文章编号:1672-3961(2024)06-0057-15 DOI:10.6040/j.issn.1672-3961.0.2023.003

基于智能合约的电商社区式问答服务平台设计

李港龙,林培光*,李金玉,王倩
(山东财经大学计算机科学与技术学院, 山东 济南 250014)

摘要:传统社区式问答平台中问答数据不仅透明度低而且容易被内部员工或外部攻击者篡改,导致客户无法信任问答平台信息的真实性,同时,因激励机制的缺乏也让已购买客户没有动力回答其他消费者的提问。为了改善现状,提出并设计一种基于区块链智能合约的社区式问答服务平台,此平台引入了随机邀请、积分激励和声誉激励机制。该平台主要由 6 个智能合约协同工作,完成用户注册、产品上架以及交易、授权、提问、回答和评分等功能,同时在链下使用基于 Minhash 和 Jaccard 算法的文本相似度检测方案,提升用户回答的多样性。在以太坊私链环境中对 Solidity 语言编写的智能合约进行测试和分析,试验结果证明了代码的正确性和方案的有效性。

关键词:区块链;智能合约;电子商务;客户服务;社区问答
中图分类号:TP393 **文献标志码**:A
引用格式:李港龙,林培光,李金玉,等.基于智能合约的电商社区式问答服务平台设计[J].山东大学学报(工学版),2024,54(6):57-71.
LI Ganglong, LIN Peiguang, LI Jinyu, et al. Design of E-commerce community Q&A service platform based on smart contract[J]. Journal of Shandong University (Engineering Science), 2024, 54(6):57-71.

Design of E-commerce community Q&A service platform based on smart contract

LI Ganglong, LIN Peiguang*, LI Jinyu, WANG Qian
(School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan 250014, Shandong, China)

Abstract: In the traditional community-based Q&A platform, its Q&A data was not only low in transparency, but also vulnerable to tampering by internal employees or external attackers, resulting in customers not being able to trust the authenticity of information such as the identity of the respondent who had purchased and the ranking of the answers, while at the same time, the lack of incentive mechanism also made customers not motivated to answer other consumers' questions. In order to improve the current situation, a community-based Q&A service platform was proposed, which introducedthe mechanism of random invitation, point incentive and reputation incentive. The platform was consisted of six smart contracts working together to complete user registration, product shelving, and functions such as trading, authorization, questioning, answering and scoring, while the text similarity detection scheme based on Minhash and Jaccard algorithm was used under the chain to improve the diversity of user answers. The smart contract was tested for functionality and performance analysis in an Ethereum private chain environment, and the experimental results proved the correctness of the code and the effectiveness of the scheme.

Keywords: blockchain; smart contract; electronic commerce; customer service; community Q&A

0 引言

随着互联网技术的发展,网上购物越来越受到消费者的欢迎,据中国互联网络信息中心发布的统计报告,截至 2020 年 12 月,中国网络购物用户达 7.82亿,占到网民总体的 79.1%。随着电子商务的快速发展和客户需求的增加,市场对于高质量客户服务的期望也在不断提升。客服是各行各业最通用的职能岗位,市场需求整体较大,但其在组织架构中往往处于下游位置,企业为了控制运营成本和用工风险往往无法提供经过培训的、足量的客服人

收稿日期:2023-01-02
第一作者简介:李港龙(1996—),男,山东枣庄人,硕士研究生,主要研究方向为区块链应用。E-mail:794156416@qq.com
*通信作者简介:林培光(1978—),男,山东烟台人,教授,硕士生导师,博士,主要研究方向为机器学习。E-mail:linpg@sdufe.edu.cn

员,因此,大多数企业倾向以外包的形式让其他公司提供客户服务,或者结合采用智能客服的方式缓解人工客服数量短缺的现状[1]。但是,由于企业文化差异和专业水平不足,在尚未完全了解所售产品和服务的情况下,外包客服往往无法给客户提供良好的咨询体验,进而给企业带来负面影响[2]。另外,基于人工智能技术的智能客服需要训练大量问答数据,这是一个耗时、知识密集和劳动密集的过程,而且仅能在重复性和标准化的咨询问答中保证正确性[3]。问答平台是电商企业推出的一种以社区式在线问答形式存在的新型客户服务,例如淘宝问大家、京东问答等,在此类平台中提问者和回答者全部为客户,通过建立买家互助和经验共享的服务社区,挖掘了购买客户之间有价值的沟通。相较于传统的在线评论模块,该平台具有双向交互和易于获取聚焦信息的特点[4]。但是,目前可用的问答平台全部基于中心化控制模型,它们由一个服务提供商控制,该服务提供商负责对客户的提问、回答和管理、存储相关反馈信息,并且基于此在回答模块中进行排名,这些数据都比较容易被内部的恶意员工或者外部的攻击者篡改[5-7]。问答数据作为客户未来购买意愿和解决方案的重要参考依据,具有一定的经济价值,因此,好评数较多或处于列表中较前位置的回答将引导消费者采取与相应内容导向一致的行为。当前中心化问答平台的主要问题是,无法向客户证明其使用的邀请机制、问答策略、排名算法和查重算法与提供的描述一致,由于缺乏透明度,消费者很难信任平台中的回答来自于已购买客户。另外,现有的问答平台中缺乏激励已购买客户积极回答问题的机制,存在咨询问题被长时间搁置而无人回复以及回复质量偏低的情况,消费者仍然无法获得良好的使用体验。区块链是一种分布式账本技术,通过结合 P2P 共识协议、密码学和时间戳等技术,拥有去中心化、不可篡改、公开透明和可追溯等特性[8]。智能合约是区块链最重要的组件之一,是一个事件驱动并且由矿工执行的有状态的计算机程序,具有不涉及第三方执行功能的特性,可以将区块链上的数据、交易和各种有形或无形资产构建为可编程的智能资产[9-10]。区块链和智能合约通过构建和执行公开透明的交易规则,有助于开发一个安全互信的交易环境,已在金融[11]、物联网[12]、医疗[13]和教育[14]等领域得到大量研究和应用。本研究主要利用区块链和智能合约技术,通过构建一个去中心化、公开透明和可信的问答与评分框架解决传统中心化社区式问答平台中存在的问题。首先,该框架设计了一种基于声誉度的随机邀请机制,通过这种简单的授权模块控制客户的回答权限,提高回答的可信度。其次,为了鼓励客户积极回答问题,向消费者提供及时的服务,引入了基于通缩和折半经济模型的积分激励机制,客户回答问题后都可以获得卖方提供的积分奖励,再次购买时使用积分将获得相应的优惠减免。然后,评分功能是一项用来鼓励客户提升回答质量的机制,也是用来筛选优质回答的依据,已购买客户可以根据自身使用体验对回答进行评分,当其高于指定值时回答者将获得额外的积分和声誉奖励,同时,评分者也会有少量积分奖励;评分计算功能依据评分者分数和声誉等信息获得回答的均分,用于在链下对回答可靠度或质量进行排名。最后,本研究引入了基于 Minhash 和 Jaccard 算法的文本相似度检测方案,通过在链下检测和链上保存 Minhash 签名矩阵,在提高回答多样性的同时也保证了可验证性。为了证明提出平台的可行性,在 Ganache 提供的以太坊私链环境中对智能合约进行了测试和分析。

1 相关工作

优秀的智能客服系统依赖于经过大量客户问答数据训练的机器学习模型,文献[15]提出了一个基于区块链和 AutoML 的客服平台,通过构建一个共享和可信的数据交易环境,实现企业间相互协作,帮助中小企业能够使用足够的脱敏后的共享数据训练模型,改善自动化客户服务。为了帮助消费者迅速确定产品或服务的优缺点,文献[16]设计了一种结合数据挖掘和分类方法的改进算法(包括奇异值分解、熵测度和双线性相似度算法),通过分析评论、问答和评分数据,有效对产品和回答进行排名;文献[17]提出一种基于区块链的声誉系统,在电子商务场景下,买方的荣誉度由卖方对买方的评分决定,而买方对商品的最终评分由实际给出的分数和荣誉度共同决定,同时系统内引入了代币激励机制鼓励客户的评分行为;文献[18]提出一个基于区块链技术的通用去中心化评分框架,通过透明的评分收集和分数计算过程解决传统推荐系统中服务商对数据绝对控制和客户无法信任排名结果的问题,另外,框架内引入了基于声誉的通证激励机制,通过“奖励与技能”、“技能与参与度”相绑定的策略,增加已购买客户参与评分的积极性。淘宝“问大家”、京东“京东问答”和亚马逊“买家

