J. Internet Comput. Serv.
ISSN 1598-0170 (Print) / ISSN 2287-1136 (Online)
http://www.jics.or.kr

정확도가 향상된 안전한 Top-k 검색 기반 서비스형 블록체인과 스마트 컨트랙트 설계☆

Design Blockchain as a Service and Smart Contract with Secure Top-k Search that Improved Accuracy

장 호 빈[1] 천 지 영[2] 정 익 래[1*] 노 건 태[2*]
Hobin Jang Ji Young Chun Ik Rae Jeong Geontae Noh

요 약

클라우드 컴퓨팅 기술 발전과 함께 이커머스, 금융 기업 등 다양한 영역에서 클라우스 서비스 제공자의 서비스형 블록체인을 활용하여 고객 이력 관리, 유통 이력 관리 등을 진행하고 있다. 하지만 추천 알고리즘, 검색 엔진 개발 등의 영역에서 사용자의 검색 이력, 구매 이력 등을 서비스형 블록체인에 활용하고자 하는 경우, 사용자의 검색 쿼리는 서비스형 블록체인을 운영하는 기업에 노출되며, 이에 대한 프라이버시 문제가 야기될 수 있다. Z. Guan 등의 연구는 컨소시엄 블록체인 환경에서 검색 가능 암호를 활용하여 사용자의 검색 쿼리와 검색 결과 간의 비연결성을 보장하며, 내적 유사도를 기반으로 사용자의 검색 쿼리와 관련성이 높은 Top-k 결과를 선정한다. 하지만 내적 유사도의 동점에 의해 Top-k 결과 중 일부가 선정 불가능한 문제점이 존재하며, 클라우드 기반의 서비스형 블록체인 환경은 고려되지 않았다. 따라서 본 논문은 코사인 유사도를 활용하여 Z. Guan 등 연구의 문제점을 해결하여 검색 결과의 정확도를 향상한다. 그리고 이를 바탕으로 정확도가 향상된 안전한 Top-k 검색 기반 서비스형 블록체인 설계 및 프라이버시를 보호하며 사용자의 검색과 관련성이 높은 Top-k 검색 결과를 얻을 수 있는 스마트 컨트랙트를 설계한다.

☞ 주제어 : 검색 가능 암호, Top-k 검색, 프라이버시 보호, 서비스형 블록체인, 스마트 컨트랙트

ABSTRACT

With advance of cloud computing technology, Blockchain as a Service of Cloud Service Provider has been utilized in various areas such as e-Commerce and financial companies to manage customer history and distribution history. However, if users' search history, purchase history, etc. are to be utilized in a BaaS in areas such as recommendation algorithms and search engine development, the users' search queries will be exposed to the company operating the BaaS, and privacy issues will be occured. Z. Guan et al. ensure the unlinkability between users' search query and search result using searchable encryption, and based on the inner product similarity, they select Top-k results that are highly relevant to the users' search query. However, there is a problem that the Top-k results selection may be not possible due to ties of inner product similarity, and BaaS over cloud is not considered. Therefore, this paper solve the problem of Z. Guan et al. using cosine similarity, so we improve accuracy of search result. And based on this, we design a BaaS with secure Top-k search that improved accuracy. Furthermore, we design a smart contracts that preserve privacy of users' search and obtain Top-k search results that are highly relevant to the users' search.

☞ keyword : Searchable Encryption, Top-k Search, Privacy Preserving, Blockchain as a Service, Smart Contract

1. 서 론

클라우드 컴퓨팅은 4차 산업혁명의 핵심 기술 중 하나로서, Amazon, Microsoft, Google 등 다양한 클라우드 서비스 제공자(Cloud Service Provider, CSP)의 서비스를 기업에서 활용하고 있다. 클라우드 서비스 제공자는 기업이 필요로 하는 요구사항에 따라 IaaS, PaaS, SaaS 등 다양한 서비스를 제공하고 있다. 이와 함께 블록체인 운영에 필요한 인프라 구축, 자원 제공 등의 기능을 제공하는 서비스형 블록체인(Blockchain as a Service, BaaS)이 존재한다.

1 Department of Information Security, Korea University, Seoul, 02841, Korea
2 Department of Bigdata & Information Security, Seoul Cyber University, Seoul, 01133, Korea
* Corresponding authors (irjeong@korea.ac.kr, gnoh@iscu.ac.kr)
[Received 12 July 2023, Reviewed 19 July 2023, Accepted 3 Septmeber 2023]
☆ 이 논문은 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2021R1F1A1060543).

대표적으로 Amazon의 Amazon Managed Blockchain, IBM의 IBM Blockchain 등 다양한 서비스형 블록체인이 존재하며, 공개형 블록체인인 Ethereum과 컨소시엄 블록체인인 Hyperledger Fabric 환경을 주로 제공한다.

이를 바탕으로 다양한 분야의 기업들이 서비스형 블록체인을 이용하여 고객 이력 관리, 유통 이력 관리 등을 진행 중이다. 기업은 단일 서버를 통한 데이터 관리 시 발생하는 단일 장애 지점 문제 해결과 데이터의 변조 불가능성을 위해 블록체인을 사용하며, 이에 따른 자원 할당과 환경 구축을 위해 서비스형 블록체인을 이용하여 기업이 원하는 서비스를 운용한다. 예를 들어, 대한항공은 COVID-19 백신 수송기 정보 관리 및 백신 재고의 위변조 여부 확인에 Amazon Managed Blockchain을 이용하였으며[1], Home Depot은 IBM Blockchain을 활용하여 고객의 구매 이력, 물건의 유통 이력 관리를 진행한다[2].

하지만 서비스형 블록체인을 사용하는 기업의 관점에서 클라우드는 외부 데이터 저장소이며, 이에 따라 기업의 데이터 및 해당 서비스를 이용하는 사용자의 프라이버시 보호 문제가 발생할 수 있다. 암호화되지 않은 데이터를 클라우드에 저장 시, 데이터에 대한 기밀성, 접근 권한 등 데이터 프라이버시가 클라우드 서비스 제공자에게 노출된다. 또한, 사용자의 검색 이력 및 정보 저장을 위한 입력 내용에 위치 정보, 개인 식별 정보 등이 포함되는 경우, 서비스 사용자의 개인 정보가 클라우드 서비스 제공자에게 노출되며, 검색 내용과 사용자 간의 연결성에 따른 프라이버시 문제가 발생할 수 있다. 이를 해결하기 위해 클라우드 환경에 대한 검색 가능 암호화 기법이 연구되었으며, 대칭키 및 공개키 기반 검색 가능 암호화, 블록체인을 활용한 기법 등 다양한 연구가 진행되고 있다[3].

