边缘计算场景下基于PSI的多方共享缓存隐私保护方案

赖成喆[1,2]，杨婷[1]，秦宝东[1]，曹进[2]
（1.西安邮电大学网络空间安全学院，陕西 西安 710121；2.西安电子科技大学网络与信息安全学院，陕西 西安 710126）

摘　要：针对多方共享缓存中的数据隐私问题，提出了一个支持高效数据共享的多方隐私集合交集（PSI）协议。该协议基于高效的多点不经意伪随机函数（OPRF），且易于扩展到多方环境中。此外，引入了可信的第三方云服务器多关键字检索Top-k算法，为用户提供精确的查询结果。通过安全性分析和效率对比，证明所提协议在半诚实安全模型下实现了计算和通信开销的平衡。

关键词：隐私集合交集；不经意伪随机函数；边缘计算；多方共享缓存；安全多方计算
中图分类号：TP309.2
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2025127

Privacy-preserving scheme multi-party cooperative cache sharing based on PSI in edge computing scenario

LAI Chengzhe[1,2], YANG Ting[1], QIN Baodong[1], CAO Jin[2]
1. School of Cyberspace Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, China
2. School of Cyber Engineering, Xidian University, Xi'an 710126, China

Abstract: To address privacy concerns associated with data in multi-party shared caches, a multi-party private set intersection (PSI) protocol supporting efficient data sharing was proposed. This protocol utilized an efficient multi-point oblivious pseudo-random function (OPRF) designed for seamless scalability in multi-party environments. Additionally, a trusted third-party cloud server implementing a multi-keyword retrieval Top-k algorithm was introduced to provide users with accurate query results. Through security analysis and efficiency comparisons, it is demonstrated that the proposed protocol achieves a balanced trade-off between computation and communication overhead within the semi-honest security model.

Keywords: private set intersection, oblivious pseudo-random function, edge computing, multi-party shared cache, secure multi-party computation

0　引言

随着无线设备和视频流、移动游戏和社交网络等带宽需求大的应用程序的激增，数据流量呈现出爆炸性增长趋势，整个无线网络的容量需求也在不断增加。因此，移动运营商迫切需要开发高性价比的解决方案，以实现高质量服务的可扩展无线访问，满足下一代通信网络中不断增长的流量需求和各种业务的多样化需求。

其中，边缘计算和缓存技术是解决无线网络中上述挑战的最有前途的技术之一[1]。边缘缓存是边缘计算的重要组成部分，通过在离用户更近的边缘节点存储和管理数据，降低数据访问时延，提高带宽利用率，从而增强用户体验。与CPU、内存等传统资源一次只能由一方使用不同，缓存的数据项被

收稿日期：2025–05–28；修回日期：2025–07–04
通信作者：赖成喆，lcz_xupt@163.com
基金项目：国家自然科学基金资助项目（No.62372370, No.U23B2024, No.62172317）；陕西高校青年创新团队基金资助项目
Foundation Items：The National Natural Science Foundation of China (No.62372370, No.U23B2024, No.62172317), The Youth Innovation Team of Shaanxi Universities Foundation

视为公共物品，可以同时为多方服务。网络边缘的设备节点的角色正在由数据使用者向能够进行数据挖掘、模式识别等大数据处理的计算节点转变。

实际上同一服务区域中的许多用户可能会请求相似的内容。因此，通过在非高峰时段提前在网络边缘（如基站、电信中心局和边缘云）主动缓存各内容提供商源服务器的公共内容项，高峰时段的部分请求可以在边缘本地处理，而不必到达原始服务器。边缘计算和缓存技术已成为未来通信网络中增强用户体验、减少回程流量并支持各种物联网应用的关键技术。然而，这一过程也会产生大量的隐私数据安全隐患，因此未来的研究趋势除了边缘计算和缓存的方案设计以外，更重要的是如何融合传统的隐私数据保护方案和边缘计算环境中共享缓存数据的存储特性，让其能在多样化的环境中充分发挥对共享数据的隐私保护能力[2]。

本文提出了一种适用于边缘计算场景下多方共享缓存的安全协议，结合了隐私集合交集（PSI, private set intersection）协议和多关键字检索 Top-k 算法，旨在不泄露底层数据隐私的前提下将公共数据项添加到网络边缘缓存中，且能在改善网络服务效率的同时为用户和内容提供商源服务器的数据提供有效的隐私安全保护。本文协议首先构建边缘网络中各节点的星形拓扑，并引入可信的第三方云服务器作为共享缓存。多个网络运营商在非高峰时期通过多方 PSI 协议将公共项目提前存储到边缘节点的共享缓存中，用户利用关键字来检索所需的文件内容，其服务请求会首先发送到边缘节点，若边缘节点的共享缓存中有用户的请求内容，则可以直接处理用户请求，否则用户需要直接访问内容提供商的源服务器获得需要的内容。

1　相关工作

1.1　边缘计算及其隐私保护研究

边缘计算是一种分布式计算架构[2]，它将计算、存储和网络服务从传统的数据中心迁移到离数据源和用户更近的“边缘”位置，与传统的集中式计算模式相比，具有显著的高效性和更快的响应速度。边缘缓存是边缘计算的重要组成部分，通过在离用户更近的边缘节点存储和管理数据，降低数据访问时延，提高带宽利用率，从而增强用户体验。其架构通常由内容源、边缘节点和客户端设备组成，采用多种缓存策略来优化数据存取。图 1 展示了传统缓存模型与边缘缓存模型的区别，边缘节点可以缓存和预取用户常用的数据和应用程序，访问时直接从边缘节点获取所需的数据和应用程序，用户的请求可以快速处理和响应，减少数据传输时延，也提高了服务的响应速度。

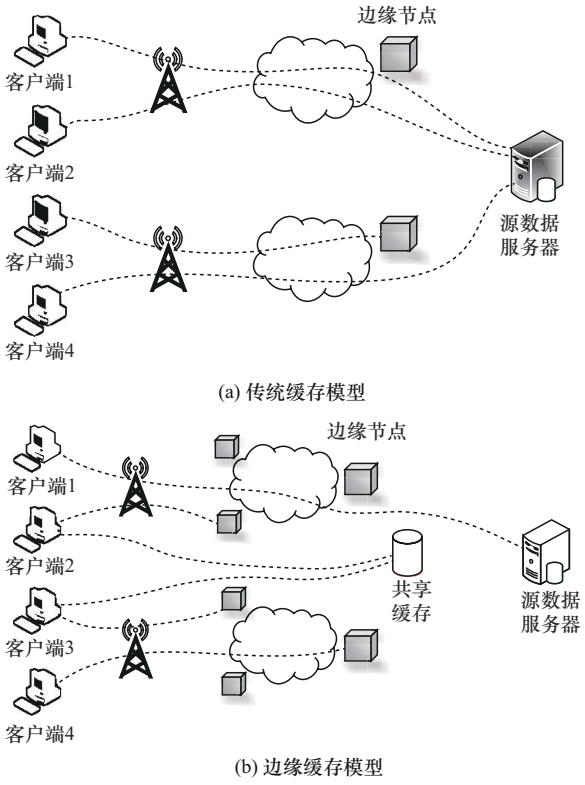

图 1　网络缓存模型

目前边缘计算场景下的隐私保护机制主要依赖于加密技术[3]。通常，数据所有者会预先对数据进行加密处理，然后将加密处理后的数据上传到云端或边缘节点进行存储，再由使用者根据需求进行选择性解密。然而，一些常用的传统数据加密算法，如基于属性加密（ABE, attribute based encryption）、代理重加密（PRE, proxy re-encryption）和全同态加密（FHE, fully homomorphic encryption）等在加密后都会不同程度降低数据的可操作性，导致数据在后续分析和计算过程中受到限制，从而降低协议的效率。

Li 等[4]提出了一种隐私保护的数据利用系统，以私有云代理作为数据所有者与公共云之间的访问接口，从而实现精确的关键字搜索和细粒度访问控制。Bahrami 等[5]利用在移动设备上进行置换操作

的基于混沌系统的伪随机置换实现轻量级加密，并在此基础上提出了移动云计算环境下用于云端数据存储的轻量级加密方法。随后，Pasupuleti等[6]提出了面向移动设备的外包云数据隐私保护方案，结合概率公钥加密和关键字排名搜索算法，在资源受限的移动终端设备上实现隐私保护的排名查询。