问题和答案”等都是中心化电商企业提出并应用的社区式问答平台,一般在产品的兴趣客户提问后,通过算法随机邀请若干已购买过产品的消费者回答问题。

目前与客服(包括社区式问答平台)相关的工作主要是在假设问答数据来源可信和质量优秀的基础上如何提升客户使用体验,并没有从根本上解决整个过程中不透明和缺乏激励的问题。区块链与电子商务相结合的研究则主要集中在利用评分和信誉系统解决评论的可信度上,并没有设计与激励相对应的惩罚措施,所以也没有解决社区式问答中存在的问题。

2　平台框架

本研究设计的社区式问答平台在支持智能合约功能的区块链网络上构建,它由多个组件协同工作满足平台所需功能,为多方实体提供一个公开、可信、易用和互利的交易环境,如图 1 所示。前端和中心化数据库是为了提供可视化、快速检索和查重检测等功能,与优化算法一样都是为了提升客户的使用体验,因为涉及到链下,所以并非本研究讨论的重点。

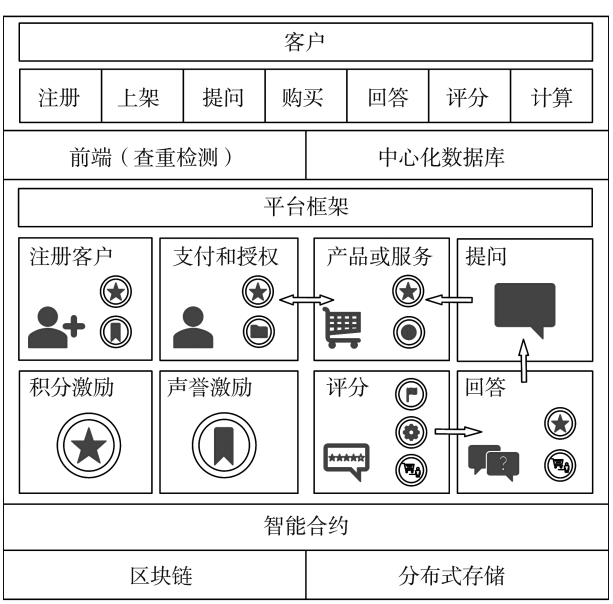

图 1　平台框架

Fig.1　Platform framework

2.1　平台组件

(1)产品或服务。由企业(卖方)提供的产品或服务的数字化表示,其拥有一组相对应的特征属性,例如名称、价格和积分发放规则等。

(2)授权模块。向已购买产品或服务的客户授予回答问题和评分回答的权力,在问题创建和支付成功后合约自动完成。

(3)积分模块。制定平台内的经济奖励机制,客户回答问题、评分回答和给出的回答获得高分(大于 7 分)都将获得一定数量的积分。因为卖方无法控制平台全局积分的发放和使用策略,因此采用产品本地积分,具体规则由卖方在合约内创建产品或服务时设定。客户再次购买时可使用积分享受减免优惠。

(4)声誉模块。声誉度是客户在问答平台中作出贡献的凭证,在回答被提问者给予高评分(大于 7 分)后获得。为了鼓励客户发布高质量回答,积分模块采用经济激励的方式,即回答每获得一个高评分,回答者都会有额外的积分奖励,而声誉模块采用声誉激励的方式,可以在前端根据分数授予差异化标识。

(5)提问模块。允许任何实体向特定产品或服务提出问题,在提问者购买成功后,获得提问者高评分(大于 7 分)的回答将认定为“可信回答”。

(6)回答模块。用于回答问题,仅能由已购买产品或服务且满足特定条件(如声誉度阈值和受邀资格)的客户使用。

(7)评分模块。用于给回答评分,仅能由已购买特定产品或服务的客户使用,其中的评分函数沿用文献[19]中的可插拔设计,支持一组不同的评分策略。

(8)存储模块。用于存储客户的问答信息,由分布式存储星际文件系统 IPFS 提供服务,仅将文件存储后的内容标识符 CID 信息上传至智能合约,解决区块链存储性能问题。

2.2　平台实体

(1)电商平台。部署智能合约,为企业和客户提供产品(或服务)上架、交易和问答功能的平台方。

(2)企业(卖方)。在平台内构建产品或服务的数字化形式,对社区式问答平台有需求的卖方。

(3)提问者。对特定产品或服务感兴趣,向社区提出问题的客户。

(4)回答者。已购买过特定产品或服务且满足特定条件,基于各种因素回答社区内问题的客户。

(5)评分者。已购买过特定产品或服务,基于各种因素对社区内回答进行评分的客户。

2.3　随机邀请机制

2.3.1　积分与声誉度

为了进一步提高回答的可信度,避免恶意刷评的情况发生,本研究提出一种可验证的、随机的邀

请回答机制。在实际应用中,该机制应当保证一定程度的容错率和较低的处理周期,回答的容错率可以通过随机邀请多位回答者获得提高,但在提高回答及时性上,仍需要一种新的策略。

通缩(发行总量有限)和折半(每隔一段时间挖矿奖励减半)是比特币等许多主流公有链采用的经济模型,通过经济激励的方式,鼓励用户在前期便积极参与到项目之中。为了让拥有回答资格的用户能够尽快参与回答,降低等待时间,本研究在发放积分时采用同样的策略。

假设卖方将每个问题可发放的积分总量设置为41个,其中20个用于奖励回答,另外20个用于奖励评分回答和优质回答,剩下的1个预留给提问者作为购买后参与评分(只在分数大于7时)的奖励,这是为了尽可能标记出具有参考意义、提问者认定的"可信回答"。每拥有一个回答后,奖励积分减半,回答者最终可获得的积分如表1所示,另外,评分回答和优质回答(获得分数大于7时)每次可获得1个积分。

表1 回答积分激励

Table 1 Answer points incentive

回答顺序	1	2	3	4	5	6
奖励积分	10	5	2	1	1	1

由表1可知,前6个回答都有奖励,只要有一个是正确的,该问题便成功解决,而后续已购买客户的评分和提问者购买后的反馈也可以将更合适的回答标记出来,以供后续拥有类似问题的客户参考。

另外,为了惩罚恶意回答者,当回答获得小于3的低分时(获得低分的次数大于10时,该回答会被删除),相应的回答者会被扣除1个单位的声誉度,若其声誉度当前为0,本次惩罚将被记录,下次获得声誉度时会被优先用来消除记录。虽然拥有惩罚记录的用户声誉度仍然标识为0,但在卖方普遍设置声誉度阈值的情况下,他们被选为回答者的概率已经变得很低,获得积分收益也将变得更加困难,以此来激励回答者作出诚实的反馈。同时,为了增加恶意用户通过联合作恶删除正常回答的难度,对用户给予低分(小于3)的频率进行限制,例如每3 d只能给予1次。

2.3.2 可验证随机邀请算法

每个产品或服务都维护一份已购买用户名单,在向用户发布邀请即确定回答资格时,可以设置声誉度阈值和邀请数目2个条件。设置的声誉度越高,意味着拥有资格的用户诚实回答的可能性越高,但可邀请的范围和回答者身份的不确定性随之降低。在2.3.1节设计的激励机制下,虽然只有前6个回答者可以获得积分奖励,但卖方仍可将邀请数目设定为大于6,因为后面的回答者可以通过高评分赢取优质回答的奖励。

区块哈希是指对区块头进行哈希计算获得的32字节哈希值,在比特币等公有链中,只有区块哈希符合要求的区块才能拥有获得共识的资格,所以它在上链后便具有确定性,很难更改[20]。文献[21]提出了一种无偏随机邀请算法,即根据区块哈希和用户声誉度从在线用户群体中确定参与用户,公式为:

$$R>0 \;\&\&\; A_i=H\%L,\ i=1,2,\cdots,k,$$

式中,R为用户的声誉值,A_i表示拥有资格的回答者在名单中的索引,H为前i个区块的哈希之和,L为已购买用户的总数,k为邀请数目。

但是,当调用该功能的多个交易在同一个区块内被打包时,将导致同一实体下不同子场景所邀请的用户相同,不利于系统的稳定,因此本研究设计了区块哈希和问题合约地址共同作为随机种子随机选择回答者的方案,因为不同的问题合约都拥有独一无二的地址,所以当已购买用户数足量时,可以防止同一个产品的不同问题在同一个区块内发出后,邀请的都是同样用户群体的不利情况发生。随机选择回答者公式为:

$$R>r \;\&\&\; A_i=(H_i+C)\%L,\ i=1,2,\cdots,k,$$

式中,H_i表示前i个区块的哈希,C表示相应问题合约的地址,r为企业设定的声誉度阈值。若A_i用户声誉度较低或已被邀请时,则自动选择下一个用户A_i+1,直到符合要求。