Z. Guan 등의 연구는 컨소시엄 블록체인을 활용한 이커머스 서비스 운용 시 사용자의 검색 내용에 대한 프라이버시를 보호하는 연구이다[4]. 컨소시엄 블록체인을 이커머스 서비스 환경의 서버로 사용하여, 사용자의 검색 쿼리에 대해 올바른 검색 결과를 제공한다. 컨소시엄 블록체인을 사용함으로써, 일반적인 서버의 단일 장애 지점 오류 해결과 검색 결과 반환 시 올바르지 않은 검색 결과를 반환하는 악의적인 노드 검출, 데이터의 변조 불가능성 등을 이룬다. 또한, 검색 가능 암호를 사용한 데이터 암호화 및 사용자의 검색 쿼리와 검색 결과인 레코드 간의 비연결성을 보장함으로써, 사용자의 검색에 대한 프라이버시를 보호한다. 그리고 내적 유사도를 기반으로 검색 쿼리와 관련성이 높은 k개의 결과인 Top-k 레코드를 사용자에게 반환함으로써, 이커머스 서비스 사용자가 원하는 상품 검색에 대한 검색 엔진 역할을 수행한다.

하지만 [4]의 연구는 사용자의 검색에 대한 프라이버시는 보장하지만, 내적 유사도를 기반으로 검색 쿼리와 레코드 간의 관련성을 계산하므로, 관련성 간의 동점으로 인한 Top-k 레코드 중 일부가 선정 불가능한 문제점이 존재하며, 클라우드 기반의 서비스형 블록체인 환경은 고려되지 않았다.

따라서 본 논문은 코사인 유사도를 바탕으로 [4]의 연구의 Top-k 레코드 선정 불가능 문제를 해결하여 정확도가 향상된 안전한 Top-k 검색을 이룬다. 또한, 이를 바탕으로 사용자의 검색에 대한 프라이버시를 보호하는 클라우드 환경에서의 컨소시엄 블록체인 기반 서비스형 블록체인 설계 및 올바른 검색 결과를 반환하는 스마트 컨트랙트를 제안한다. 본 연구를 바탕으로 사용자의 검색 기록 및 입력 정보 기반의 상품 추천 서비스, 검색 엔진 개발 등 다양한 영역에서 사용자의 검색 프라이버시를 보호하며 신뢰성 있는 서비스형 블록체인을 운용할 수 있을 것으로 기대한다.

2. 배경 지식

2.1 검색 가능 암호

데이터 소유자가 외부 데이터 저장소를 통해 데이터를 저장하는 경우, 데이터의 기밀성을 위해 암호화된 상태로 저장할 필요가 있다. 하지만 데이터에 대한 일부 검색을 위해 암호화된 데이터 전체를 복호화하는 것은 비효율적이므로, 데이터에 대한 키워드를 통해 필요한 검색 내용만을 갖는 데이터를 복호화하는 방법이 검색 가능 암호이다[3]. 검색 가능 암호는 대칭키, 공개키 암호 등 다양한 암호화 기법이 적용될 수 있으며, 키 생성, 검색 가능 암호화, 트랩도어 생성, 검색 등의 과정을 거친다.

그리고 안전성 및 효율성 확대를 위해 암호화된 데이터의 키워드 정보를 갖는 인덱스 부여, 검색 대상자에 대한 접근 통제, 다중 키워드 검색 등의 다양한 연구가 이루어졌다. 또한, 암호화된 데이터와 해당 키워드 간의 비연결성, 검색 내용과 검색 결과 간의 비연결성 보장 등 프라이버시 보호를 위한 검색 가능 암호가 연구되었다[5].

최근 클라우드 컴퓨팅 기술 발전과 함께 클라우드 환경에 검색 가능 암호를 적용하고자 하는 연구가 진행되고 있다. 다양한 검색 가능 암호와 클라우드 컴퓨팅 기술을 바탕으로 의료 산업, 사물인터넷 등 여러 산업 환경에 맞는 검색 가능 암호화 기법이 연구되고 있다[6, 7]. 그리

고 블록체인을 활용하여 검색 결과에 대한 무결성 검증 및 보상 지급 방안 등에 관한 연구가 진행 중이다[8].

2.2 컨소시엄 블록체인

블록체인은 분산 컴퓨팅 기술 중 하나로써, 탈중앙화, 데이터 불변성 등의 특징을 갖는다. 블록체인은 노드 간에 탈중앙화된 합의를 통해 거래 기록을 담은 블록을 생성하며, 모든 노드가 동일한 블록을 체인 형태로 소유한다. 이를 바탕으로 블록에 기록된 데이터는 불변성을 갖게 된다. 이후 Ethereum에서 원하는 계약을 체결할 수 있는 스마트 컨트랙트 기능을 추가하였으며, PoS, BFT 등의 다양한 합의 알고리즘 기반의 블록체인이 등장하였다[9].

컨소시엄 블록체인은 허가된 노드만이 참여할 수 있는 프라이빗 블록체인의 일종으로, Linux Foundation, IBM 등이 참여 중인 Hyperledger Foundation으로부터 시작되었다. Bitcoin, Ethereum과 같은 공개형 블록체인과 달리, 허가받은 노드만이 참여할 수 있는 기업에 특화된 블록체인이다. 현재 Hyperledger Fabric, Indy 등 다양한 프로젝트가 진행 중이며, IBM이 참여한 Hyperledger Fabric을 이용한 서비스 및 연구가 주로 이루어지고 있다[10].

Hyperledger Fabric은 Peer, Organization, Orderer로 구성된 채널에 따라 블록체인을 구성한다. 각 채널은 독립적이며, Organization에 대한 접근 통제, 합의 알고리즘 선정 등의 정책을 구성한다. Organization은 블록체인 노드인 Peer의 모음으로, Peer에 대한 인증서, 참여 중인 채널에 대한 인증서 등을 관리하는 MSP(Membership Service Provider) 역할을 한다. Orderer는 블록을 생성하는 역할로, Peer 간에 발생한 트랜잭션을 취합하여 블록을 생성한다. 블록 생성 시 BFT 합의 알고리즘 기반의 Raft 알고리즘을 사용하며, 생성된 블록을 Peer에게 전달한다. Peer는 블록체인을 운영하는 노드로 트랜잭션 발생 및 검증을 수행하며, 트랜잭션의 모음인 블록을 저장한다. Anchor Peer는 Orderer로부터 받은 블록을 동일 Organization의 다른 Peer에게 전달한다. 그리고 Hyperledger Fabric은 Chaincode라는 스마트 컨트랙트 기능을 제공한다. 공개형 블록체인의 스마트 컨트랙트가 배포된 이후 수정이 어려운 것과 달리, Chaincode는 채널에 참여한 Organization 간의 합의를 통해 수정 및 버전 라이프사이클 관리가 가능하다[11].