沈剑等[7]提出了面向边缘计算场景下隐私保护问题的安全协议，该协议结合基于策略的密钥分配协议和约束伪随机函数分别实现了轻量高效和灵活细粒度的策略选择。Nguyen等[8]首次将PSI技术用于解决网络边缘的协作内容缓存问题，并且提出了一种新颖的隐私保护多方协作缓存（MPCCache, mulit-party cooperative cache）共享框架，它允许多个网络运营商共同确定一组访问频率最高的公共数据项，将其存储在其容量有限的共享缓存中，同时保证其个人数据集的隐私。随后，Zhang等[9]提出了应用于边缘缓存的基于多点不经意伪随机函数（OPRF, oblivious pseudo-random function）的高效多方PSI协议，使用不经意传输（OT, oblivious transfer）协议以及字符串的探测和异或为主要构建块，该协议提供了单方面的恶意安全证明，并实现了通信和计算开销之间的平衡。

1.2 多方PSI协议研究

相比于传统的两方PSI协议，多方PSI协议更能够适用于多个参与方以及更大规模数据的场景，而如何实现协议效率与安全性的平衡也成了近年来的主要研究方向。

Kolesnikov等[10]使用OT扩展（OTE, oblivious transfer extension）协议来实现不经意伪随机函数，并且将此概念运用到PSI中，这也成为后续基于不经意传输的PSI协议的主要方向，但该方案仅适用于两方PSI场景。对于多方PSI，Kolesnikov等[11]首次提出了使用OPRF构造可编程的不经意伪随机函数（OPPRF, oblivious programmable PRF）的概念，基于OPPRF和零秘密共享（SS, secret sharing）协议分别构造了半诚实敌手模型和增强的半诚实敌手模型下的多方PSI协议。为了降低单点OPRF的开销，Chase等[12]使用基于矩阵和OT扩展协议的轻量级的多点OPRF协议构造了多方PSI，有效降低了协议的计算开销，与单点OPRF协议相比，其优势在于在构造两方PSI协议的过程中，发送方的每个元素均只需要加密计算一次即可[13]。

文献[14]在文献[15]的基础上分别提出了半诚实敌手模型和增强的半诚实敌手模型下的多方PSI协议。该协议中直接使用OT协议进行数据传输，且采用混淆布隆过滤器结构，使协议的时间开销与参与方数目呈次线性关系，相比文献[11]有效提升了协议整体效率。在后续研究中，恶意敌手模型下的OT扩展协议也取得了进展。结合文献[16]的星形通信模型和布隆过滤器（BF, Bloom filter）技术，以恶意两方PSI协议[17]为基础，Zhang等[18]提出了能对抗恶意敌手的多方PSI协议，但该协议要求2个特定的参与方不能同时被腐败，故不完全符合标准的恶意敌手模型。针对此问题，Ben-Efraim等[19]设计出了标准恶意敌手模型下抗合谋攻击的多方PSI协议，该协议以文献[17]中的恶意两方PSI协议和文献[14]中的半诚实多方PSI协议为基础同时结合了混淆布隆过滤器（GBF, garbled Bloom filter），是第一个具体且高效的恶意模型下的多方PSI协议，且该协议的在线阶段仅需要2轮通信。

在存储结构的优化方面，不经意键值存储（OKVS, oblivious key-value store）与传统多项式相比节约计算开销，与GBF相比节约存储开销，对计算和存储开销实现了有效的平衡[20]。Nevo等[21]利用可编程的不经意伪随机函数和OKVS技术，是目前为止表现最优的针对恶意敌手的多方PSI协议，该协议在半诚实安全模型和恶意安全模型下被证明是安全的。Jiang等[22]提出了一种基于3H-GCG（3-hash garbled Cuckoo graph）的轻量级OKVS结构，有效减少了原有Cuckoo哈希图可能会出现的哈希冲突，节省了存储空间，并且实现了半诚实模型和恶意模型下的PSI协议安全。

近来，Wu等[23]提出了2种有效的多方PSI协议，实现了半诚实模型下的协议安全，且能抵抗任意数量的合谋攻击。协议中实现了多方情况下OKVS和OPRF的2种细粒度优化，通过引入O-Ring和K-Star这2种拓扑结构，优化了参与方之间的数据传输，显著减少了所需的通信轮次和带宽开销。与此同时，Wei等[24]提出了2种适用于大量参与者和小集合大小的高效多方PSI协议，并在半诚实模型和恶意模型中正式证明安全防止碰撞攻击，在通信和计算方面相较于传统的多方PSI协议取得了更好的性能。Lv等[25]提出了一种名为TH-PSI的高效恶意多方PSI协议，该协议建立在基于可信执行环境

的新型星形拓扑网络通信框架之上，在计算和通信开销方面都具有显著优势。

2　基础原语

2.1　秘密共享

秘密共享指将秘密信息分割成多个份额并分配给不同的参与方，只有当满足特定条件时才能恢复出原始秘密值。秘密值s通过秘密分配算法Share(s,n)分发给n个参与方，其中$t(t \leqslant n)$个参与方通过秘密重构算法来重构秘密值s。本文的PSI协议中采用的是基于多项式插值的Shamir秘密共享方案，方案描述如下。

参数：秘密值s，阈值t，n个参与方$P_i, i \in [1,n]$。

1) 秘密分配

选取$t-1$个随机值$r_1,r_2,\cdots,r_{t-1}$，用于构造多项式。

$$f(x) = s + r_1 x + r_2 x^2 + \cdots + r_{t-1} x^{t-1} \tag{1}$$

选取m个随机数$x_1,x_2,\cdots,x_m$，计算秘密份额$s_i = f(x_i)$并将份额发送给各参与方P_i。

2) 秘密重构

任意t个参与方可通过拉格朗日插值恢复秘密值s，可表示为

$$s = \sum_{i \in S_t} \left[s_i \prod_{j \in S_t, j \neq i} \frac{-x_j}{x_i - x_j} \right] \tag{2}$$

2.2　OT扩展协议

不经意传输协议是安全多方计算领域最重要的密码原语之一。最基础的OT协议是Rabin等[26]提出的2−选−1OT协议，接收方基于选择比特选择性地接收任意一条信息，发送方无法确定接收方的选择比特且接收方也不能获得未选择消息的相关信息。

Beaver[27]首次提出了OT扩展协议，通过少量基于公钥原语的OT实例结合伪随机数生成器、伪随机函数等对称密钥原语实现OT实例的高效扩展。随后Ishai等[28]提出了半诚实敌手模型下OTE协议的经典框架——IKNP（Ishai-Kilian-Nissim-Petrank）协议。IKNP协议利用矩阵变化实现少量1-2OT和对称密钥构造大量OT实例，本文使用的OTE协议相当于IKNP协议的变体，理想功能如下。

参数：协议安全参数k，随机预言机$H:[m] \times \{0,1\}^k \to \{0,1\}^l$，理想的$\mathrm{OT}_m^k$原语。

输入：S输入m对l长的字符串$\left(x_j^0, x_j^1\right), j \in [1,m]$，R输入$m$个选择比特$\boldsymbol{r} = (r_1,r_2,\cdots,r_m)$。

1) S初始化随机向量$\boldsymbol{s} \in \{0,1\}^k$，R随机生成$m \times k$矩阵$\boldsymbol{T}$，用$\boldsymbol{t}^i$表示矩阵的列向量，$\boldsymbol{t}_j$表示矩阵的行向量。

2) 参与方调用OT_m^k协议，其中S作为接收方输入随机向量$\boldsymbol{s}$，R作为发送方输入$\left(\boldsymbol{t}^i, \boldsymbol{r} \oplus \boldsymbol{t}^i\right)$，其中$i \in [1,k]$。

3) 用$\boldsymbol{Q}$表示S收到$m \times k$的矩阵，其中

$$\boldsymbol{q}^i = \left(s_i \cdot \boldsymbol{r}\right) \oplus \boldsymbol{t}^i, \boldsymbol{q}_j = \left(\boldsymbol{r}_j \cdot \boldsymbol{s}\right) \oplus \boldsymbol{t}_j \tag{3}$$

4) S方计算

$$y_j^0 = x_j^0 \oplus H\left(j, \boldsymbol{q}_j\right)$$
$$y_j^1 = x_j^1 \oplus H\left(j, \boldsymbol{q}_j \oplus \boldsymbol{s}\right), j \in [1,m] \tag{4}$$