当企业设定的声誉度阈值越高、给予的积分奖励(参会评分的用户)越多时,其高分回答就越可信。另外,在随机邀请回答者时,本研究针对可能出现的2种情况分别设计了不同的应对方案:(1)当前符合要求(超过声誉度阈值)的已购买用户总数小于设定的邀请数目时,将信用度阈值不断降低,直到符合要求的用户数满足邀请数目;(2)当前已购买用户总数小于设定的邀请数目时,授予所有已购买用户回答权限,且在已邀请数小于设定的值或提问者购买后未评分情况下,所有新购买的用户都自动获得回答权限。

2.4 查重检测

在传统问答中,为了提高回答多样性,避免用户抄袭别人的回答,一般都添加了查重检测的功

能。现有的文本相似度检测算法主要分为传统和基于深度学习的方案,在搜索系统和查重检测中都有大量应用。其中,传统的方法包括 Simhah[21]、Minhash[22]和 TF-IDF[23]等模型,基于深度学习的方法包括 SimCSE[24]、Word2vec[25] 和 BERT[26] 等模型。

一般来说,基于深度学习的方法具有更高的精度,并且能够应用于形近字、拆分字和音形码等特殊场景,但相较于传统方法,它需要大量的数据训练模型,成本较高,而且无法像传统方法一样在简易的环境中快速运行,例如集成在前端中,使用门槛也较高[27]。在引言中提到了传统平台存在的查重算法不透明的现状,包括整个算法过程的不透明(平台出于商业考虑未公开所使用模型的全部信息)和用户无法对结果进行低成本、快速复现或检验(用户无法提供运行模型所需要的环境和配置)。所以,本研究选择使用传统的文本检测算法,虽然无法识别部分恶意内容,但结合 2.3 节中描述的惩罚机制,仍可以实现预期目标。

Minhash 算法是局部敏感哈希算法的一种,它的思想是对一个列向量按行进行随机排列,重排后的第一个非零元素就是最小哈希值,理论上说,如果两个文本完全相同,那么无论如何重排,它们的 Minhash 值都应当是一样的。和其他传统算法相比,局部敏感哈希算法在空间消耗和查询处理效率上具有优势,能够快速估计两个文本文档之间的相似性[28],而 Minhash 相较于 Simhash 更适合对短文本进行处理。

Jaccard 相似度[29]是一种用来比较有限样本集之间相似性与差异性的方法,经常与 Minhash 算法一起使用,当 Jaccard 相似度越大(最大为 1),表示样本相似度越高。假设 X 是一个集合,则 A 和 B 是 X 的子集,Jaccard 相似度

$$J(A,B)=\frac{|A\cap B|}{|A\cup B|}=\frac{|A\cap B|}{|A|+|B|-|A\cap B|}。$$

基于 Minhash 和 Jaccard 算法的文本相似度计算主要有 3 个步骤。

(1) 将文本表示为集合。

扫描文本,依次获得除空格和其他标点符号外的 n 个字符,并将每个字符转换为 32 位无符号整数 u_i,集合 $A=(u_1,u_2,\cdots,u_n)$。

(2) 生成对应的 Minhash 签名。

当向量矩阵非常大时,对其进行随机行排列是一件非常耗时的工作,目前主要通过使用随机哈希函数将 32 位整数转换为另一个 32 位整数,以此模拟打乱的效果。

定义了 7 个哈希函数,公式为:

$$H_j(x)=(a_jx+b_j)\%(P+j),\ j=1,2,\cdots,7,\quad (1)$$

式中:a_j、b_j 是随机选择的,a_j、$b_j\in\mathbf{Z}$,$\mathbf{Z}$ 是整数集合且 a、b 取值为小于 x 的最大值($2^{32}-1$);P 是比 x 最大值稍大的素数,该值可以确保哈希函数拥有较小的碰撞概率,并且可以忽略较小的碰撞概率对大型数据集的影响[29]。将 u_i 依次输入,调用由社区式问答平台定义的哈希函数 $\boldsymbol{H}_j(x)$,将最小哈希值作为特征集签名矩阵的元素,由于使用 7 个哈希函数,重复上述步骤后将最终获得拥有 7 个元素的签名矩阵 $\boldsymbol{H}_{\min}(\cdot)$。

(3)计算相似度。

使用 Jaccard 算法计算已知 Minhash 签名矩阵的 2 个文本的相似度

$$J(A,B)=\frac{|\boldsymbol{H}_{\min}(A)\cap\boldsymbol{H}_{\min}(B)|}{t},\quad (2)$$

式中 t 为哈希函数的个数。

由式(2)可知,t 越大,Jaccard 的准确率越高,但计算效率越低。从理论上讲,如果 t 足够大,则估计值和精确值几乎相等,但考虑到签名矩阵上传到智能合约时所需的成本,本研究设置 t 为 7[29-30]。

用户上传回答时,可以首先在链下进行快速和低成本的查重检测,避免与已有回答相似度过高,当通过检测后,便可将回答文本的 IPFS 地址和 Minhash 签名矩阵等信息上传至智能合约。当用户对平台展示的结果存在异议时,可以从智能合约获得平台已上传的 a_j、b_j 和 P 的具体数值,再根据 Minhash 签名矩阵进行检验和证明。需要说明的是,在实际应用场景中,设置查重检测并不是禁止语义重复的回答出现,因为在大部分用户诚实的情况下,同一问题下回答的主要内容应当是趋同的,但它应当能够防止用户通过简单复制的方式,导致完全重复的、无意义的回答出现,进而在一定程度上提高回答的多样性。

2.5　主要操作

区块链网络中存在诸多提供相似服务的平台,因此客户应首先在意向平台内使用“注册”功能,在输入昵称后成为该平台的注册用户,平台更新用户地址列表。一个新用户的积分和惩罚记录初始都为 0。注册成功后,客户可以使用“上架”功能为自己添加新产品或服务,并为它们设置相应的关键属性,客户获得企业身份。

其他客户浏览到感兴趣的产品或服务时可以使用“提问”功能向已购买客户咨询自己所关心的问题，拥有权限的客户可以根据问题内容和自己的实际体验使用“回答”功能上传评论，如果当前产品或服务的积分发放数未达到限定值，回答者将收到卖方设定的固定数量的积分奖励。

提问者根据回答确定购买意愿，使用“购买”功能支付相应数量的代币，若其拥有该产品或服务的积分，则在付款时积分将自动兑换为折扣。交易成功后，提问者也获得评分的权限，其可以使用“评分”功能，向给予正确意见的回答作出较高的评分（例如大于7分），这些回答也将成为其他客户拥有相同问题时的重要参考，当然，相应的优质回答者也将获得额外的积分发放和提升声誉的奖励。

为了提升回答的可参考性和用户的可选择范围，除了被采纳的精华回答外，客户也可调用评分“计算”功能选择合适的计算策略（例如加权求均分）。获得所有回答的均分后，用户可以在链下将计算结果进行排名，结合名次较前（例如前10）的回答来一同参考，因为该结果依赖于所有参与评分的已购买用户，所以这样综合起来得到的意见将更加可靠。在提问、回答和评分过程中，相应的问题（回答）IPFS（interPlanetary file system）地址、分数、时间戳、调用者地址和声誉等信息都将根据功能需要保存在智能合约中。

3 智能合约设计

本章将按照前面所述的平台框架，设计能够实现各个实体在组件中所需要的相关功能和操作的智能合约，按照UML（unified modeling language）标准建模的智能合约关系图如图2所示，其中每个方框（类）由名称、属性和功能三部分组成，为了方便查看，图中省略了部分字段。