3. 관련 연구

N. Cao 등의 연구는 클라우드 환경에서 프라이버시를 보호하는 검색 가능 암호화 기법이다[12]. 해당 연구는 대칭키 기반의 검색 가능 암호이며, 대칭키 암호키를 바탕으로 사용자의 검색 시 별도의 암호키를 생성한다. 이를 바탕으로 사용자의 검색과 암호화된 레코드 간의 비연결성을 보장한다. 또한, 다중 키워드 검색이 가능하며, 내적 유사도를 통해 관련성이 높은 레코드를 계산한다.

L. Chen 등의 연구는 의료 데이터 공유를 위해 블록체인을 활용한 검색 가능 암호에 관한 연구이다[13]. 해당 연구에서 클라우드는 암호화된 레코드만을 가지며, 블록체인의 스마트 컨트랙트를 통해 레코드의 인덱스를 관리한다. 이후 사용자는 스마트 컨트랙트를 통해 얻은 인덱스를 바탕으로 쿼리를 요청하고, 복호화를 진행한다. 또한, 기존 레코드 소유자가 의료 기록을 제공함에 대해 블록체인 기반의 보상 과정을 설계하였다.

Z. Guan 등의 연구는 컨소시엄 블록체인 기반의 이커머스 서비스 운용에 대한 검색 가능 암호화 기법 연구이다[4]. 해당 연구는 [12]의 검색 가능 암호를 바탕으로 사용자의 검색 쿼리와 컨소시엄 블록체인에 기록된 레코드 간의 비연결성을 보장한다. 또한, 내적 유사도를 바탕으로 관련성이 높은 Top-k 레코드를 얻도록 설계하였다.

이처럼 다양한 환경의 검색 가능 암호에 관한 연구가 진행되고 있다. 하지만 [13]의 연구와 같이 블록체인을 활용해 클라우드에 대한 검색 가능 암호를 진행하는 경우, 클라우드와 블록체인 환경 구성을 별도로 진행해야 하는 문제점이 존재한다. [4]의 연구는 내적 유사도를 기반으로 검색 쿼리와 관련성이 높은 Top-k 레코드를 반환하므로 사용자는 모호한 결과를 얻을 수 있다. 따라서 본 논문은 [4]의 문제점 해결을 통해 Top-k 검색의 정확도 향상 및 이를 바탕으로 안전한 검색이 가능한 클라우드 환경의 서비스형 블록체인 및 스마트 컨트랙트를 설계한다.

4. 컨소시엄 블록체인 환경의 검색 가능 암호

4.1 기존 연구의 시나리오

[4]의 연구는 이커머스 서비스 운용 시 컨소시엄 블록체인을 서버로 사용할 때, 사용자의 검색 프라이버시를 보장하는 연구이다. 이커머스 서비스에 필요한 데이터 저장 및 사용자의 검색 쿼리에 대한 응답을 위해 컨소시엄 블록체인을 사용하며, 단일 장애 지점 오류 해결, 데이터 변조 불가능성 등의 장점을 갖는다. 또한, 블록체인을 구성하는 노드 간의 합의를 통해 올바르지 않은 검색 결과를 제공하는 노드를 검출한다.

그리고 검색 가능 암호를 사용하여, 이커머스 서비스가 보유 중인 데이터의 암호화 및 사용자의 검색 쿼리와 검색 결과 간의 비연결성을 보장한다. 컨소시엄 블록체인을 통해 검색 가능 암호화를 하고자 하며, 사용자에게 반환되는 데이터를 레코드, 검색 가능 암호화된 레코드를 인덱스라 한다. 최종적으로 내적 유사도를 기반으로 사용자의 검색 쿼리와 관련성이 높은 k개의 결과인 Top-k 레코드를 사용자에게 반환하는 검색 엔진 역할을 수행한다.

[4]의 연구 환경을 구성하는 요소는 표 1과 같다. 전체적인 환경은 인증 기관, 데이터 제공자, 컨소시엄 블록체인, 데이터 요청자로 구성되며, 전체적인 시나리오는 다음과 같다.

〈표 1〉 [4]의 구성 요소

(Table 1) Components of [4]

구성 요소	역할
인증 기관	- 구성 요소에 대한 인증서 발급 - 검색 가능 암호에 사용되는 암호키 발급
데이터 제공자	- 컨소시엄 블록체인과 상호 인증 - 컨소시엄 블록체인에 데이터 제공
컨소시엄 블록체인	- 암호키를 통한 레코드의 검색 가능 암호화 - 데이터 요청자의 검색에 관한 결과 반환
데이터 요청자	- 검색 쿼리를 검색 토큰으로 변환 - 컨소시엄 블록체인에 검색 결과 요청

- 인증 기관 : 데이터 제공자, 컨소시엄 블록체인, 데이터 요청자에 대한 인증서를 발급한다. 컨소시엄 블록체인과 데이터 요청자에게 검색 가능 암호에 사용될 대칭키 암호키를 발급한다.
- 데이터 제공자 : 이커머스 서비스 운용 시 필요한 데이터를 제공하는 역할이다. 인증 기관으로부터 발급받은 인증서를 바탕으로 컨소시엄 블록체인과 올바른 구성 요소임을 서로 확인 후, 자신이 소유한 데이터를 컨소시엄 블록체인에 제공한다.
- 컨소시엄 블록체인 : 이커머스 서비스를 운용하는 서버 역할이다. 상호 인증을 완료한 데이터 제공자로부터 전달받은 데이터에 대해 어떤 키워드를 가질 수 있는 지 분류한다. 그리고 인증 기관으로부터 발급받은 암호키를 이용하여 검색 가능 암호화한다. 검색 가능 암호화 대상인 데이터를 레코드, 레코드를 검색 가능 암호화한 것을 인덱스라 한다. 컨소시엄 블록체인을 구성하는 각 노드는 모두 동일한 레코드, 인덱스 쌍을 가지며, 데이터 요청자의 검색 쿼리에 대한 결과를 반환한다.
- 데이터 요청자 : 이커머스 서비스를 이용하는 사용자 역할이다. 인증 기관으로부터 발급받은 인증서를 바탕으로 컨소시엄 블록체인과 올바른 구성 요소임을 서로 확인한다. 그리고 자신이 원하는 키워드가 어떤 것인지를 검색 쿼리로 사용하며, 인증 기관으로부터 발급받은 암호키를 사용하여 검색 쿼리를 검색 토큰으로 변환한다. 검색 토큰과 함께 검색 쿼리와 관련성이 높은 k개의 결과를 컨소시엄 블록체인에 요청한다. 검색 토큰 생성 시 랜덤값을 이용함으로써, 검색 토큰과 검색 결과 간의 비연결성이 보장된다. 이를 통해 데이터 요청자는 자신의 검색에 대한 프라이버시를 보호한다.