并将$\left(y_j^0, y_j^1\right)$发送给R。

5) R输出

$$z_j = y_j^{r_j} \oplus H\left(j, \boldsymbol{t}_j\right) \tag{5}$$

2.3　OPRF协议

OPRF的构造方法主要有基于密钥协商DH（Diffie Hellam）和基于OT或OTE协议2种。其中，基于OT协议的又分为单点OPRF理想功能（如图2所示）和多点OPRF理想功能（如图3所示），前者主要依靠公钥操作和逐位操作来实例化大量的OPRF实例。二者之间最主要的两点区别在于多点OPRF允许发送方自己选择伪随机函数的密钥，且更适用于多方环境。本文使用的多点OPRF协议描述如下。

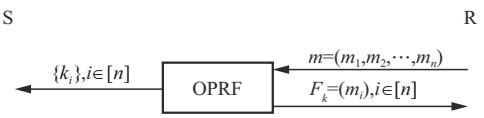

图2　单点不经意伪随机函数理想功能

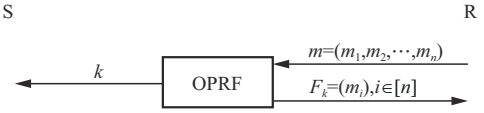

图3　多点不经意伪随机函数理想功能

参数：抗碰撞的哈希函数$H:\{0,1\}^{*}\to\{0,1\}^{w}$，不经意伪随机函数$F:\{0,1\}^{*}\to\{0,1\}^{w}$，理想的$\mathrm{OT}_m^w$协议，协议参数$m$和$w$，协议接收方R和发送方S。

输入：R选择m个查询字符串$q=(q_1,q_2,\cdots,q_m),q_i\in\{0,1\}^{*}$。

1) S选择w长的随机字符串$s\leftarrow\{0,1\}^{w}$。

2) R随机选取伪随机函数（RPF）F的密钥k，生成$m\times w$矩阵$\boldsymbol{A}$和$\boldsymbol{B}$。其中矩阵$\boldsymbol{A}$和$\boldsymbol{B}$的行向量分别为

$$\boldsymbol{A}_i=F_k\left(q_i\right),\boldsymbol{B}_i=\boldsymbol{A}_i\oplus H\left(q_i\right) \tag{6}$$

3) S与R调用OT_m^w协议。S作为接收方输入随机字符串$s=(r_1,r_2,\cdots,r_w)$，R作为发送方输入矩阵$\boldsymbol{A}$和$\boldsymbol{B}$的列向量$\left(\boldsymbol{A}^j,\boldsymbol{B}^j\right)$，S输出$\left\{\boldsymbol{C}^j\right\}_{j\in[k]}$。

4) S生成$m\times k$的矩阵$\boldsymbol{C}$，其中

$$\boldsymbol{C}_i=F_k\left(q_i\right)\oplus\left[H\left(q_i\right)\wedge s\right] \tag{7}$$

5) S随机选取PRF F的密钥k'，计算$\boldsymbol{C}_i\oplus F_{k'}(s)$并将其发送给R。

6) S计算

$$\boldsymbol{C}_i\oplus F_{k'}(s)\oplus F_k\left(q_i\right) \tag{8}$$

并将其作为OPRF的输出。

2.4 相关性评分

相关性评分是信息检索和数据挖掘领域中用来评估某个文档与查询关键词之间匹配程度的常用指标，其目的是帮助用户从大量信息中找到最相关的内容。本文采用被广泛使用TF-IDF（term frequency-inverse document frequency）加权法和建立向量空间模型这2种方法来计算相关性评分[29]，前者用于描述文件中单个关键词的权重，后者在描述多关键字在文件中的权重同时允许查询关键字与文件之间的连续相似度。

TF-IDF加权法涉及2个属性：词频和逆文档频率。词频（$\mathrm{tf}_{t,f}$）表示某个关键词t在文件f中出现的频率，定义为

$$\mathrm{tf}_{t,f}=\frac{\text{关键词}t\text{在文件}f\text{中出现的次数}}{\text{文件}f\text{的总词数}} \tag{9}$$

文档频率（df_t）指包含该关键词的文件数量，逆文档频率（idf_t）用来衡量该关键词在整个文件库中的权重，定义为

$$\mathrm{idf}_t=\mathrm{lb}\frac{N}{\mathrm{df}_t} \tag{10}$$

其中，N表示文件总数。故TD-IDF加权法计算式为

$$\text{tf - idf}_{t,f}=\mathrm{tf}_{t,f}\times\mathrm{idf}_t \tag{11}$$

向量空间模型是一种代数模型，用于对多关键字进行评分。该模型将文件表示为向量$\boldsymbol{v}_f$，向量的每个维度代表一个关键词，即如果该关键词出现在文件中，那么它在向量中的值为非零，否则为零。向量空间模型支持多项和非二进制表示且允许计算查询和文件之间的连续相似度，然后根据它们的相关性对文件进行排名，满足了本文对多关键字检索Top-k算法的需求。向量空间模型将查询表示为向量$\boldsymbol{q}$，向量的每个纬度根据是否查询该关键词分配为0或1，则该查询与文件的相关性评分由2个向量的内积推导得出，如式(12)所示。

$$q\left(\mathrm{score}_{f,q}\right)=\boldsymbol{v}_f\boldsymbol{q} \tag{12}$$

根据相关性评分即可找到与查询最相关的文件。

3 模型和设计目标

3.1 方案模型

本文方案的主要目标是解决边缘网络中的共享缓存问题，利用距离用户更近的边缘节点来存储和管理网络中各运营商的公共数据，降低数据访问时延，提高宽带利用率，从而增强用户体验。系统模型如图4所示，主要由可信任的云服务器、用户和边缘网络中的运营商三部分构成。首先各网络运营商通过多方PSI协议计算出公共数据项并将其发送给可信任的云服务器，用户通过关键字检索数据时先访问云服务器，云服务器利用多关键字检索Top-k算法可以将公共数据项中相关性评分最高的前k个文档返回给用户。

3.2 安全模型

本文采用半诚实安全模型，且假设方案中的通信均采用经过认证的安全可信信道如TLS等。模型包括$n(n>2)$个参与方、半诚实敌手A以及腐败的参与方集合$C(t\leqslant n-2)$。敌手A能获取集合C内各参与方的相关信息并进行分析从而获得潜在数据，但不能干预诚实参与方的正常协议执行。作为安全多方计算框架下的协议，PSI通常采用理想−现实模型[30]进行协议的安全性证明。

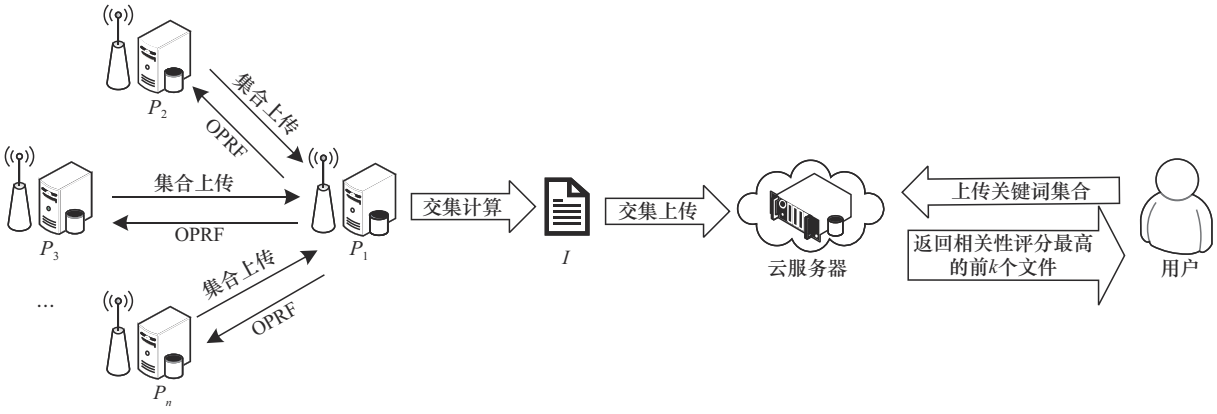

图4　系统模型

理想−现实模型中以F为PSI协议Π的功能函数，f_i为构建F的安全多方计算底层功能函数。如果在f_i−混合模式下模拟者(Sim)运行协议Π的攻击者视角Sim_F与直接运行协议Π的攻击者视角View_{Π}不可区分，则证明协议Π能够安全实现功能函数F，即证明了协议的安全性。

1) 混合模式。如果协议Π_i安全实现功能函数f_i，协议Π利用协议Π_i安全实现功能函数F，则称协议Π在f_i−混合模式下安全实现功能函数F。

2) 理想−现实模型。P_i有私有输入集合$X_i=\left\{x_1^i,x_2^i,\cdots,x_m^i\right\}$，输出集合$Y_i=\left\{y_1^i,y_2^i,\cdots,y_m^i\right\},i\in[1,n]$。$\Pi$是PSI协议，$F$是PSI功能函数，$C$为被腐败的参与方集合，安全参数$k$。