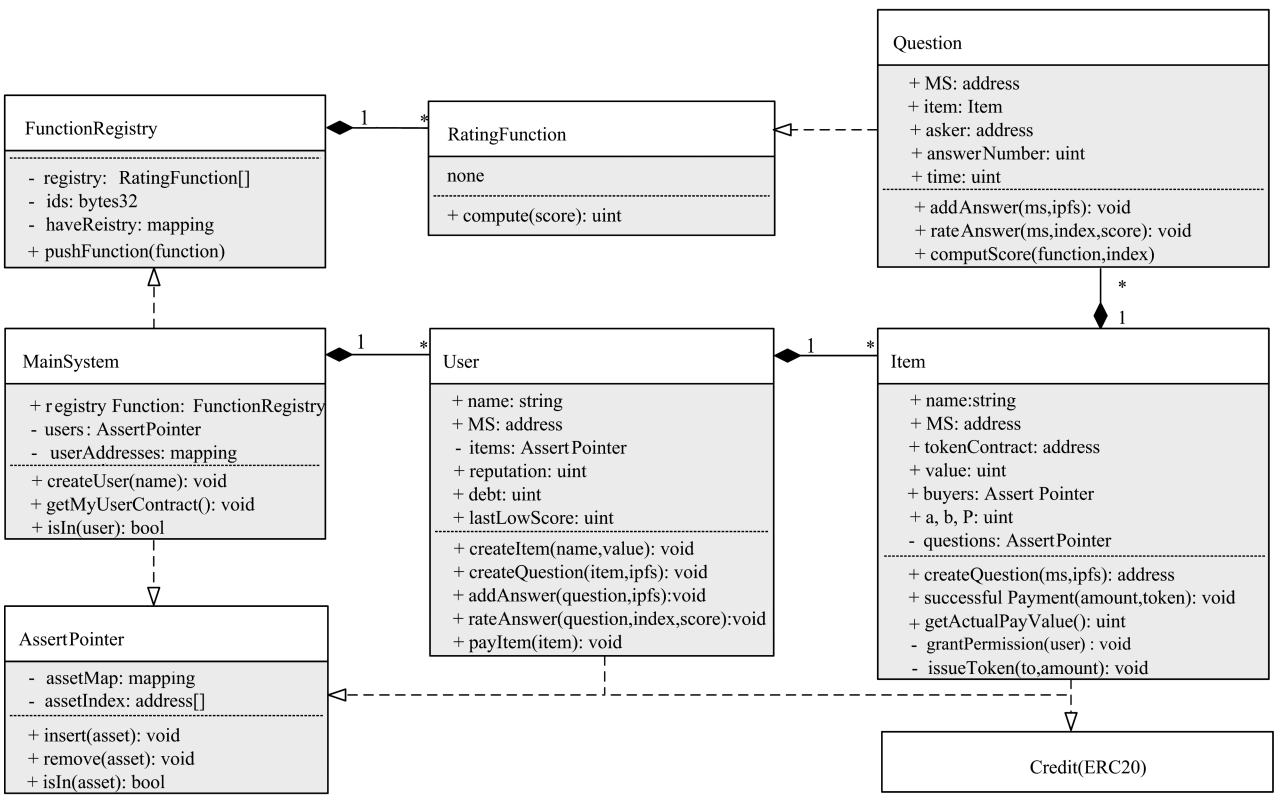

图2 智能合约框架

Fig.2 Smart contracts framework

3.1 智能合约框架

MainSystem（主）合约负责创建和存储平台已注册用户信息，通过createUser（）功能为调用者生成相应的用户合约。User（用户）合约负责为客户提供创建、存储和购买产品或服务的功能以及与问答相关的提问、回答和评分功能。其中，reputation代表声誉度，当客户回答被采纳时获得提升，初始为10；debt表示惩罚记录，当荣誉度为0时该值会发生变化。Item（产品或服务）合约在客户调用User合约中的createItem（）功能时创建，负责提供客户支付时用到的相关信息和功能，同时制定客户回答问题和评分回答权限的规则。客户在自己的

User 合约中调用 createQuestion( ) 功能时会与 Item 合约交互,并在其中调用同名功能,生成问题合约。Question(问题)合约负责存储来自于客户的回答和评分信息。

由图 2 可知,MainSystem 合约与 User 合约、User 合约与 Item 合约、Item 合约与 Question 合约之间都是一对多的关系。Credit(积分)合约按照 ERC20 标准创建,与 Item 合约一一对应,每个 Question 合约在创建时都将获得由创建者设置的固定数目的积分,用以奖励客户回答和评分。AssertPointer 用于存储和记录资产地址的合约,其中,MainSystem 合约用其存储平台内注册客户 User 合约地址,User 合约用其存储已上架产品或服务 Item 合约地址,Item 合约用其存储客户提出的问题 Question 合约地址和已购买客户地址。

为了提供更全面的评分机制,FunctionRegistry(评分功能注册)合约负责认证和存储评分计算合约,并且仅能由平台调用。RatingFunction(评分计算)合约只包含一个 compute( )功能,根据用户的不同选择要求,采用合约中不同的计算策略。

本研究提供了普通求均值和加权平均 2 种计算策略,计算公式分别为:

$$A=\frac{1}{n}\sum_{i=1}^{n}S_i, \tag{3}$$

$$W_A=\frac{\sum_{i=1}^{n}S_i\cdot H_i}{\sum_{i=1}^{n}H_i}, \tag{4}$$

式中:A 为计算的平均数;S_i 为已购买客户对“回答”打出的分数;H_i 为打分者的声誉度,若将权重信息更改为打分者所拥有的该产品或服务的积分数 C_i 时,只需将 H_i 更改为 C_i。加权平均的计算方法,可以减轻临时生成的用户留下的评论有效性,增加声誉度高的用户的优先性。

3.2　场景演示

为了简化对智能合约框架的描述,以 2.5 节中的关键操作为基础,设计如下场景。

(1) TB 是一个电子商务平台,它希望使用本研究设计的问答平台为自己平台内的卖方提供相关服务,因此在区块链网络上部署了 MainSystem 合约,FunctionRegistry 合约随之自动创建,并部署和认证了普通求均分和加权求均分 2 种策略的 RatingFunction 合约,所有合约都将其设置为所有者。

(2) DJ 是 TB 的一个卖方,他通过调用 MainSystem 合约中的 createUser( )功能成为注册用户,获得拥有者为其地址的 User 合约实例。

(3) DJ 通过自己 User 合约中的 createItem( )功能上架了第 1 个商品 SP,在生成相应 Item 合约实例的同时自动创建了用于发放积分的 ERC20 合约。需要注意的是,使用积分的最大限制为“商品价格-1”,即客户无法仅仅使用积分完成购买活动。同时,DJ 将 Item 合约中问题的积分发放总数和声誉度阈值分别设置为 41 和 0。

(4) TW 是平台内的注册用户,在浏览商品时看中了 SP,并向客服咨询关于 SP 的问题,但客服没有及时回答,于是调用自己 User 合约中 createQuestion( )功能生成相应的 Question 合约实例,将问题发布到了该商品的提问社区,生成问题合约时的随机邀请算法如算法 1 所示。

算法 1　随机邀请

输入:邀请数目 M、声誉度阈值 Y

输出:被邀请回答者的地址集合 Z

```
if M>已购买客户总数 N then
Z=所有已购买客户地址集合;
else
  for i=1; i≤M; i++ do
    A=(blockhash(now-i)+address(this))%N; //随机种子。
    j=0;
    while User[A].repution <Y ‖ User[A] 已经被邀请 do
      j++;
      A=(A+1)%N;
      if j==N then
        Y==;
      j=0;
      end if
      end while
      Z[i]=User[A].address;
  end for
end if。
```

(5) HD 是平台内的注册用户并且已经购买过 SP,当他收到社区的事件提醒时,发现了 TW 提出的问题和自己拥有的回答资格,于是根据使用经验上传了回答。回答文本需要经过链下中心化的查重检测,通过检测后获得它的签名矩阵;调用链上 User 合约中的 addAnswer( )功能,输入签名矩阵和其他相关参数(已提前将回答文本上传至 IPFS 并获得相应的内容标识符 CID);User 合约自动与相应的 Question 合约交互并调用其中的同名功

能,成功上传回答后,HD 获得 10 积分奖励。参与客户在使用问答平台时拥有的智能合约如图 3 所示。

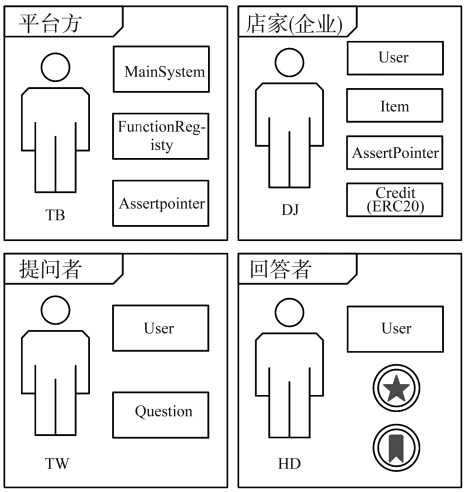

图 3 用户拥有的合约

Fig.3 User-owned smart contracts

(6)TW 收到 HD 发布的回答后确定了自己的购买意愿,于是调用自己 User 合约中的 payItem()功能支付代币,这时与相应 Item 合约交互并调用其中的 successfulPayment()功能将代币和积分转移到 DJ 地址,如算法 2 所示,由于 TW 是新顾客,所以没有这家店的积分,在购买时也无法享受减免优惠。购买成功后合约自动调用 grantPermission()功能授予 TW 评分权限。

算法 2 购买和支付

输入:Item 合约地址

输出:交易结果

amount = msg. value;//msg. value 为输入代币数。

require(amount >0); //保证无法仅使用积分完成交易。

从 Item 合约获得拥有积分总数 totalTokenAmount;