4.2 기존 연구의 기법

컨소시엄 블록체인이 인덱스를 생성하는 과정은 다음과 같다. 데이터 제공자 i가 컨소시엄 블록체인 노드 j에게 제공한 데이터를 $record_{i,j}$라 할 때, j는 인증 기관으로부터 발급받은 암호키 $SK=\{S, M_1, M_2\}$를 이용하여 $record_{i,j}$에 대해 검색 가능 암호화한 인덱스 $I_{i,j}$를 생성한다. SK의 S는 $(n+1)$차원 불리언 벡터, M_1, M_2는 $(n+1)\times(n+1)$크기의 가역 행렬이며, n은 가능한 키워드의 개수이다. $record_{i,j}$에 존재하는 키워드를 n차원 불리언 벡터로 표현한 $V_{i,j}$을 $(n+1)$차원 벡터 $V^*_{i,j}=(1, V_{i,j})$로 변환한다. 그리고 $t\in[1,n+1]$에 대해 식 (1), (2)와 같이 $(n+1)$차원 불리언 벡터 $V'_{i,j}$, $V''_{i,j}$를 계산 후, 식 (3)과 같이 $I_{i,j}$를 계산한다.

$$\text{if } S[t]=0,\ V'_{i,j}[t]=V''_{i,j}[t]=V^*_{i,j}[t] \quad (1)$$

$$\text{if } S[t]=1,\ V'_{i,j}[t]+V''_{i,j}[t]=V^*_{i,j}[t] \quad (2)$$

$$I_{i,j}=\left\{V'_{i,j}M_1^T,\ V''_{i,j}M_2^T\right\} \quad (3)$$

데이터 요청자의 검색 토큰 생성은 다음과 같다. 데이터 요청자는 검색 쿼리에 존재하는 키워드를 n차원 불리언 벡터로 표현한 Q, 랜덤값 r, ϵ을 이용해 $(n+1)$차원 $Q^*=(r, \epsilon Q)$를 생성한다. 그리고 SK, $t\in[1,n+1]$에 대해 식 (4), (5)와 같이 $(n+1)$차원 불리언 벡터 Q', Q''를 계산 후, 식 (6)과 같이 검색 토큰 ST를 계산한다.

$$\text{if } S[t]=0,\ Q'[t]+Q''[t]=Q^*[t] \quad (4)$$

$$\text{if } S[t]=1,\ Q'[t]=Q''[t]=Q^*[t] \quad (5)$$

$$ST = \{Q^{'}M_1^{-1}, Q^{''}M_2^{-1}\} \quad (6)$$

데이터 요청자는 레코드 검색을 위해 ST를 컨소시엄 블록체인 노드에 전달한다. 컨소시엄 블록체인은 식 (7)과 같은 내적 연산($\bullet$)을 통해 ST와 관련성이 높은 Top-k 개의 $I_{i,j}$를 탐색 후, 해당하는 레코드를 데이터 요청자에게 반환한다. 컨소시엄 블록체인의 $I_{i,j} \bullet ST$ 연산 결과는 실제 레코드의 키워드 여부인 $V_{i,j}$와 검색 쿼리의 키워드 여부인 Q의 내적 $V_{i,j} \bullet Q$에 선형성을 가지므로 $V_{i,j}, Q$의 공통 키워드 개수에 따라 Top-k 레코드가 결정된다. 그리고 r, ϵ이 각각 η_r, η_ϵ 비트, S에 존재하는 0의 개수를 α라 할 때, 데이터 요청자가 동일한 ST를 생성할 확률은 $2^{-(\eta_r+\eta_\epsilon+\alpha(n+1))}$이다. 따라서 충분히 큰 η_r, η_ϵ에 의해 ST와 $I_{i,j}$의 비연결성이 보장된다.

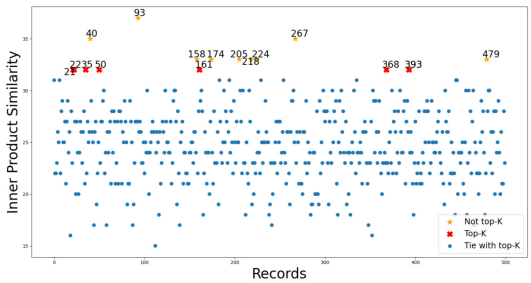
$$\begin{aligned} & I_{i,j} \bullet ST \quad (7) \\ &= \{V_{i,j}^{'}M_1^T, V_{i,j}^{''}M_2^T\} \bullet \{Q^{'}M_1^{-1}, Q^{''}M_2^{-1}\} \\ &= V_{i,j}^{'} \bullet Q^{'} + V_{i,j}^{''} \bullet Q^{''} = (1, V_{i,j}) \bullet (r, \epsilon Q) \\ &= r + \epsilon V_{i,j} \bullet Q \end{aligned}$$

4.3 기존 연구의 문제점

[4]의 연구는 데이터 요청자의 검색 토큰과 레코드의 인덱스 간의 비연결성을 보장하지만, 내적 연산을 기반으로 검색 쿼리와 레코드 간의 공통 키워드 수를 관련성으로 계산한다. 따라서 내적 연산의 결과가 동점이 되는 레코드가 존재할 수 있으며, 그림 1와 같이 Top-k 레코드 선정이 불가능한 경우가 발생한다.

