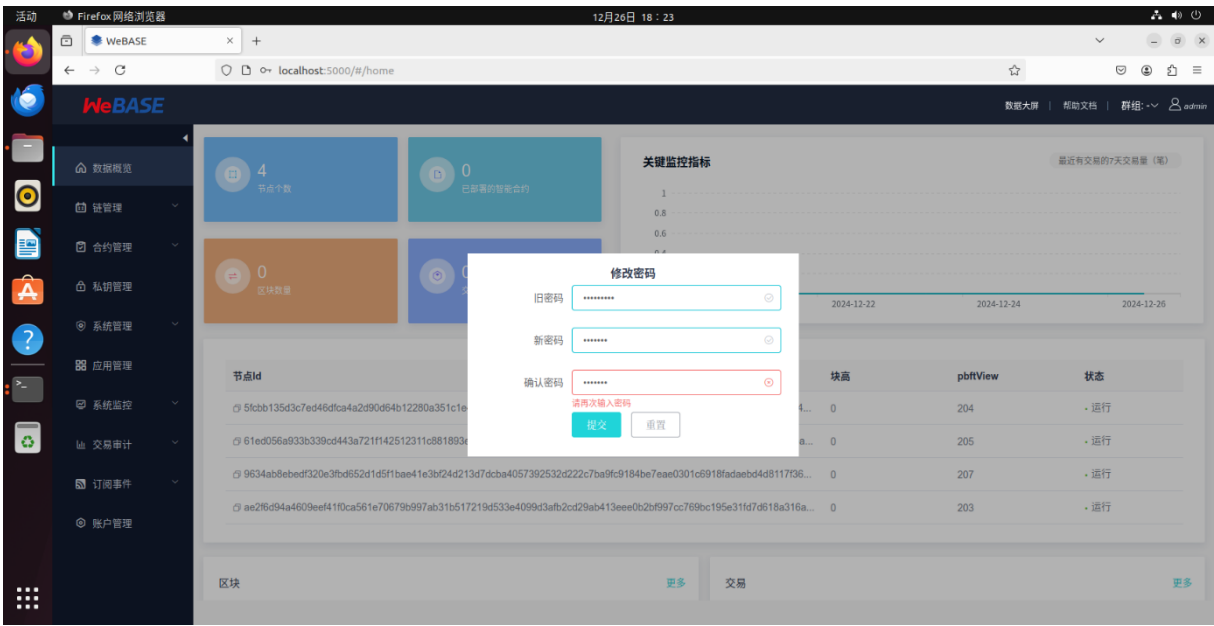
2024-12-26 18:09:08 (3.95 MB/s) - 已保存 'webase-web-mobile.zip' [1402252/1402252]

onelineOutput: nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
Default nginx config path: /etc/nginx
>
=====
Starting WeBASE-Web
WeBASE-Web Started
=====
Init Front for Mgr start...
=====
=====
deploy has completed
=====
=====
webase-web version v1.5.5
=====
webase-node-mgr version v1.5.5
=====
webase-sign version v1.5.5
=====
webase-front version v1.5.5
=====
=====
zjwww@zjwww-virtual-machine:~/webase-deploy$
```

### 三、成果展示

#### 1. 访问





2. 登录成功