从 Item 合约获得实际应支付金额 actualPayment;

if actualPayment ! = 0 && amount ≥ actualPayment then

item. tokenContract ( ). approve (address (item), totalTokenAmount); //支付积分。

item. successfulPayment { value: actualPayment}(actualPayment,

totalTokenAmount); //支付代币。

else if actualPayment==0 then

totalTokenAmount = item. value ( ) - amount;

item.tokenContract().approve(address(item), totalTokenAmount); //支付积分。

item. successfulPayment { value: amount } (amount, totalTokenAmount); //支付代币。

end if。

(7)一段时间后,TW 收到了 SP 的快递,并且实际效果与 HD 的回答一致,于是调用自己 User 合约中的 addRate()功能给予了该回答 10 分(评分范围 0~10)评价,这时与相应 Question 合约交互并调用其中的同名功能,该回答被设置为“可信回答”,TW 获得 1 积分参与奖励,HD 获得“积分+1”和“声誉+1”的贡献奖励,如算法 3 所示。

算法 3 评分回答

输入:回答索引号、分数

输出:交易结果

if 调用者==提问者 then

answers[index]. answerer. updateReputation();//更新提问者荣誉度,该回答被设置为精华。

触发事件 AnswerConfirmed();

end if

answers[index].scoreArray.push(score);

answers[index].raterArray.push(user);//记录评分者信息。

item. tokenContract ( ). transfer (address (user), 1);//奖励评分者 1 个积分。

if score <3 then

require(距离调用者上次给低分已经经过了 3 d);

if 回答者 reputation ==0 then

回答者的 debt += 1;

else

回答者 reputation -= 1;

end if

answers[index].lowLog += 1; //记录该回答已获得低分的次数。

if answers[index].lowLog == 10 then

delete answers[index]; //删除该回答在合约中的记录。

end if

end if

if score>7 then

item.tokenContract().transfer(address(answers[index].answerer), 1); //获得高分,奖励回答者积分。

end if。

激励执行流程如图 4 所示,当 Question 合约分

配到固定的积分数后,由其按规则自动分发。

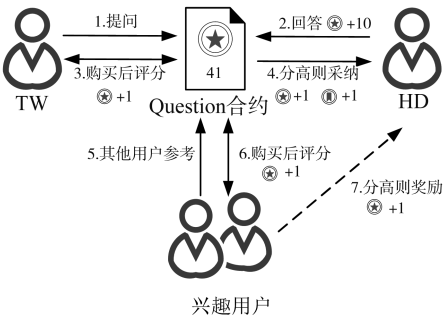

图 4　激励机制执行流程

Fig.4　Implementation process of incentive mechanism

上述场景中主要操作(交易)的流程图如图 5 所示,为简化结构,调用者(TW 和 HD)的用户合约实例统一表示为 User 合约对象,其根据调用者不同而不同。

现实情况中,客户可能通过其他渠道购买产品或服务,而这些交易在问答平台的智能合约中都没有记录,也就无法获得回答和评分的权限。为了尽可能提高回答来源的可信度,本研究并没有给卖方提供额外的授权功能,亦即客户必须通过该平台交易后才能获得相关权限。在估量回答的真实和可靠程度时,客户可以根据自己的策略需要设计 RatingFunction 合约,本研究使用了普通求均分和加权求均分(权值为评分者的声誉度)2 种策略的评分计算合约,应当鼓励用户调用经过平台认证的评分计算合约,避免未认证合约作恶即产生不正确计算结果的情况发生。问答数据的处理过程如图 6 所示。

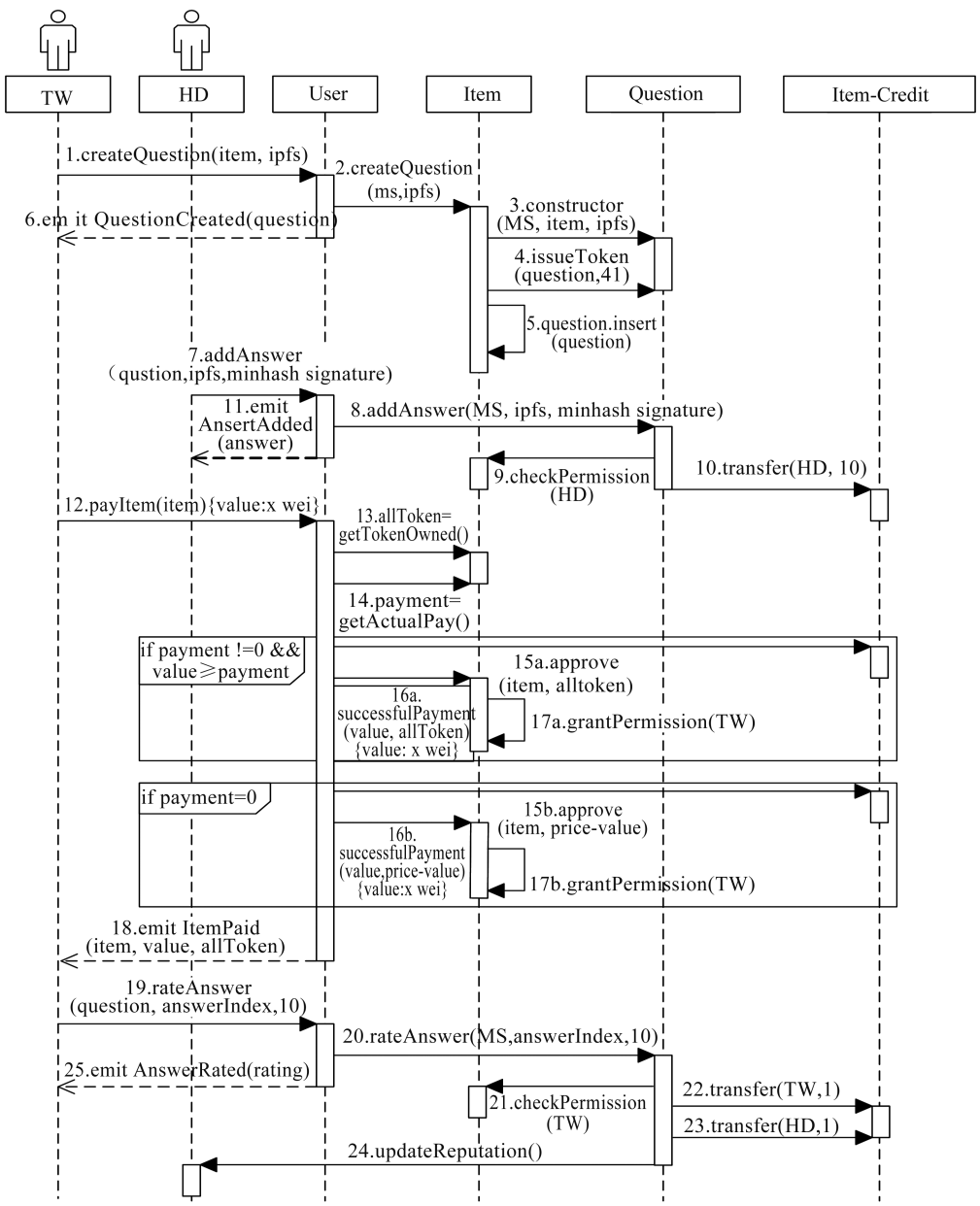

图 5　主要交易流程图

Fig.5　Main transactions flowchart

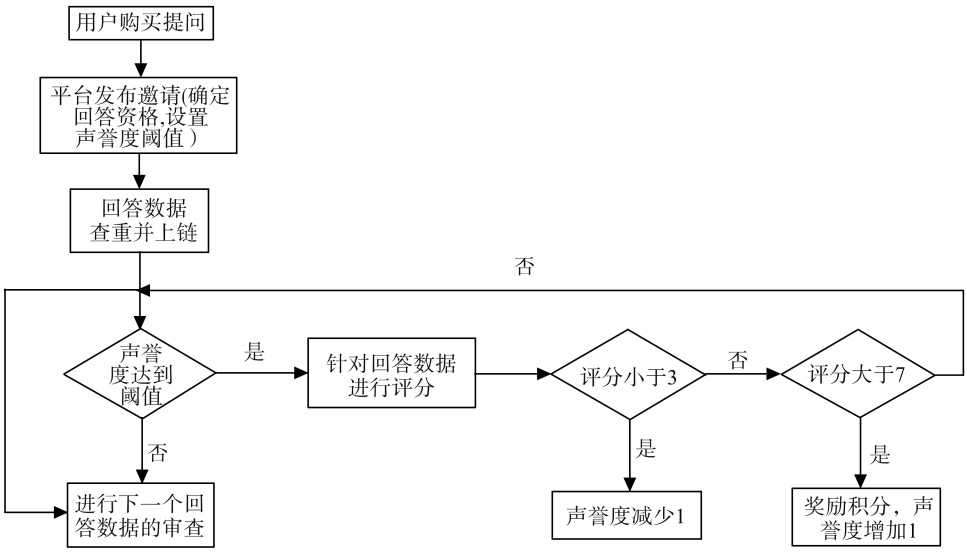

图 6 问答数据处理过程图

Fig.6 Q&A data processing process chart

4 试验与分析

4.1 功能测试

本节中将 Solidity 语言编写的智能合约部署在 Ganache-cli 提供的以太坊[31]私链环境中,并按照本研究设计的场景进行功能测试。试验中用到的账号地址和区块链环境配置如表 2 所示。