그림 1은 [4]의 기법으로 100개의 키워드를 가질 수 있는 랜덤한 500개의 레코드를 갖는 데이터셋에 대해 5개의 키워드를 갖는 랜덤한 쿼리와 관련성이 높은 10개의 Top-k 레코드를 선정한 결과이다. 그리고 500개의 레코드는 키워드 개수에 대해 이항분포 $B(100, 0.5)$를 따르도록 구성하였다. 즉, 500개의 레코드 중 키워드 m개를 갖는 레코드는 $500 \times {}_{100}C_m \, 2^{-100}$ 개가 존재한다. m개 키워드의 위치는 랜덤하게 선택하였다. 그리고 쿼리의 검색 토큰 생성에 사용되는 r, ϵ은 안전성에 대한 파라미터이므로 $r = 0, \epsilon = 1$으로 설정하였다.

9개의 주황색 점은 Top-k 레코드를 의미하며, 40, 93, 158, 174, 205, 218, 224, 267, 479번 레코드가 Top-k 레코드로 선정되었다. 하지만 7개의 빨간색 점인 21, 22, 35, 50, 161, 368, 393번 레코드에 대해서는 내적 연산 동점으로 인해 나머지 하나의 Top-k 레코드 선정이 불가능하다. 따라서, [4]의 연구를 바탕으로 사용자의 검색 쿼리인 검색 토큰과 검색 결과인 인덱스 간의 관련성이 높은 Top-k 레코드 선정 시, 일부에 대한 선정이 불가능하다.



(그림 1) Top-k 레코드 선정 불가능 예시
(Figure 1) Example of Top-k records selection not possible

4.4 해결 방안 및 성능 비교

위 문제를 해결하기 위해 본 논문에서는 코사인 유사도를 기반으로 데이터 요청자의 검색 토큰 ST와 관련성이 높은 Top-k 레코드를 선정한다. 코사인 유사도는 두 벡터 간의 내적 연산 시 사용되는 코사인 값을 이용하여 유사도를 측정하는 방법이다. 유사도가 1에 가까운 값을 가질수록 두 벡터가 유사하며, -1에 가까운 값을 가질수록 두 벡터가 유사하지 않음을 의미한다. 차원이 같은 두 벡터 a, b에 대해 $\| a \|$, $\| b \|$는 a, b의 크기, θ는 a, b가 이루는 각일 때, $a \bullet b = \| a \| \| b \| \cos(\theta)$ 이다. 따라서 a, b의 코사인 유사도는 식 (8)과 같다.

$$sim(a, b) = \cos(\theta) = \frac{a \bullet b}{\| a \| \| b \|} \quad (8)$$

코사인 유사도를 사용하여 데이터 요청자의 검색 토큰 ST와 관련성이 높은 Top-k 레코드를 탐색하는 알고리즘은 그림 2와 같다. 그림 2의 알고리즘은 [4]의 기법에 사용된 r, ϵ을 이용하여 검색 토큰 ST을 구성하므로 사용자의 검색 쿼리와 암호화된 레코드의 비연결성을 동일하게 보장한다. 그림 2의 알고리즘을 이용하여 4.2절의 실험과 동일한 레코드와 쿼리를 사용하여 10개의 Top-k 레코드를 선정한 결과는 그림 3와 같다.

총 10개의 주황색 점인 22, 40, 50, 93, 134, 158, 205,

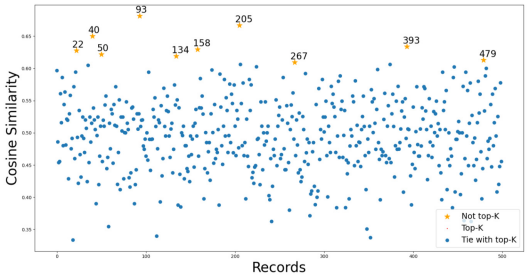
267, 393, 479번 레코드가 Top-k 레코드로 선정되었으며, Top-k 레코드 선정이 불가능한 빨간색 점은 나타나지 않는다. 또한, 그림 1의 Top-k 레코드 선정이 불가능한 7개 중 22, 50, 393번 레코드가 Top-k 레코드로 선정되었으며, 134번 레코드가 새롭게 Top-k 레코드로 선정되었다. 그리고 그림 1의 Top-k 레코드 중 174, 218, 224번 레코드는 그림 2의 알고리즘 이용 시, Top-k 레코드로 선정되지 않았다. 따라서 [4]의 연구의 문제점인 내적 유사도의 동점에 따른 Top-k 레코드 선정 불가능 문제를 코사인 유사도 기반의 그림 2 알고리즘으로 해결 가능함을 확인하였다.

Algorithm 1 :
Top-k Records Based on Cosine Similarity

Input : $\{I_{i,j} \mid i \in [1, DS], j \in [1, BN]\}, ST, k$
where DS is the number of Data Sources, BN is the number of Consortium Blockchain Nodes, and k is the number of records to find.
Output : $\{record_{i,j} \mid sim(I_{i,j}, ST)\ is\ in\ Top-k\}$
where sim is cosine similarity

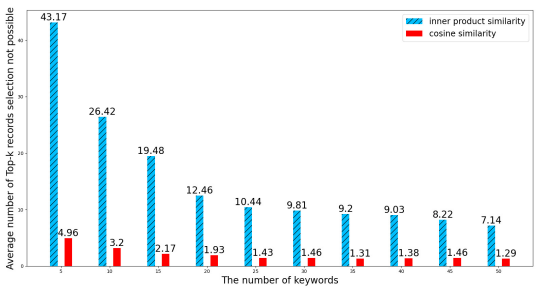
1. Create $I = \{\ \}$, $R = \{\ \}$
2. For $i = 1$ to DS :
3. For $j = 1$ to BN :
4. Add $sim(I_{i,j}, ST)$ to I
5. End For
6. End For
7. Sort I in descending order
8. For $t = 1$ to k :
9. Add $record_{i,j}$ corresponding to $I[t]$ to R
10. End For
11. Return R

(그림 2) 코사인 유사도 기반 Top-k 레코드 탐색
(Figure 2) Top-k records based on cosine similarity



(그림 3) 랜덤 데이터셋에 대한 코사인 유사도 기반 Top-k 레코드 판별
(Figure 3) Top-k records based on cosine similarity on random dataset