现实模型$\mathrm{View}_{\Pi}\left(k,C;X_1,X_2,\cdots,X_n\right)$：$n$个参与方$P_i$分别输入$X_i$到协议$\Pi$，参与方共同计算$\Pi\left(X_1,X_2,\cdots,X_n\right)\rightarrow\left(Y_1,Y_2,\cdots,Y_n\right)$，令$\mathrm{View}_i$作为$P_i$的最终视图$\left\{\mathrm{View}_i|i\in C\right\}$。

理想模型$\mathrm{Sim}_F\left(k,C;X_1,X_2,\cdots,X_n\right)$：定义协议$\Pi$的安全性要求$S$，$n$个参与方$P_i$分别输入$X_i$到协议$\Pi$计算$F\left(X_1,X_2,\cdots,X_n\right)\rightarrow\left(Y_1,Y_2,\cdots,Y_n\right)$。模拟者Sim模仿现实模型中敌手视图$\mathrm{Sim}\left(C,\left\{\left(X_i,Y_i\right)|i\in C\right\}\right)$。

3) 安全性。当$\mathrm{View}_{\Pi}\left(k,C;X_1,X_2,\cdots,X_n\right)$与$\mathrm{Sim}_F\left(k,C;X_1,X_2,\cdots,X_n\right)$的输出在安全参数$k$下不可区分时，则认为协议$\Pi$在安全性要求$S$下安全实现功能函数$F$。

3.3　设计目标

对于本文提出的多方共享缓存隐私保护方案，有以下几个设计目标。

1) 方案需提供最基本的机密性保障，能保证网络中各运营商的数据隐私安全。

2) 方案能为运营商客户端提供安全且高效的交集运算。

3) 方案能为用户提供安全的资源访问，保证用户检索结果的准确性。

4　方案描述

4.1　方案流程

整个方案主要分为2个阶段：多方PSI协议阶段和多关键字检索阶段。在多方PSI协议阶段，网络中的运营商作为协议的参与方P_i，各自拥有数据集合X_i。其中，P_1作为领导者，同时也作为发送方与其余参与方调用OPRF。随后P_1将计算出的交集数据上传到可信的第三方云服务器上，用户通过上传一组关键字集合进行查询。云服务器通过TF-IDF加权法和向量空间模型来计算用户关键字与文件的相关性评分，最后调用Top-k算法得到相关性评分最高的前k个文件并将结果返回给用户，方案流程如图5所示。

本文方案中云服务器主要承担2个工作：一是接收多方PSI协议的公共数据项；二是负责处理用户的关键字查询请求，其中包括接收用户的关键字集合、计算公共数据项中各文档的相关性评分、建立索引、调用Top-k算法等关键步骤。在这个过程中，云服务器接触的均是明文数据，如果云服务器不可信，那么公共数据和用户的查询数据都会有泄漏风险，因此需要可信的云服务器来保障隐私数据的安全性。

4.2　多方PSI协议阶段

协议主要基于多点OPRF和零密钥共享，通过

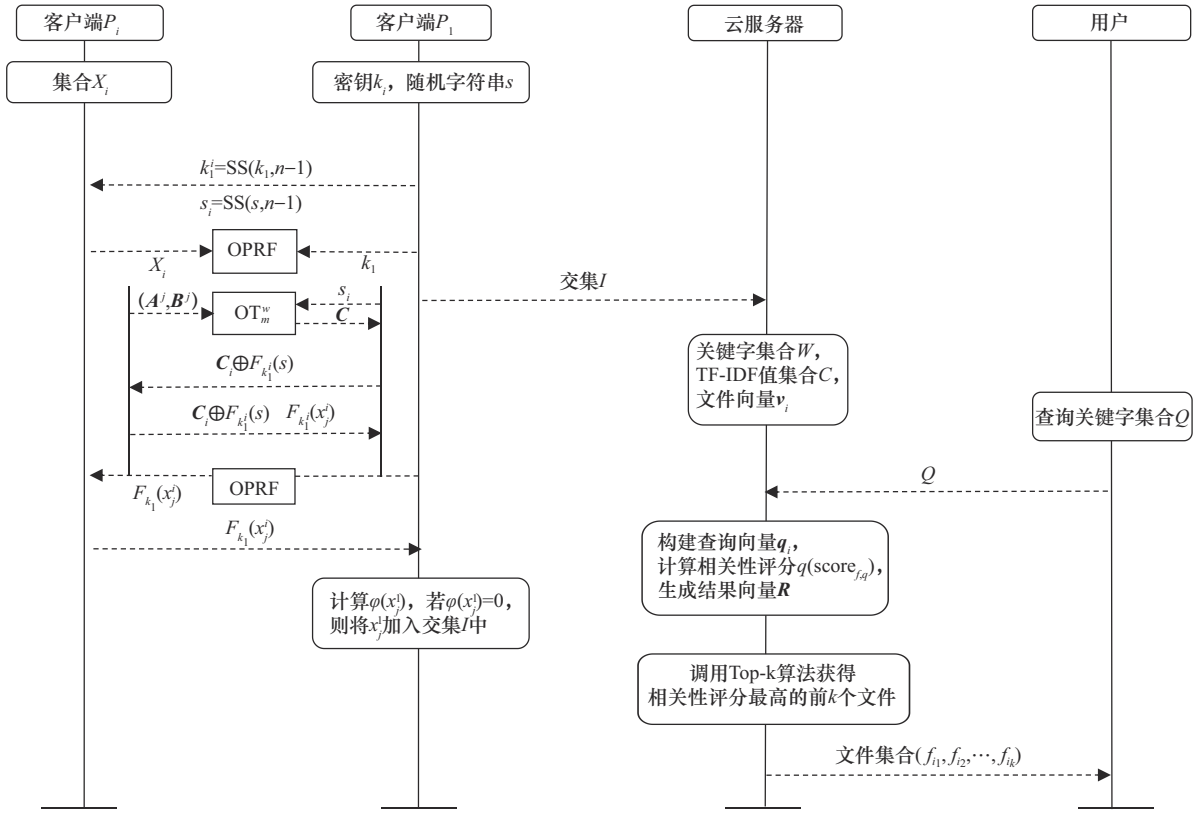

图5　方案流程

OT扩展协议来构造多点OPRF。方案描述如下。

参数：n个参与方$P_1,P_2,\cdots,P_n$，各参与方持有大小为m的数据集$X_i=\{x_1^i,x_2^i,\cdots,x_m^i\},i\in[1,n]$，安全参数$\lambda$、$k$，协议参数$w$，抗碰撞的哈希函数$H:\{0,1\}^*\to\{0,1\}^w$，不经意伪随机函数$F:\{0,1\}^*\to\{0,1\}^w$，理想的$\mathrm{OT}_m^w$协议。

1) 初始化阶段

① P_1随机选取密钥k_1，并用$(n-1,n-1)$秘密共享方案将其分为$n-1$份发送给$P_i,i\in[2,n]$，有$k_1=\oplus_{i=2}^n k_1^i$。

② P_1作为发送方S与其他参与方P_i调用OPRF。

P_1输入密钥k_1。

P_i输入集合$X_i=\{x_1^i,x_2^i,\cdots,x_m^i\}$。

P_i输出$F_{k_1}(x_j^i),j\in[1,m]$。

2) OPRF阶段

① P_1生成w长的随机字符串$s=(r_1,r_2,\cdots,r_w)$，用$(n-1,n-1)$秘密共享方案分成$n-1$份，有$s=\oplus_{i=2}^n s_i$。

② P_i生成$m\times w$矩阵$\boldsymbol{A}$和$\boldsymbol{B}$，其中矩阵$\boldsymbol{A}$的行向量为$\boldsymbol{A}_i=F_{k_1^i}(x_j^i)$，矩阵$\boldsymbol{B}$的行向量为$\boldsymbol{B}_i=\boldsymbol{A}_i\oplus H(x_j^i)$。

③ P_1作为接收方与P_i调用OT_m^w协议。

P_1输入随机字符串s_i，P_i输入矩阵$\boldsymbol{A}$和$\boldsymbol{B}$的列向量$(\boldsymbol{A}^j,\boldsymbol{B}^j)$。