表 2 区块链配置表

Table 2 Account addresses and blockchain configuration

环境配置	参数信息
开发框架	Truffle 版本为 5.2.2
智能合约编译器	Solc 版本为 0.8.6
以太坊客户端	Ganache-cli 版本为 6.12.2
Default Gas Price	0
BlockGas Limit	80 000 000
Call Gas Limit	50 000 000

试验中相关实体拥有合约的名称和地址如表 3 所示。

表 3 部署后智能合约地址

Table 3 Smart contract address after deployment

名称	地址
TB	0xc9a5EA40c936BDdd7f5B83Ab0B7f6321b70A9aE1
User-DJ	0x31A4f743d7b5fdAA7D40F00e75f7130970fd30f3
User-TW	0x3F218D24dc2F06Ca36750F791f0C77bacec9f738
User-HD	0x86385838E41358Bd3225d67F1549c47C30CbE905
Item	0x90c271963b599853a1e1b765A012B012644254fe
Credit	0x2700700e82e730F0b3a8E1A17fb8b2082DD775Fe
Question	0xE3D134f49Cb7b8cdf82cBf9536B402e81793D88c

TW 调用 createQuestion( ) 功能时的交易记录,触发 QuestionCreated( ) 事件。

Question {

address: '0x3F218D24dc2F06Ca36750F791f0-C77bacec9f738',

blockHash: '0x08f4c9b937a0c0643637fced3c29-facec53ddca443a7809bfaba8f12d29dc735',

blockNumber: 10,

logIndex: 1,

removed: false,

transactionHash: '0xf42d541df6e090cb04ca81c-070b6fce17119eba09969d75a049e86e^{0}692eala',

transactionIndex: 0,

id: 'log c004ab93',

event: 'Newauestion',

args: Result { 'o': '8XE3D134f49Cb7b8cdf-82cBf9536B402e8173Dc',

length: 1, question: '0xE3D134f49Cb7b8cdf82-cBf9536B402eD88' }。

HD 调用 addAnswer( ) 功能时的交易记录,触发 AnswerAdded( ) 事件。

Answer {

address: '0x86385838E41358Bd3225d67F1549-c47C30CbE905',

blockHash: '0x41d27280da20a2d6ebad1f76aad-4cbfcb7d4f80544534a5789a75c91afddc536',

blockNumber: 11,

logIndex: 1,

removed: false,

transactionHash: '0x596e14511343c50c6e31ea-

950a296d549aaa3f33b4fe855231ab76578fcb4b45',
transactionIndex:0,
id:'log_01c28f66',
event:'NewAnswer',
args:Result{'o':'0XE3D134f49Cb7b8cdf82-cBf9536B402e81793Dc',
'1':'QmaR3PW3MEP95xj3iFQ8w48Hsiie4k-VFdaMBSSK2XMQ7Ce',
'2':0x1BBA2766dA503908cE00D1FbcAa-2226Bc1dddF4b',
length:1, question:'0xE3D134f49Cb7b8cd-f82cBf9536B402e81793D88c'
'3':'0x701e465f411c71e0ba51d8e2ee771d-de27doe042fb85ad042fb85ad01e465f4',
question:'xE3D134f49Cb7b8cdf82cBf9536B4-02D88c',
ipfs:'QmaR3PW3MEP95xj3iFQ8w48Hsiie-4kVFdaMBSSK2XMQ7Ce',
answer:'0x1BBA2766dA503908CE00D1FbcA-a2226Bc1dddF4b',
minhash:'0x701e465f411c71eoba51d8e2ee771-dde27doe042fb85ad042fb85ad0le465f41'。

TW 调用 payItem()功能时的交易记录如图 7 所示,触发 ItemPaid()事件。

图 7　“购买”产品交易信息
Fig.7　"Buy" product transaction information

TW 调用 rateAnswer()功能时的交易记录如图 8 所示,触发 AnswerRated()事件。

图 8　“评分”回答交易信息
Fig.8　"Rate" answer transaction information

4.2　成本分析

在以太坊网络中部署智能合约、执行交易、与智能合约交互都需要消耗一定的 Gas,交易花费是指整个交易消耗的 Gas 量,执行花费是指 EVM 执行合约代码时消耗的 Gas 量,使用 2020 年 1 月 6 日的以太 ether(1 ETH = 135.5 USD)和平均速度 Gas 价格(1 Gas = 9 Gwei, 1 ETH = 10^9Gwei),交易费用等于 Gas 消耗量乘 Gas 单价。

试验中交易消耗的总 Gas 量和所需的美元成本如表 4 所示,除了电商平台部署合约时需要承担较多的费用外,卖方和消费者调用功能的交易成本都在可接受范围内。用户注册功能 createUser()因为涉及到创建新合约而且相关功能全部集成在了 User 合约中,所以消耗的 Gas 最多,成本也相对较高,但对于客户而言只需要调用一次,虽然提高了成本但之后的调用操作变得更加便利。

表 4　交易成本
Table 4　Transaction costs

合约名称	调用者	操作	消耗 Gas/ETH	成本/美元
TB(0x72..aA)	TB(0x49..f0)	deploy	5 586 533	6.812 7
TB(0x72..aA)	TW(0x5b..05)	createUser()	4 657 939	5.680 3
User(0x3c..E5)	DJ(0x11..aE)	createItem()	3 640 148	4.439 1
User(0xE4..ca)	TW(0x5b..05)	createQuestion()	1 221 302	1.489 3
User(0xc3..92)	HD(0x55..41)	addAnswer()	269 622	0.328 8
User(0xE4..ca)	TW(0x5b..05)	payItem()	129 535	0.157 9
User(0xE4..ca)	TW(0x5b..05)	rateAnswer()	9	2.342 0

4.3　性能分析

区块链平台认为性能是最受关注但又常常被忽略的问题,给定区块链平台的性能可以通过执行时间衡量[32]。文献[32]提出了一种通过评估工作量的方法作为衡量区块链系统平台的性能指标。工作量表示发送到给定区块链平台的交易请求数量。本研究即是通过评估工作量,通过记录对比智能合约的关键操作执行 100 次的时间,进而研究和评估其对本研究设计的区块链系统的性能影响。

在 VMware 虚拟机中运行 Ubuntu 20.04 (64 bit)操作系统,分配有 4 GB 内存,2 个 AMD 4800U 1.80 GHz处理器且核心数为 2,使用 Truffle 框架提供的测试功能,在 Ganache-cli 以太坊私链环境中评估本研究开发的智能合约关键操作(注册用户 createUser()、上架产品 createItem()、提问 createQuestion()和回答 addAnswer()功能)的时间消耗。试验结果如图 9、10 和 11 所示。

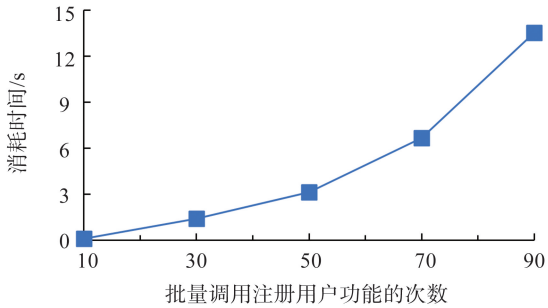

图 9 createUser( )功能性能测试

Fig.9 Performance test of the createUser( ) function

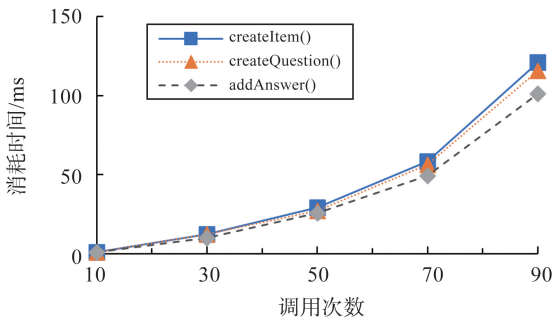

图 10 其他功能性能测试

Fig.10 Performance testing of the other functions

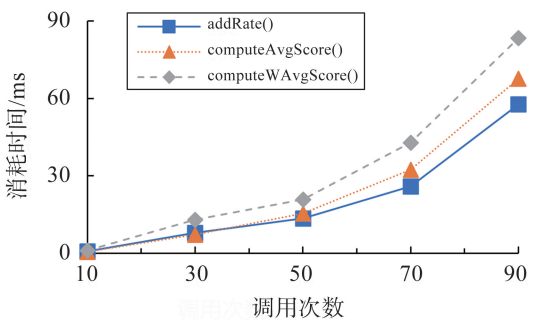

图 11 评分相关功能性能测试

Fig.11 Rating related functions performance test

与实现功能的复杂度相对应,注册用户 createUser( )消耗的 Gas 量最多,所需时间也最长,由图 9 可知,执行 100 次平均消耗 13.532 s,这也是唯一达到秒级的调用。由图 10 可知,addAnswer( )只涉及合约状态更改,因此时间消耗最少,执行 100 次平均消耗 101.114 ms, createItem( )和 createQuesiton( )相近,执行 100 次前者平均消耗 121.119 ms,后者平均消耗115.883 ms。