추가적인 비교를 위해 가능한 키워드 수가 100개인 랜덤한 500개의 레코드에 대해 이항분포 $B(100, p)$를 따르는 경우를 살펴본다. 즉, 500개의 레코드 중 키워드를 m개 갖는 레코드는 $500 \times {}_{100}C_m\, p^m (1-p)^{100-m}$ 개가 존재한다. 쿼리는 총 100개이며, 해당 분포의 평균인 $100p$개의 랜덤한 키워드를 갖는다. 그리고 그림 1의 빨간색 점과 같이 쿼리와 관련성이 높은 레코드 중 Top-k 선정이 불가능한 레코드의 평균 개수를 비교한다. p는 0.05부터 0.5까지 0.05의 간격이다. 즉, 레코드가 갖는 평균 키워드 개수는 5개부터 50개까지이다. 결과는 그림 4와 같으며, 파란색 막대는 [4]의 기법을 사용한 경우, 빨간색 막대는 그림 2의 알고리즘을 사용한 경우이다. [4]의 기법을 사용 시, 최대 43.17개의 Top-k 선정이 불가능한 레코드가 존재하며, 그림 2의 알고리즘 사용 시, 최대 4.96개의 Top-k 선정이 불가능한 레코드가 존재한다. 두 기법의 선정 불가능한 Top-k 레코드 개수는 평균 키워드가 50개일 때 최소 5.53배, 15개일 때 최대 8.98배의 차이가 남을 알 수 있다.



(그림 4) 랜덤 데이터셋에 대한 [4]의 기법과 그림 2 알고리즘의 비교
(Figure 4) Comparison algorithm of [4] with figure 2 on random dataset

이어서 실제 데이터셋을 바탕으로 [4]의 기법과 그림 2의 알고리즘을 비교한다. 데이터셋은 Market Basket 데이터셋과 MovieLens 데이터셋을 사용하였다[14, 15]. [14]의 데이터셋은 Apple, Bread, Butter 등의 키워드로 구성되며, 레코드가 해당 키워드를 갖는지에 대한 불리언 벡터로 구성되어 있다. 총 키워드는 16개이며, 전체 레코드 수는 999개이다. 실험에서는 999개의 레코드 중 100개의 레코드를 랜덤하게 선택하여 쿼리로 사용하며, 나머지 899개의 레코드 중 관련성이 높은 Top-k 레코드를 선정한다. 그리고 두 기법을 이용하여 각 쿼리와 관련성이 높은 레코드 중 Top-k 선정이 불가능한 레코드의 평균 개수를 계

산한다. [15]의 데이터셋은 레코드가 Adventure, Drama, Comedy 등의 장르를 갖는 영화인지를 분류한다. 영화 장르의 유사도 측정을 위해 장르를 키워드로 갖도록 레코드를 전처리 후 실험을 진행하였다. 총 키워드는 19개이며, 전체 레코드 수는 27278개이다. 27278개의 레코드 중 100개의 레코드를 랜덤하게 선택하여 쿼리로 사용하며, 나머지 27178개의 레코드 중 관련성이 높은 Top-k 레코드를 선정한다. 그리고 각 쿼리와 관련성이 높은 레코드 중 Top-k 선정이 불가능한 레코드의 평균 개수를 계산한다.

실험은 Top-k 레코드 선정에 사용되는 k를 10, 20, 30, 40, 50으로 분류하여 진행하였으며, 실험 결과는 표 2와 같다. 사용되는 쿼리는 k에 대해 동일한 쿼리를 사용하였다. [14]의 데이터셋의 경우, [4]의 기법 사용 시, 최대 115.61개의 Top-k 선정이 불가능한 레코드가 존재하며, 그림 2의 알고리즘 사용 시, 최대 16.69개의 Top-k 선정이 불가능한 레코드를 얻는다. [15]의 데이터셋의 경우, [4]의 기법 사용 시, 최대 4498.02개의 Top-k 선정이 불가능한 레코드를 얻으며, 그림 2의 알고리즘 사용 시, 최대 1757.64개의 Top-k 선정이 불가능한 레코드를 얻는다. 그리고 두 기법은 [14]에 대해 k=10일 때 최대 10.96배, k=40일 때 최소 6.64배 차이가 발생한다. 그리고 [15]에 대해 k=10일 때 최대 2.60배, k=50일 때 최소 2.56배 차이가 발생함을 알 수 있다. 따라서 검색 쿼리인 검색 토큰과 검색 결과인 인덱스의 관련성 계산 시, 코사인 유사도 기반의 Top-k 레코드 선정으로 정확도가 향상됨을 확인하였다.

(표 2) 실제 데이터셋에 대한 [4]의 기법과 그림 2 알고리즘의 비교
(Table 2) Comparison algorithm of [4] with figure 2 on real dataset

Dataset \ Top-k		10	20	30	40	50
[14]	[4]	68.81	80.46	88.23	99.43	115.61
	our	6.28	8.34	10.94	14.97	16.69
[15]	[4]	4455.87	4457.63	4484.36	4488.04	4498.02
	our	1715.91	1728.54	1732.33	1737.62	1757.64

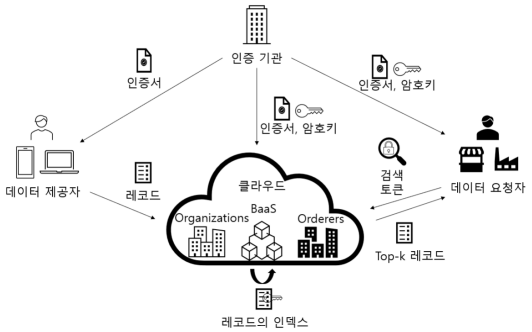
5. 안전한 검색이 가능한 서비스형 블록체인

5.1 제안 시스템 환경 구성

4장을 통해 [4]의 연구의 문제점인 사용자의 검색 결과에 대한 Top-k 레코드 선정 시, 내적 유사도에 따른 동점 문제를 해결하였다. 이를 바탕으로 정확도가 향상된 안전한 Top-k 검색을 이용하여 사용자의 검색에 대한 프라이버시를 보호하는 서비스형 블록체인 설계 및 안전한 검색을 위한 스마트 컨트랙트를 제안한다. 전체적인 구조는 그림 5와 같으며, 환경 구성 요소는 표 3과 같다.