P_1输出矩阵$\boldsymbol{C}$，其中行向量为

$$\boldsymbol{C}_i=F_{k_1^i}(x_j^i)\oplus\left[H(x_j^i)\wedge s_i\right] \tag{13}$$

P_1计算$\boldsymbol{C}_i\oplus F_{k_1^i}(s)$并发送给$P_i$，$P_i$收到后计算

$$\boldsymbol{C}_i\oplus F_{k_1^i}(s)\oplus F_{k_1^i}(x_j^i) \tag{14}$$

并将其作为OPRF的输出$F_{k_1}(x_j^i)$发送给P_1。

3) PSI阶段

P_1计算

$$\varphi(x_j^i)=F_{k_1}(s)\oplus\left[H(x)\wedge s\right]\oplus\left[\oplus_{i=1}^n F_{k_1}(x_j^i)\right] \tag{15}$$

如果$\varphi(x_j^i)=0$，则P_1将x_j^1添加到交集I中。

4.3　多关键字检索阶段

该阶段主要使用多关键字检索Top-k算法。运营商将得到的公共数据集发送给可信任的云服务

器，首先在云服务器端对文件集合进行预处理，使用IR（information retrieval）社区的词干处理技术从文件集合中构建可搜索索引I。用户输入关键词后，云服务器端会计算关键词和文件的相关性评分并以此对文件进行排名，最后将相关性评分最高的前k个文件返回给用户。算法描述如下。

1) 初始化阶段

① 数据拥有者从每个文件中都提取l个关键词组成集合$W=\left(w_1,w_2,\cdots,w_l\right)$，同时输出关键词在文件集合中的TF和IDF值的集合$C=\left(f_1,f_2,\cdots,f_n\right)$。

② 对于$\exists f_i\in C$，数据拥有者都建立一个$(l+1)$维向量$\boldsymbol{v}_i=\left(\mathrm{id}_i,t_{i,1},t_{i,2},\cdots,t_{i,l}\right)$，其中

$$t_{i,j}=\mathrm{tf}\text{ - }\mathrm{idf}_{w_j f_i},j\in[1,l] \tag{16}$$

可搜索索引$I=\left\{\boldsymbol{v}_i\right\},i\in[1,n]$。

2) 检索阶段

① 用户检索文件时先输入关键词集合$Q=\left\{w'_1,w'_2,\cdots,w'_s\right\}$，云服务器通过用户和本地的关键词集合构建查询向量$\boldsymbol{q}_i=\left(m_1,m_2,\cdots,m_l\right)$，构建规则为当$w_i\in Q$时，$m_i=1$；否则$m_i=0,i\in[1,l]$。

② 云服务器为索引中的每一个文件向量$\boldsymbol{v}_i$计算内积作为该文件与关键词的相关性得分，如式(17)所示。

$$q\left(\mathrm{score}_{f,q}\right)=\boldsymbol{v}_i\boldsymbol{q}_i \tag{17}$$

并生成结果向量$\boldsymbol{R}=\left\{\left(\mathrm{id}_1,q_1\right),\left(\mathrm{id}_2,q_2\right),\cdots,\left(\mathrm{id}_n,q_n\right)\right\}$。

③ 调用Top-k算法获得结果向量中相关性评分最高的前k个文件的标识符$\left(i_1,i_2,\cdots,i_k\right),i\in[1,n]$，然后云服务器将标识符对应的文件集合$\left(f_{i_1},f_{i_2},\cdots,f_{i_k}\right)$返回给用户。

本文的Top-k算法采用最小堆数据结构高效实现Top-k查询功能。其核心机制在于动态维护一个容量为k的最小堆作为候选集，堆顶元素始终表示当前第k大的值。Top-k算法采用动态维护策略，首先初始化空堆后遍历输入序列，当堆元素数量小于k时直接执行堆插入操作。当堆已满载时，通过堆顶比较操作仅在新元素值大于等于堆顶值时才触发堆顶替换操作并执行堆结构调整，该操作能确保堆内始终保留遍历过程中最大的k个元素。最终阶段对堆内元素按值升序排序并提取对应索引序列，输出结果严格满足元素值升序排列的索引列表。算法利用堆的$O(\mathrm{lb}\,k)$插入和删除特性，将时间复杂度优化至$O(n\mathrm{lb}\,k)$，在k远小于n的大规模数据处理场景中展现出优异的计算效率。

5　方案分析

5.1　安全性分析

1) OPRF协议安全性

本文的OPRF协议在半诚实安全模型下是安全的。

定理1　本文的OPRF协议对半诚实的服务器是安全的。

证明　被腐败的服务器的模拟器$\mathrm{Sim}^{\mathrm{S}}_{\mathrm{oprf}}$的行为如下。

情况1　在OPRF协议的执行过程中，S与R调用OT^w_m协议步骤3)，S诚实地生成OT^w_m协议的发送者视图并以OT协议发送者的身份运行OT协议模拟器。这种情况与现实世界相同，S运行OT模拟器来生成OT发送者的模拟视图，由于OT协议的安全性，这种情况在计算上与现实世界无法区分，证毕。

定理2　本文的OPRF协议可以抵抗半诚实客户端的合谋攻击。

证明　被腐败的客户端的模拟器$\mathrm{Sim}^{\mathrm{C}}_{\mathrm{corrupted\text{ - }oprf}}$的行为如下。

情况2　在OPRF协议中，S与R调用OT^w_m协议步骤3)，S诚实地生成OT^w_m协议的发送者视图并以OT协议接收者的身份运行OT协议模拟器。由于OT协议的安全性，这种情况在计算上与现实世界无法区分。

情况3　S随机选取PRF F的密钥k'并计算$\boldsymbol{C}_i\oplus F_{k'}(s)$步骤5)，S用随机字符串替换$\boldsymbol{C}_i\oplus F_{k'}(s)$。$\boldsymbol{C}_i$是$\mathrm{OT}^w_m$协议的输出，本身具有随机性，同时$F_{k'}(s)$也是伪随机函数的输出，所以被替换的值对于发送方来说本身就是随机的。由于OT协议和伪随机函数的安全性，故情况3不会对协议的正确性造成影响，证毕。

2) 多方PSI协议安全性

定理3　本文的多方PSI协议在半诚实安全模型和任意数量（$t\leqslant n-2$）的合谋攻击下是安全的。

引理1　本文的多方PSI协议对半诚实的服务器是安全的。

证明 被腐败的服务器的模拟器$\mathrm{Sim}^{\mathrm{S}}_{\text{corrupted-oprf-mpsi}}$的行为如下。

情况4 在OPRF协议阶段，模拟器用OPRF的理想功能替换OPRF实例。由于OPRF协议的安全性已被证明，因此该情况与现实世界无法区分，证毕。

引理2 本文的多方PSI协议可以在半诚实安全模型下抵抗$t(t \leqslant n-2)$个参与方的合谋攻击。

情况5 领导者P_1诚实。

证明 协议中假定领导者P_1和至少一个客户端是诚实的，其他被腐败的客户端的模拟器$\mathrm{Sim}^{\mathrm{C}}_{\text{corrupted - mpsi}}$行为如下。

因为P_1是诚实的参与方，所以腐败方不能从OPRF阶段获得除输出外的任何消息，并且多关键字检索阶段不涉及腐败的参与方，故协议是安全的。

其次，在秘密共享阶段，P_1将秘密共享给其余的$n-1$个参与方，但这些参与方中至多只能有$n-2$个腐败方，他们并不能恢复P_1共享的密钥和随机字符串，故协议是安全的，证毕。

情况6 领导者P_1腐败。

证明 协议中至少有一个诚实的客户端，其他被腐败的客户端的模拟器$\mathrm{Sim}^{\mathrm{C}}_{\text{corrupted - mpsi}}$的行为如下。

在OPRF阶段，$\mathrm{Sim}^{\mathrm{C}}_{\text{corrupted - mpsi}}$调用OPRF理想功能模拟输出。对于每个不在交集中的元素x_j^1，用随机字符替换$\varphi\left(x_j^i\right)$。