调用 addRate( )和 computeScore( )功能的性能消耗如图 11 所示,调用 100 次评分功能时平均消耗 57.739 ms,随着评分数量的增加,普通求均分和加权求均 2 种评分计算策略的消耗也都在不断增加,当处理 100 条分数时,前者平均花费 67.713 ms,由于后者还需要处理权重信息,所以消耗更多,平均花费 83.396 ms。大部分功能执行 100 次平均所需要的时间都是毫秒级,因此在可接受范围内。

执行事务所需的时间取决于传播请求的网络延迟和挖掘。在撰写本文时,以太坊的平均开采时间为 13~14 s[18]。通过调用合约的几个关键功能进行工作量评估,进而发现这些功能进行一定工作量所花费的时间成本较低,可以接受。这也证明了本研究设计的系统可以以低成本的方式运行。

4.4 相似度分析

从提供社区式问答服务的中心化购物平台中获得问答数据(问题拥有的回答数量≥10),并以优质(排名第一位)回答作为参考,与其他回答进行比较,试验中使用的 7 个哈希函数 a_j 和 b_j 的取值如表 5 所示,另外大素数 P 为 4 294 967 311。

表 5 a_j 和 b_j 的值

Table 5 The value of a_j and b_j

a_j	b_j
1 432 518 515	919 446 848
3 617 697 886	1 167 230 696
3 148 201 956	1 059 205 949
3 008 659 646	897 178 059
1 694 972 510	4 262 371 542
2 172 439 383	3 586 029 784
2 149 673 042	3 028 144 581

将回答文本根据 2.4 节中的前 2 个步骤处理后,获得相应的 Minhash 签名矩阵 $\boldsymbol{H}_{\min}(\cdot)$,为了节省上传 $\boldsymbol{H}_{\min}(\cdot)$时的 Gas 费用,在上传前可以先对其进行压缩处理。如图 12 所示,把每个元素转换为 16 进制,并在上传回答时将其以 bytes32 形式保存到智能合约中,其中每个元素长度为 9,首部第 1 个位置为符号位,所以总长度为 63。

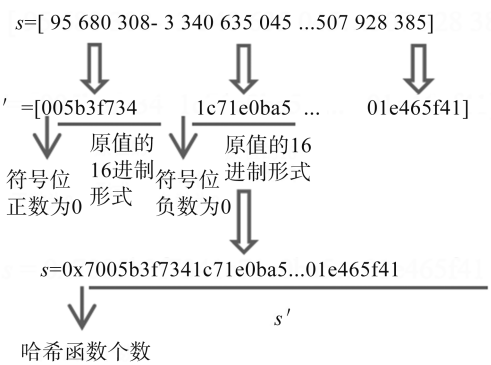

图 12 压缩 Minhash 签名

Fig.12 Compressed Minhash signature

使用压缩和不压缩策略后,调用 addAnswer( )功能所需的成本如表 6 所示,其中 n 为哈希函数个数。显然,压缩签名矩阵后用户使用成本更低,且在链下可迅速复原,不会对比较相似度时的结果造成影响。

表 6　成本比较
Table 6　Cost comparison

压缩策略	Gas 消耗/10^3	成本/美元
压缩(n=7)	269.622	0.328 800 00
压缩(n=20)	282.870	0.344 959 97
不压缩(n=7)	279.307	0.340 620 00
不压缩(n=20)	296.168	0.361 170 00

根据回答文本生成的 Minhash 签名矩阵和 Jaccard 算法计算得到的平均相似度如表 7 所示,因为中心化平台在展示时已经进行了一定程度的过滤,所以选取的同一个问题下的回答相似度都处于较低的水平。当使用 7 个和 20 个哈希函数时,它们的平均相似度均为 0.1,且使用 7 个哈希函数时产生的 Minhash 签名矩阵更小即上传成本更低,因此本研究将相似度阈值设置为 0.1。

表 7　平均相似度
Table 7　Average similarity

商品	平均余弦相似度	
	n=7	n=20
某手机	0.173 33	0.094 4
某电脑	0.078 80	0.138 8
某电视	0.203 30	0.127 7
某台灯	0.031 10	0.055 5
某冰箱	0.093 30	0.133 3

5　讨论

5.1　安全性分析

(1)不可篡改。本研究设计的问答平台部署在区块链之上,在采用 POW 工作量证明共识的公有链中,修改区块数据需要拥有超过全网 51% 的算力,这是一个成本极高且难以完成的攻击;在联盟链中,若其他联盟成员不认可攻击者发布的恶意区块,该攻击便无法完成。这种特性保证了卖方无法恶意篡改用户回答,通过引导舆论方向欺骗新用户并获利。

(2)DDoS 拒绝服务攻击。在智能合约上执行功能都需要消耗一定的 Gas,一般区块链中对 Gas 的最大数量都有限制,例如以太坊中为 8 000 000,当 Gas 消耗完毕后功能将停止执行,若功能未执行成功,则交易恢复到初始状态,另外,交易费用的支出也增加了攻击成本。

(3)信息可信度。①回答信息可信度。客户获得回答权限,必须在链上有购买商品的交易记录,而且要满足邀请条件,整个过程公开透明、可验证,这增加了客户恶意刷评时的成本;平台禁止已购买客户提问并回答的自问自答形式(但允许未购买客户提问,购买物品后进行问题回答)发布误导性回答,若仍然有恶意回答存在,假设大部分已购买用户诚实的情况下,该回答将获得很低的分数或者被直接删除;为了处理意外事件,平台内可以设立中心化或去中心化仲裁者的特别实体[21],其应当在用户中有一定的声誉,并且经过实名认证,智能合约赋予其特殊权限,用于管理回答社区,删除恶意的无效回答,并扣除回答者相应积分和声誉度。②排名信息可信度。用户在平台内的交易全部上链,公开透明,虽然平台内提供了计算分数的功能,但由于智能合约的自身限制,一方面部分复杂算法无法实现,另一方面可以实现的也需要消耗大量的费用,所以“对回答进行排名”应当放在链下进行,但用户可以根据链上信息在链下对排名进行检验,以实现对平台的监督。若平台因相似度过高对用户的回答进行了屏蔽,用户对此有异议时,同样可利用链上的数据和参数,在链下进行快速和低成本的校验。

5.2　与其他电商平台的比较

因为现有的区块链和问答平台相结合的研究处于空白阶段,本节中将传统中心化的电商问答平台与本研究设计的基于区块链和智能合约技术的社区式问答服务平台进行比较,如表 8 所示。

表 8　项目比较
Table 8　Project comparison

项目	特征	透明度	数据可信度	交易信息可验证	问答数据不可篡改	激励	排名机制
淘宝问大家	中心化	低	低	否	否	无	基于回答时间、字数、点赞数和回答者等级
京东问答	中心化	低	低	否	否	无	基于回答时间、点赞数和回答者等级
亚马逊提问和回答	中心化	低	低	否	否	无	基于回答时间或投票数
本研究设计	去中心化	高	高	是	是	积分激励和声誉激励	可基于评分和回答者声誉与贡献度等数据设计各种机制

透明度指交易过程的透明性,这是提高回答者“已购买用户”身份可信度的有力方式,利用区块链的技术特征,本研究设计的平台在去中心化、透明度、不可篡改和信息可验证上都有很大优势。

在随机邀请算法中,本研究设计的随机邀请机制相较于淘宝问大家的“随机邀请回答”机制具有可验证性,即用户可以根据智能合约内设定的规则,确定回答者群体的具体地址。在查重检测中,用户也可以根据智能合约中获得的结果,在链下进行简单和快速的验证。

评分计算机制是本研究设计的问答平台的另一个优势,客户可根据已有数据设计不同的RatingFunction合约,通过差异化的权重和排名结果,为用户提供更多的参考,在未来的可拓展性上也更强。相较于传统中心化平台,本研究设计的平台中,用户对于平台所展示的任何结果,都能够进行验证。