(표 3) 서비스형 블록체인 구성 요소
(Table 3) Components for BaaS

구성 요소	역할
CA	- 인증 기관 - 구성 요소에 대한 인증서 및 검색 가능 암호화의 대칭키 암호키 관리
DS_x	- x번째 데이터 제공자 - 컨소시엄 블록체인에 레코드 제공
DR_y	- y번째 데이터 요청자 - 검색 토큰 송신 및 Top-k 레코드 수신
CB	- 컨소시엄 블록체인 - 레코드에 대한 검색 가능 암호화 - 데이터 요청자에게 Top-k 레코드 반환
Ch	- Organization, Orderer가 참여하는 채널
Org_i	- i번째 Organization - 0번째 Peer는 Anchor Peer로 운영
$Peer_{i,j}$	- i번째 Organization의 j번째 Peer - $j=0$인 경우, Anchor Peer를 의미
$Orderers$	- Ch에 참여 중인 Orderer의 집합
$cert_\alpha$	- CA가 발급한 구성 요소 α의 인증서
$cred_{Org_i}$	- Org_i의 클라우드 크리덴셜 정보
$record_{x,i}$	- DS_x가 $Peer_{i,j}$에게 제공한 레코드
$I_{x,i}$	- $record_{x,i}$의 검색 가능 암호화된 인덱스
SK	- 검색 가능 암호화에 사용되는 암호키
ST_y	- DR_y의 검색 토큰
KW	- 가능한 키워드의 전체 집합



(그림 5) 안전한 검색을 위한 서비스형 블록체인 환경
(Figure 5) BaaS environment for secure search

전체적인 환경은 인증 기관, 데이터 제공자, 서비스형 블록체인, 데이터 요청자로 구성된다. 인증 기관은 각 구성 요소가 올바른 구성 요소임을 증명하는 인증서를 관리하며, 검색 가능 암호에 사용될 대칭키 암호키를 관리한다. 데이터 제공자는 서비스형 블록체인에 필요한 데이터인 레코드를 제공하며, 컨소시엄 블록체인으로 구성된 서비스형 블록체인은 데이터 제공자로부터 받은 레코드를 검색 가능 암호화한 인덱스를 생성한다. 데이터 요청자는 검색에 대한 프라이버시를 보호하기 위해 검색 퀴리를 검색 토큰으로 변환 후, 원하는 Top-k 레코드 개수인 k와 검색 토큰을 서비스형 블록체인에 전달한다. 서비스형 블록체인은 검색 토큰과 관련성이 높은 Top-k 레코드를 선정 후, 데이터 요청자에게 결과를 반환한다.

5.2 제안 시스템 설계

5.2.1 인증 기관을 통한 인증서 발급

안전한 검색이 가능한 서비스형 블록체인 설계를 위해 그림 6과 같이 인증 기관으로부터 각 구성 요소는 인증서 및 검색 가능 암호화를 위한 암호키를 발급받는다. 인증 기관은 데이터 제공자, 데이터 요청자에 대한 본인 인증 후, 그에 대한 인증서를 발급한다. 또한, 인증 기관은 서비스형 블록체인의 클라우드 크리덴셜 정보를 확인 후, 서비스형 블록체인의 각 Organization에 대한 인증서를 발급한다. 그리고 서비스형 블록체인의 각 Organization과 데이터 요청자에게 검색 가능 암호화를 위한 암호키를 발급한다.

Procedure 1 : Certificate to Components
1. CA checks personal authentications of all DS_x and DR_y, and checks $cred_{Org_i}$ of Org_i.
2. CA sends $cert_{DS_x}$ to each DS_x.
3. CA sends $cert_{DR_y}$, SK to each DR_y.
4. CA sends $cert_{Org_i}$, SK to each Org_i.

(그림 6) 인증서 발급 과정

(Figure 6) Procedure that certificate to components

5.2.2 레코드 검증 및 저장

서비스형 블록체인이 레코드 검증 및 저장하는 과정은 그림 7과 같다. 데이터 제공자 DS_x는 레코드를 제공하고자 하는 서비스형 블록체인의 노드인 $Peer_{i,j}$와 인증 기관으로부터 발급받은 인증서를 이용하여 상호 인증을 진행한다. 이후, DS_x는 $Peer_{i,j}$에게 서비스 운영에 필요한 데이터인 $record_{x,i}$를 제공한다. $Peer_{i,j}$는 인증 기관으로부터 발급받은 검색 가능 암호키를 이용하여 $record_{x,i}$를 검색 가능 암호화한 인덱스 $I_{x,i}$를 그림 8과 같은 스마트 컨트랙트를 통해 계산하며, 이에 대한 트랜잭션이 발생한다. 모든 서비스형 블록체인의 Peer는 해당 트랜잭션이 올바른 암호키로 $I_{x,i}$를 생성한 것인지를 검증한다. 검증이 완료된 트랜잭션은 서비스형 블록체인의 Orderer에게 전달되며, 해당 트랜잭션이 포함된 블록을 생성한다. 생성된 블록은 각 Organization의 Anchor Peer에게 전달되며, Anchor Peer는 동일 Organization의 모든 Peer에게 해당 블록을 전달한다. 이를 바탕으로 모든 Peer는 동일한 레코드와 인덱스 쌍을 갖는다.

Procedure 2 : Confirm and Store Record
1. DS_x sends $cert_{DS_x}$ to $Peer_{i,j}$.
2. $Peer_{i,j}$ sends $cert_{Org_i}$ to DS_x.
3. DS_x checks $cert_{Org_i}$ and $Peer_{i,j}$ checks $cert_{DS_x}$.
4. DS_x sends $record_{x,i}$ to $Peer_{i,j}$.
5. $Peer_{i,j}$ makes transaction that contain $record_{x,i}$ and $I_{x,i}$ using Figure 8.
6. All $Peer_{i,j}$ confirm the transaction whether SK is applied.
7. $Orderers$ make block that contain the transaction, and send block to all $Peer_{i,0}$
8. All $Peer_{i,0}$ send the block to all $Peer_{i,j}$ in Org_i.
9. All $Peer_{i,j}$ store the block.

(그림 7) 레코드 검증 및 저장 과정

(Figure 7) Procedure that confirm and store record

데이터 제공자로부터 받은 레코드를 검색 가능 암호화하는 과정은 그림 8과 같다. 검색 가능한 키워드 집합 KW에 대해 레코드 $record_{x,i}$가 어떤 키워드를 갖는가에 대한 여부를 불리언 벡터로 표현한 $V_{x,i}$를 계산한다. 그리고 $V^*_{x,i}=(1, V_{x,i})$에 대해 인증 기관으로부터 발급받은 암호키 $SK=\{S, M_1, M_2\}$의 S를 이용하여 $S[t]=0$인 경우, $V'_{x,i}[t]=V''_{x,i}[t]=V^*_{x,i}[t]$를 계산한다. 그리고 $S[t]=1$인 경우, $V'_{x,i}[t]+V''_{x,i}[t]=V^*_{x,i}[t]$를 만족하는 $V'_{x,i}[t], V''_{x,i}[t]$를 랜덤하게 선택한다. 최종적으로 $I_{x,i}=\{V'_{x,i}M_1^T, V''_{x,i}M_2^T\}$를 이용해 $record_{x,i}$에 대한 인덱스 $I_{x,i}$를 계산한다.