下面证明该模拟与现实世界的不可区分性。

① 该混合除了OPRF实例被替换为理想功能外与现实世界相同，而OPRF的安全性已经被证明，故该模拟与现实世界不可区分。

② 对于每个不在交集中的元素x_j^1而言，$\varphi\left(x_j^i\right)$是伪随机函数输出的随机值，故该模拟与现实世界不可区分。

证毕。

定理4 本文的多方PSI协议在半诚实安全模型下能够抵抗中间人攻击。

证明 半诚实敌手A的模拟器$\mathrm{Sim}_{\mathrm{A}}$的行为如下。

首先$\mathrm{Sim}_{\mathrm{A}}$将调用OT协议理想功能模拟输出；半诚实敌手A窃听OPRF的传输矩阵；用随机响应代替随机预言机H。

下面证明该模拟与现实世界不可区分。

① 由于协议的整个通信过程均使用经过认证的可信信道传输，故敌手A无法获取或篡改OPRF的输入输出数据。

② 由于OT协议的安全性，中间人敌手A无法获知OT协议的输入和选择比特的相关信息，故无法篡改OT协议的输出，使$\mathrm{Sim}_{\mathrm{A}}$的模拟世界与现实世界不可区分。

③ 在2.2节的步骤4)中，由于随机预言机下的$H\left(j,\cdot\right)$输出均匀随机，使$\mathrm{Sim}_{\mathrm{A}}$的模拟世界与现实世界不可区分。

证毕。

3) 多关键字检索Top-k算法安全性

多关键字检索阶段主要是用户与云服务器的交互过程，用户上传关键字集合，云服务器通过计算后给用户返回相关的文件。整个过程双方均通过可信信道进行通信，且方案中引入的是可信的第三方云服务器，能够保障运营商及用户数据的安全性。

5.2 性能分析

5.2.1 功能分析

本节主要是对方案的整体功能进行分析，选取了与边缘计算最相关的文献[8]和文献[9]进行比较。

首先是方案的安全性保证，如表1所示。文献[8]和本文方案均采用半诚实安全模型，文献[9]则采用安全性保证更强的单边恶意安全模型。文献[8]提出了2种MPCCache架构，一种是采用树状拓扑的去中心化模型，下面简称为文献[8](a)，能抵抗$n-1$个客户端的合谋攻击，但计算和通信开销较高；另一种是服务器辅助模型，以下简称为文献[8](b)，在非共谋服务器的前提下能抵抗$n-2$个客户端的合谋攻击，效率相较于前者有所提升，但在参与方数量方面仍有局限。文献[9]和本文方案都采用领导者协调的星形拓扑结构，相较于文献[8]更适用于边缘环境。且本文方案设计更加简易，突破了领导者的开销瓶颈，领导者和客户端的开销仅与参与方集合大小有关，使其更易于扩展到多方环境中。

其次从方案整体而言，文献[8]完整设计了包含两阶段流程的边缘缓存框架：1) 通过OPPRF和零共享技术保密计算公共数据项的价值总和；2) 使用Barcher排序网络选出最高价值的缓存项。

表1　　各方案功能对比

方案	安全模型	拓扑结构	抗合谋客户端数量/个	扩展性	共享缓存策略	功能
文献[8](a)	半诚实	树状拓扑	$n-1$	较差	基于文档访问频率	为用户返回访问频率最高的前k个文件
文献[8](b)	半诚实	星形拓扑	$n-2$	可扩展	基于文档访问频率	为用户返回访问频率最高的前k个文件
文献[9]	单边恶意	星形拓扑	$n-2$	可扩展，但领导者开销可能成为瓶颈	基于文档访问频率	为用户返回访问频率最高的前k个文件
本文方案	半诚实	星形拓扑	$n-2$	易扩展	基于文档与用户关键字的相关性评分	为用户返回相关性评分最高的前k个文件

亮点在于支持多方协作，且首次实现交集项关联值的隐私求和计算，但方案并没有给出如何计算单个数据项的关联值的详细说明。方案整体侧重架构的通用性但整体效率还有待提升。文献[9]的重点在于高效地计算集合交集，使用PaXoS数据结构和OT扩展技术优化了通信效率，使用和文献[8]相似的共享缓存框架实现高频数据识别。本文方案在实现了高效交集计算的前提下设计了多方协作共享缓存框架，文献[8]和文献[9]共享缓存策略都是基于运营商定义的访问频率，而本文方案增加了对文档的处理过程，支持语义相似度计算，通过TF-IDF加权法和向量空间模型计算文档和关键字的相关性评分，最后利用多关键字检索Top-k算法为用户提供了更精准的内容匹配。

5.2.2　多方PSI协议

本节将从多方PSI协议部分的理论计算和通信开销以及协议的实际运行效率两方面分别与其他协议进行对比。

首先，本文的多方PSI协议部分采用OT扩展协议构建多点OPRF协议，主要依赖于对称加密和矩阵运算对数据进行并行化处理，使协议在保证数据隐私和协议安全性的前提下展现出了较高的可扩展性。该协议实现了计算与通信开销之间的平衡，适用于多个参与方和大规模数据集的场景，在边缘网络环境中性能表现良好。其次，在多关键字检索Top-k算法部分使用诚实的云服务器作为可信第三方，能有效保护数据拥有者的隐私信息以及用户的关键字集合。整体协议实现了半诚实模型下的安全，且能抵抗一定数量（$t \leqslant n-2$）参与方的合谋攻击。

1)理论分析

本文的多方PSI协议主要依赖于矩阵变换和按位运算，其中P_1作为领导者，其余参与方当作客户端参与协议。实际上，P_1承担了协议大部分的通信开销，而计算开销则是均匀分布在各参与方。

① 计算开销

该协议主要分为3个阶段：初始化阶段、OPRF阶段和PSI阶段。协议共n个参与方，各参与方数据集合大小为m。为了方便计算将PRF和哈希函数的计算时间看作常数。在初始化阶段，P_1调用秘密共享算法为其余$n-1$个参与方分发共享密钥和随机字符串，其计算开销为$O(n-1)$。随后开始调用OPRF协议，这也是该协议计算开销的主要组成部分，总体近似为$O\big((n-1)m\lambda\big)$。最后的PSI阶段，参与方P_1要通过P_i返回的$n-1$个m大小的集合计算出交集元素，计算开销为$O(m\lambda)$。由此，整个PSI协议部分的计算开销为$O(n-1)+O(m\lambda)+O\big((n-1)m\lambda\big)$，近似为$O(nm\lambda)$。

② 通信开销

协议共n个参与方，各参与方数据集合大小为m。在初始化阶段，P_1要通过秘密共享为其他$n-1$个参与方分发密钥，以安全参数λ作为密钥长度，其中w可视为依赖于安全参数λ的值，则该过程的通信开销为$O\big((n-1)\lambda\big)$。同理，P_1将w长的随机字符串分发给其他$n-1$个参与方的通信开销为$O\big((n-1)w\big)$。接下来进入OPRF阶段，P_1与P_i调用OPRF，其中主要的通信开销是OTE协议中的OT交互过程，整个过程中共需要进行$n-1$轮通信，总的通信开销为$O\big((n-1)m\lambda\big)$。最后的PSI阶段中通信开销主要是参与方P_i的输出，近似为

$O((n-1)m\lambda)$。故PSI协议部分的总体通信开销为$O((n-1)\lambda)+O((n-1)w)+O((n-1)m\lambda)$，近似为$O((n-1)m\lambda)$。

表2中将本文协议与最近的一些MPSI（mulity private set intersection）协议进行了比较，其中文献[8]首次将PSI技术引入边缘计算场景中且提供了2种MPCCache框架，分别是去中心化的MPCCache结构和服务器辅助的MPCCache结构。这2种协议架构均采用半诚实的安全模型，下面对MPSI协议部分的理论开销进行了分析比较。其中n为参与方的数量，m为各参与方数据集的大小，k为散列函数的数量，λ为协议的安全参数。本文的MPSI协议中，领导者的通信和计算开销分别为$O((n-1)m\lambda)$和$O(m\lambda)$，而客户端的通信和计算开销均为$O(m\lambda)$，仅领导者的通信开销与参与方数量呈线性关系，客户端的开销只与数据集合大小有关。

如表2所示，与文献[31]相比本文协议在相同的安全模型下拥有更好的计算效率。文献[14]采用增强的半诚实安全模型，虽然实现了更高的安全性保证，但该协议提供了更好的计算和通信效率。文献[32]和文献[9]都是恶意模型下的多方PSI协议，相比之下虽然本文协议的安全性保证较弱但在协议效率方面更有优势。其中文献[9]采用的是单边恶意模型而非标准的恶意模型，但协议中领导者的计算效率不佳。综上本文协议有良好的计算和通信效率，且实现了协议效率与安全性之间的平衡。