关于提问数量和间隔等限制,本研究设计的问答平台可由平台和卖方通过修改少量代码自行配置,这种差异化为客户使用提供了更大的灵活度。

反馈方式是客户对回答的认同度表现,主要用来为购买用户(尤其是参考回答后的)提供反馈渠道,有利于筛选出真正有用的回答,而传统问答平台中往往忽略当未购买产品的提问者成为已购买者后,对其发布提问下的回答进行反馈的重要意义。另外,相较于单一化的点赞或踩的评价机制,采用十分制具有更宽泛的选择性。

目前传统的中心化问答平台普遍缺乏激励机制,而这并不利于用户的加入和问答平台的发展,但在本研究中利用积分和声誉激励机制,将有效调动起客户参与社区问答和已购买客户发布高质量回答的积极性。

除了激励机制外,本研究设计的平台内也包含了基于声誉度和评分的惩罚机制。企业通过串通用户给予正向评论或回答,从而引导用户购买商品的欺诈行为也存在于问答平台中,但公开可验证的随机邀请机制增加了这样联合作恶的难度,另一方面,即使该行为仍然发生,但在平台内大部分用户诚实的情况下,由于评分激励的存在,恶意回答者也会因为低分获得扣除声誉度的惩罚,进而被标识出来,通过一定程度的社区自治提高了平台内回答者的整体质量。

6 结束语

为了解决传统中心化社区式问答平台中缺乏数据可信度和行为激励等问题,本研究提出一种基于区块链智能合约的电商社区式问答服务平台。该平台主要由购买、提问、回答和评分4部分组成,通过智能合约为多方实体提供灵活高效、公开透明、操作留痕和不可篡改的社区式问答服务。在提问模块中,通过随机邀请机制,提高了用户恶意刷评的成本;在回答模块中,引入了基于积分和声誉的激励机制,这不仅有利于已购买用户积极发布高质量回答,而且提高了客户、企业和平台之间的黏性;评分模块通过有效利用回答者声誉、分数和身份等信息,设计基于不同策略的评分计算合约,为客户提供了多样性和自主性更强的排名参考。与此同时,使用基于Minhash和Jaccard算法设计的查重检测机制,并将回答数据生成的Minhash签名保存在智能合约中,保证了易用和可验证性,也为保护用户对回答内容的所有权和提高回答的多样性提供了一种解决方案。在Ganache-cli提供的以太坊私链环境中,使用Truffle开发工具部署了本研究设计的社区式问答平台的智能合约,通过对设计场景的再现和关键操作的性能分析,验证了该平台的可行性。

未来将进一步完善和优化代码,探索将深度学习与零知识证明技术相结合,通过密码学手段证明呈现结果与实际模型运行结果相一致的方案。也将探索应用密码学和预言机等技术加强合约交互时的安全性、交易的隐私性和回答的可靠性的方案。另外,问答平台本身仍在不断发展和迭代,在未来也需要持续探讨基于区块链问答平台的适用性。

参考文献:

[1] 徐佳敏. 客服行业:除痛点找拐点[J]. 人力资源, 2021(15): 44-46.
XU Jiamin. Customer service industry: remove pain points and find inflection points[J]. Human Resources, 2021(15): 44-46.

[2] 王海茹. 互联网企业客服外包风险分析与对策研究[D]. 武汉:华中师范大学, 2019.
WANG Hairu. Risk analysis and countermeasure research of customer service outsourcing of Internet enterprises[D]. Wuhan: Central China Normal University, 2019.

[3] SHENG J, AMANKWAH-AMOAH J, WANG X. A multidisciplinary perspective of big data in management research[J]. International Journal of Production Economics, 2017, 191:97-112.

[4] 浦娟. 电商问答对购买意愿的影响机制研究[D]. 南京:南京大学, 2020.
PU Juan. Research on the influence of E-commerce

questions and answers on purchase intention[D]. Nanjing: Nanjing University, 2020.

[5] 李娜. 电商在线问答的信息属性和社交属性对消费者购买意愿的影响[D]. 西安:西安电子科技大学, 2020.
LI Na. The impact of information aspect and social aspect of E-commerce online Q&A on consumers' purchase intention[D]. Xi'an: Xidian University, 2020.

[6] SALAH K, ALFALASI A, ALFALASI M. A blockchain-based system for online consumer reviews [C]//IEEE INFOCOM2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). N Y, USA: IEEE, 2019: 853-858.

[7] BELLINI E, IRAQI Y, DAMIANI E. Blockchain-based distributed trust and reputation management systems: a survey[J]. IEEE Access, 2020, 8: 21127-21151.

[8] OKSIIUK O, DMYRIEVA I. Security and privacy issues of blockchain technology[C]// 2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET). N Y, USA: IEEE, 2020: 1-5.

[9] PECK M E, MOORE S K. The blossoming of the blockchain[J]. IEEE Spectrum, 2017, 54(10): 24-25.

[10] MAGAZZENI D, MCBURNEY P, NASH W. Validation and verification of smart contracts: a research agenda[J]. Computer, 2017, 50(9): 50-57.

[11] AHLUWALIA S, MAHTO R V, GUERRERO M. Blockchain technology and startup financing: atransaction cost economics perspective[J]. Technological Forecasting and Social Change, 2020, 151: 119854.

[12] DAI H N, ZHENG Z, ZHANG Y. Blockchain for internet of things: a survey[J]. IEEE Internet of Things Journal, 2019, 6(5): 8076-8094.

[13] GRIGGS K N, OLYA O, KOHLIOS C P, et al. Healthcare blockchain system using smart contracts for secure automated remote patient monitoring[J]. Journal of Medical Systems, 2018, 42:1-7.

[14] GUO J, LI C, ZHANG G, et al. Blockchain-enabled digital rights management for multimedia resources of online education[J]. Multimedia Tools and Applications, 2020, 79: 9735-9755.

[15] LI Z, GUO H, WANG W M, et al. A blockchain and automl approach for open and automated customer service[J]. IEEE Transactions on Industrial Informatics, 2019, 15(6): 3642-3651.

[16] SABHARWAL C L, ANJUM B. An SVD-Entropy and bilinearity based product ranking algorithm using heterogeneous data[J]. Journal of Visual Languages & Computing, 2017, 41: 133-141.

[17] ZHOU Z, WANG M, YANG C N, et al. Blockchain-based decentralized reputation system in E-commerce environment[J]. Future Generation Computer Systems, 2021, 124: 155-167.

[18] LISI A, DE SALVE A, MORI P, et al. Rewarding reviews with tokens: an ethereum-based approach[J]. Future Generation Computer Systems, 2021, 120: 36-54.

[19] CHEN Y, LI H, LI K, et al. An improved P2P file system scheme based on IPFS and Blockchain[C]//2017 IEEE International Conference on Big Data. Washington, USA: IEEE, 2017: 2652-2657.

[20] ZHOU H, OUYANG X, REN Z, et al. A blockchain based witness model for trustworthy cloud service level agreement enforcement[C]//International Conference on Computer Communications. N Y, USA: IEEE, 2019: 1567-1575.

[21] 曹卫东,胡炜,王家亮,等.基于 SimHash 和混合相似度的多模式匹配方法[J]. 计算机应用研究, 2020, 37(1): 198-202.
CAO Weidong, HU Wei, WANG Jialiang, et al. Multiple schema matching method based on SimHash and mixed similarity[J]. Application Research of Computers, 2020, 37(1): 198-202.

[22] BRODER A Z, CHARIKAR M, FRIEZE A M, et al. Min-wise independent permutations[C]//Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing. N Y, USA: ACM, 1998: 327-336.

[23] 章宁,陈钦. 基于 TF-IDF 算法的 P2P 贷款违约预测模型[J]. 计算机应用, 2018, 38(10): 3042-3047.
ZHANG Ning, CHEN Qin. P2P loan default prediction model based on TF-IDF algorithm[J]. Application Research of Computers, 2018, 38(10): 3042-3047.

[24] JAISWAL A, BABU A R, ZADEH M Z, et al. A survey on contrastive self-supervised learning[J]. Technologies, 2020, 9(1): 2.

[25] 王永贵,郑泽,李玥. Word2vec-ACV: OOV 语境含义的词向量生成模型[J]. 计算机应用研究, 2019, 36(6): 1623-1628.
WANG Yonggui, ZHENG Ze, LI Yue. Word2vec-ACV: word vector generation model of OOV context meaning[J]. Application Research of Computers, 2019, 36(6): 1623-1628.

[26] FARAMARZI N S, DARA A, BANERJEE R. Combining attention-based models with the MeSH ontology for semantic textual similarity in clinical notes [C]//2022 IEEE 10th International Conference on Healthcare Informatics (ICHI). N Y, USA: IEEE, 2022: 74-83.

(下转第 81 页)