下面对文献[8]的多方合作共享缓存框架中多方PSI协议部分的开销进行分析。文献[8](a)采用了类似星形拓扑的网络结构，使各参与方的计算和通信开销较为平均，其中领导者和客户端的通信开销分别为$O(mn(\lambda+\text{lb}\,n)+m^2)$和$O(m^2(\lambda+\text{lb}\,n))$，参与方的计算开销均为$O(m^2(\lambda+\text{lb}n))$。可以看出，无论在计算开销或是通信开销上本文协议在大集合和多参与方的场景下都具有明显优势。为了进一步优化协议效率，文献[8](b)又引入了服务器辅助模型。该模型假定2个非共谋的参与方为领导者，其余参与方为客户端，协议的大部分开销由2个领导者承担。如表1所示，该模型下客户端的计算和通信开销分别近似为$O(mn(\lambda+\text{lb}\,n))$和$O(mn(\lambda+\text{lb}\,n)+m)$，与文献[8](a)相比有显著改善。在通信开销方面，由于本文协议客户端的通信开销仅与参与方集合大小有关，故综合表现最优。

2)实验分析

为了更直观地反映协议效率，本文实验对比表1中提到的几个方案。经过分析文献[8](a)随参与方数量和集合大小的增加会产生远大于其他协议的时间开销，故这里只与文献[8](b)进行比较。实验采用文献[9]设置的环境，服务器采用Windows 10，Intel(R) Core (TM) i5-8250U CPU @1.6 GHz 1.80 GHz，8.00 GB RAM，编程语言采用C++。利用单机模拟多方环境，通过OpenSSL1.1.1和GMP6.2.1来实现相关的密码学操作，方案中OTE协议的实现结合了IKNP协议框架和文献[10]中的相关工作。假设参与方数量$n=100$，安全参数$\lambda=128$，集合大小$m=2^{10},2^{11},2^{12},2^{13},2^{14}$，各协议运行总时间如图6所示。

表2　　**多方PSI协议效率对比**

协议	通信开销		计算开销		安全模型
	领导者	客户端	领导者	客户端	
文献[31]	$O(mn\lambda)$	$O(m\lambda k)$	$O(mn\lambda)$	$O(mn\lambda)$	半诚实
文献[14]	$O(mn\lambda k\,\text{lb}\,n)$	$O(mn\lambda k\,\text{lb}\,n)$	$O(mn\lambda k)$	$O(mn\lambda k)$	增强的半诚实
文献[32]	$O((n^2+mn)\lambda)$	$O(m\lambda)$	$O(mn\,\text{lb}\,m)$	$O(m\,\text{lb}\,m)$	恶意
文献[9]	$O(mn\lambda)$	$O(m\lambda)$	$O(mn\lambda)$	$O(m\lambda)$	单边恶意
文献[8](a)	$O(mn(\lambda+\text{lb}\,n)+m^2)$	$O(m^2(\lambda+\text{lb}\,n))$	$O(m^2(\lambda+\text{lb}\,n))$	$O(m^2(\lambda+\text{lb}\,n))$	半诚实
文献[8](b)	$O(mn(\lambda+\text{lb}\,n)+m)$	$O(m(\lambda+\text{lb}\,n))$	$O(mn(\lambda+\text{lb}\,n))$	$O(m(\lambda+\text{lb}\,n))$	半诚实
本文协议	$O((n-1)m\lambda)$	$O(m\lambda)$	$O(m\lambda)$	$O(m\lambda)$	半诚实

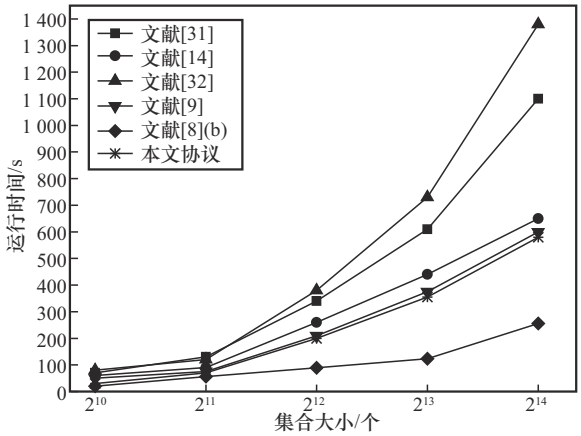

图6　协议运行时间随集合大小变化情况

从图6的实验结果中可以看出，当参与方数量不变时，虽然各个协议的整体时间开销都随着集合大小呈线性增长，但当集合大小大于2^{11}时，文献[31]和文献[32]的时间开销的增长率显著高于本文协议，且整体上具有最慢的时间增长率。文献[8](b)由于引入了服务器辅助模型，有2个非共谋的参与方作为领导者，承担了大部分的协议开销，同时通过客户端的并行计算进一步提升了协议效率，故而在协议的时间开销上更有优势，但本文采用单领导者星形拓扑架构更适配于边缘网络拓扑。

此外，本文还进一步考虑了参与方数量对协议运行时间的影响。假设各方集合大小$m = 2^{11}$，安全参数$\lambda = 128$，参与方数量分别设置为$n = 10^1,10^2,10^3,10^4$。各协议整体运行时间变化如图7所示。实验表明当集合大小固定时，本文协议与文献[8](b)具有相近的时间开销和涨幅，且当参与方数量小于10^3时本文协议的运行效率更优。总体而言，相较于其他协议，参与方数量的增加对本文协议整体的时间开销影响最小。

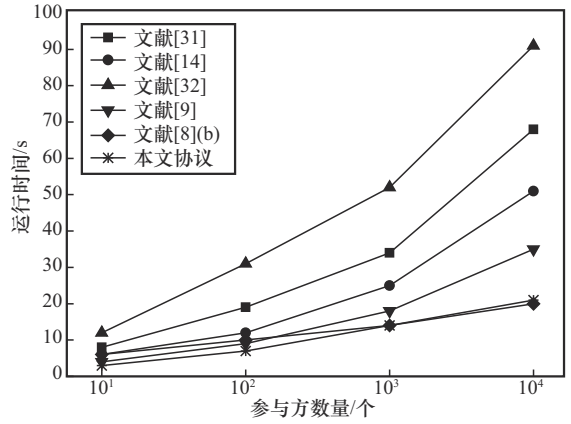

图7　协议运行时间随参与方数量变化情况

5.2.3　多关键字检索 Top-k 算法

本节对多关键字检索Top-k算法的功能及算法性能进行分析。

文献[8]中提出的MPCCache架构规定每个文件都有其标识符和对应的关联值，最后通过各文件的关联值大小来确定该文件被访问的频率，即k优先级算法中的前k个文件。方案中的关联值贯穿整个协议，但并没有给出该关联值的定义和计算，相比之下本文方案考虑了对文件的处理过程，通过TF-IDF加权法和空间向量模型[29]来计算用户关键词与文件的相关性评分，能为用户返回更准确、更相关的搜索结果。MPCCache架构关注的是文件的访问频率，而本文则更关注文件与用户请求的相关性，同时具有较高的可扩展性，更适用于多方环境。

用n表示文件数量，每个文件大小不超过λ，用户输入s个关键字，客户端从文件中提取l个关键字，其中$s \leqslant l$，为了方便分析这里将关键字大小视为常数。分析可得协议总的计算开销近似于$O(l + n + n\,\mathrm{lb}\,k)$，而通信开销近似于$O(l + n + \lambda k)$。由此可以看出与算法运行时间最相关的2个因素为总文件数量和返回的文件数量。为了更直观地分析协议效率，本文通过实验得出了算法运行时间与文件数量n和参数k之间的关系，如图8和表3所示。根据表3和图8可以看出，协议的运行时间与文件数量n和参数k呈线性增长关系，公共文件和返回的文件数目越大，算法的运行时间越长。

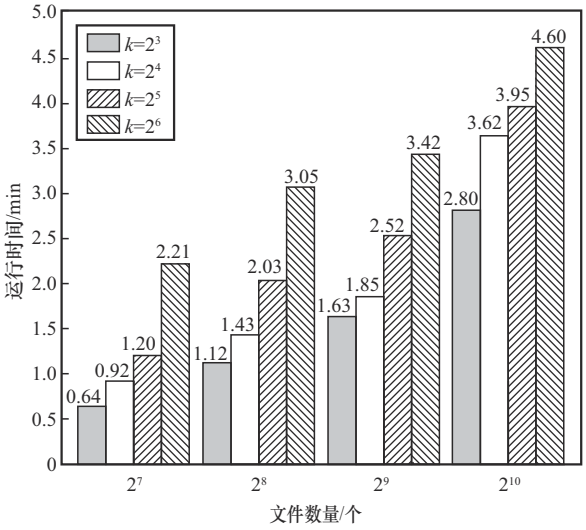

图8　算法运行时间随文件数量n和参数k的变化情况

表3 多关键字检索 Top-k 算法运行时间对比

文件数量/个	运行时间/min			
	$k = 2^3$	$k = 2^4$	$k = 2^5$	$k = 2^6$
2^7	0.64	0.92	1.20	2.21
2^8	1.12	1.43	2.03	3.05
2^9	1.63	1.85	2.52	3.42
2^{10}	2.80	3.62	3.95	4.60

6 结束语

本文针对通信网络中数据流量激增导致的网络带宽拥塞问题设计了一种适用于边缘计算场景的基于 PSI 的多方共享缓存隐私保护方案，分析表明该方案不仅实现了半诚实模型下的安全，还能够抵抗一定数量参与方的合谋攻击。本文采用基于 Shamir 秘密共享和 OTE 协议的多点 OPRF 协议，在此基础上构建了适用于多方环境的 PSI 协议，并且借助可信的云服务器和多关键字检索 Top-k 算法完成用户的资源访问，通过在非高峰时期将各运营商的公共数据项存储到共享缓存中，有效提高了网络带宽利用率，增强了用户体验。与现有的协议相比，本文协议在同等安全保证下平衡了计算和通信开销，且能为用户返回更准确的请求结果。

参考文献：

[1] PASCHOS G S, IOSIFIDIS G, TAO M X, et al. The role of caching in future communication systems and networks[J]. IEEE Journal on Selected Areas in Communications, 2018, 36(6): 1111-1125.

[2] BASTUG E, BENNIS M, DEBBAH M. Living on the edge: the role of proactive caching in 5G wireless networks[J]. IEEE Communications Magazine, 2014, 52(8): 82-89.

[3] 张佳乐, 赵彦超, 陈兵, 等. 边缘计算数据安全与隐私保护研究综述[J]. 通信学报, 2018, 39(3): 1-21.
ZHANG J L, ZHAO Y C, CHEN B, et al. Survey on data security and privacy-preserving for the research of edge computing[J]. Journal on Communications, 2018, 39(3): 1-21.

[4] LI J W, LI J, CHEN X F, et al. Privacy-preserving data utilization in hybrid clouds[J]. Future Generation Computer Systems, 2014, 30: 98-106.

[5] BAHRAMI M, SINGHAL M. A light-weight permutation based method for data privacy in mobile cloud computing[C]//Proceedings of the 2015 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering. Piscataway: IEEE Press, 2015: 189-198.

[6] PASUPULETI S K, RAMALINGAM S, BUYYA R. An efficient and secure privacy-preserving approach for outsourced data of resource constrained mobile devices in cloud computing[J]. Journal of Network and Computer Applications, 2016, 64: 12-22.

[7] 沈剑, 周天祺, 王晨, 等. 面向边缘计算的隐私保护密钥分配协议[J]. 网络与信息安全学报, 2021, 7(1): 93-100.
SHEN J, ZHOU T Q, WANG C, et al. Privacy protection key distribution protocol for edge computing[J]. Chinese Journal of Network and Information Security, 2021, 7(1): 93-100.

[8] NGUYEN D T, TRIEU N. MPCCache: privacy-preserving multi-party cooperative cache sharing at the edge[C]//International Conference on Financial Cryptography and Data Security. Berlin: Springer, 2022: 80-99.

[9] ZHANG J, YANG L, TANG Y L, et al. A novel edge cache-based private set intersection protocol via lightweight oblivious PRF[J]. Entropy, 2023, 25(9): 1347.

[10] KOLESNIKOV V, KUMARESAN R, ROSULEK M, et al. Efficient batched oblivious PRF with applications to private set intersection[C]//Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2016: 818-829.

[11] KOLESNIKOV V, MATANIA N, PINKAS B, et al. Practical multiparty private set intersection from symmetric-key techniques[C]//Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2017: 1257-1272.

[12] CHASE M, MIAO P H. Private set intersection in the Internet setting from lightweight oblivious PRF[C]//Advances in Cryptology-CRYPTO 2020. Berlin: Springer, 2020: 34-63.

[13] KAVOUSI A, MOHAJERI J, SALMASIZADEH M. Efficient scalable multi-party private set intersection using oblivious PRF[C]//International Workshop on Security and Trust Management. Berlin: Springer, 2021: 81-99.

[14] INBAR R, OMRI E, PINKAS B. Efficient scalable multiparty private set-intersection via garbled bloom filters[C]//International Conference on Security and Cryptography for Networks. Berlin: Springer, 2018: 235-252.

[15] DONG C Y, CHEN L Q, WEN Z K. When private set intersection meets big data: an efficient and scalable protocol[C]//Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security. New York: ACM Press, 2013: 789-800.

[16] HAZAY C, VENKITASUBRAMANIAM M. Scalable multi-party private set-intersection[C]//Public-Key Cryptography-PKC 2017. Berlin: Springer, 2017: 175-203.

[17] RINDAL P, ROSULEK M. Improved private set intersection against malicious adversaries[C]//Advances in Cryptology-EUROCRYPT 2017. Berlin: Springer, 2017: 235-259.

[18] ZHANG E, LIU F H, LAI Q Q, et al. Efficient multi-party private set intersection against malicious adversaries[C]//Proceedings of the 2019 ACM SIGSAC Conference on Cloud Computing Security Workshop. New York: ACM Press, 2019: 93-104.

[19] BEN-EFRAIM A, NISSENBAUM O, OMRI E, et al. PSImple: practical multiparty maliciously-secure private set intersection[C]//Proceedings of the 2022 ACM on Asia Conference on Computer and Communications Security. New York: ACM Press, 2022: 1098-1112.

[20] GARIMELLA G, PINKAS B, ROSULEK M, et al. Oblivious key-value stores and amplification for private set intersection[C]//Advances in Cryptology-CRYPTO 2021. Berlin: Springer, 2021: 395-425.

[21] NEVO O, TRIEU N, YANAI A. Simple, fast malicious multiparty private set intersection[C]//Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM Press, 2021: 1151-1165.

[22] JIANG Z T, GUO X X, YU T, et al. Private set intersection based on lightweight oblivious key-value storage structure[J]. Symmetry, 2023, 15(11): 2083.
[23] WU M L, YUEN T H, CHAN K Y. O-ring and K-star: efficient multi-party private set intersection[C]//Proceedings of the 33rd USENIX Security Symposium. USENIX Association, 2024: 6489-6506.
[24] WEI L F, LIU J H, ZHANG L, et al. Efficient multi-party private set intersection protocols for large participants and small sets[J]. Computer Standards & Interfaces, 2024, 87: 103764.
[25] LV S Y, WEI Y, JIA J Y, et al. New approach for efficient malicious multiparty private set intersection[J]. Information Sciences, 2024, 678: 120995.
[26] RABIN M O. How to exchange secrets with oblivious transfer[R]. 2005.
[27] BEAVER D. Correlated pseudorandomness and the complexity of private computations[C]//Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing. New York: ACM Press, 1996: 479-488.
[28] ISHAI Y, KILIAN J, NISSIM K, et al. Extending oblivious transfers efficiently[C]//Advances in Cryptology-CRYPTO 2003. Berlin: Springer, 2003: 145-161.
[29] YU J D, LU P, ZHU Y M, et al. Toward secure multikeyword top-k retrieval over encrypted cloud data[J]. IEEE Transactions on Dependable and Secure Computing, 2013, 10(4): 239-250.
[30] EVANS D, KOLESNIKOV V, ROSULEK M. A pragmatic introduction to secure multi-party computation[J]. Foundations and Trends® in Privacy and Security, 2017, 2(2/3): 70-246.
[31] KAVOUSI A, MOHAJERI J, SALMASIZADEH M. Efficient scalable multi-party private set intersection using oblivious PRF[C]//International Workshop on Security and Trust Management. Berin: Springer, 2021: 81-99.
[32] GHOSH S, NILGES T. An algebraic approach to maliciously secure private set intersection[C]//Advances in Cryptology-EUROCRYPT 2019. Berin: Springer, 2019: 154-185.

[作者简介]

赖成喆（1985–），男，陕西汉中人，博士，西安邮电大学教授，主要研究方向为安全协议设计与分析、车联网安全。

杨婷（2000–），女，陕西汉中人，西安邮电大学硕士生，主要研究方向为隐私保护集合求交。

秦宝东（1982–），男，江苏徐州人，博士，西安邮电大学教授，主要研究方向为公钥密码理论及其应用。

曹进（1985–），男，陕西西安人，博士，西安电子科技大学教授，主要研究方向为5G、6G、天地一体化网络安全。