Contract 1 : Generate Index of Record
Input : $SK=\{S, M_1, M_2\}$, $record_{x,i}$ where $n=\parallel KW \parallel$, S is $(n+1)$ dimension boolean vector, M_1, M_2 are $(n+1)\times(n+1)$ invertible matrix. Output : $I_{x,i}=\{V'_{x,i}M_1^T, V''_{x,i}M_2^T\}$
1. If $record_{x,i}$ is in CB : Return $I_{x,i}$ in CB 2. $V_{x,i}=(0,...,0)$ where $\parallel V_{x,i} \parallel = n$ 3. $V'_{x,i}=(0,...,0)$, $V''_{x,i}=(0,...,0)$ where $\parallel V'^{x,i} \parallel = \parallel V''_{x,i} \parallel = n+1$ 4. For $w=1$ to n : 5. If $KW[w]$ is in $record_{x,i}$: $V_{x,i}[w]=1$ 6. End For 7. $V^*_{x,i}=(1, V_{x,i})$ 8. For $t=1$ to $n+1$: 9. If $S[t]=0$: $V'_{x,i}[t]=V''_{x,i}[t]=V^*_{x,i}[t]$ 10. If $S[t]=1$: Select $V'_{x,i}[t], V''_{x,i}[t]\in\{0,1\}$ randomly such that $V'_{x,i}[t]+V''_{x,i}[t]=V^*_{x,i}[t]$ 11. End For 12. Return $I_{x,i}=\{V'_{x,i}M_1^T, V''_{x,i}M_2^T\}$

(그림 8) 레코드의 인덱스 생성

(Figure 8) Generate index of record

5.2.3 Top-k 레코드 전송

데이터 요청자에 대한 검색 결과인 Top-k 레코드를 전송하는 과정은 그림 9와 같다. 데이터 요청자 DR_y와 서비스형 블록체인 노드 $Peer_{i,j}$는 인증 기관으로부터 발급받은 인증서로 상호 인증을 진행한다. DR_y는 그림 10의 과정을 통해 자신의 검색 쿼리를 인증 기관으로부터 발급받은 암호키 $SK=\{S, M_1, M_2\}$ 를 이용하여 검색 토큰 ST_y을 생성한다. 그리고 ST_y와 원하는 검색 결과 개수인 k를 $Peer_{i,j}$에게 전송한다. $Peer_{i,j}$는 그림 11의 스마트 컨트랙트를 이용하여 검색 결과인 Top-k 레코드를 DR_y에게 반환한다.

Procedure 3 : Send Top-k Records
1. DR_y sends $cert_{DR_y}$ to $Peer_{i,j}$ 2. $Peer_{i,j}$ sends $cert_{Org_i}$ to DR_y. 3. DR_y checks $cert_{Org_i}$ and $Peer_{i,j}$ checks $cert_{DR_y}$. 4. DR_y makes ST_y using Figure 10. 5. DR_y sends ST_y, k to $Peer_{i,j}$ 6. $Peer_{i,j}$ sends Top-k records using Figure 11

(그림 9) Top-k 레코드 전송 과정

(Figure 9) Procedure that send Top-k records

Algorithm 2 : Generate Search Token
Input : $SK=\{S, M_1, M_2\}, Query$ where $Query$ is the set of keywords what DR_y want Output : $ST_y=\{Q'_yM_1^{-1}, Q''_yM_2^{-1}\}$
1. $Q_y=(0,...,0)$ where $\parallel Q_y \parallel = n$ 2. $Q'_y=(0,...,0)$, $Q''_y=(0,...,0)$ where $\parallel Q'_y \parallel = \parallel Q''_y \parallel = n+1$ 3. For $w=1$ to n : 4. If $KW[w]$ is in $Query$: $Q_y[w]=1$ 5. End For 6. Select $r, \epsilon\in\{0,1\}^\lambda$, $\epsilon\neq 0$ randomly where λ is security parameter and calculate $Q^*_y=(r, \epsilon Q_y)$ 8. For $t=1$ to $n+1$: 9. If $S[t]=0$: Select $Q'_y[t], Q''_y[t]\in\{0,1\}$ randomly such that $Q'_y[t]+Q''_y[t]=Q^*_y[t]$ 10. If $S[t]=1$: $Q'_y[t]=Q''_y[t]=Q^*_y[t]$ 11. End For 12. Return $ST_y=\{Q'_yM_1^{-1}, Q''_yM_2^{-1}\}$

(그림 10) 검색 토큰 생성

(Figure 10) Generate search token

데이터 요청자 DR_y의 검색 토큰 생성 과정은 그림 10과 같다. DR_y는 검색 가능한 키워드 집합 KW에 대해 검색 쿼리가 포함하는 키워드가 존재하는 지 여부를 불리언 벡터로 표현한 Q_y를 계산한다. 그리고 안전성을 만족하는 충분히 큰 r, ϵ에 대해 $\epsilon\neq 0$를 만족하도록 랜덤한 r, ϵ를 선택한다. 그리고 $Q^*_y=(r, \epsilon Q_y)$에 대해 인증 기관으로부터 발급받은 암호키 $SK=\{S, M_1, M_2\}$ 의 S를 이용하여 $S[t]=0$인 경우, $Q'_y[t]+Q''_y[t]=Q^*_y[t]$를 만족하도록 $Q'_y[t], Q''_y[t]$을 랜덤하게 선택한다. 그리고 $S[t]=1$인 경우, $Q'_y[t]=Q''_y[t]=Q^*_y[t]$를 계산한다. 최종적으로 계산되는 DR_y의 검색 토큰 ST_y는 $ST_y=\{Q'_yM_1^{-1}, Q''_yM_2^{-1}\}$ 이다.

데이터 요청자에 대한 검색 결과인 Top-k 레코드를 반환하는 과정은 그림 11과 같다. 데이터 요청자 DR_y는 자신의 검색 쿼리에 대한 검색 토큰 ST_y와 원하는 검색 결과 개수인 k를 서비스형 블록체인 노드 $Peer_{i,j}$에게 전송한다. $Peer_{i,j}$는 자신이 저장하고 있는 모든 블록에 기록된 레코드의 인덱스 $I_{i,j}$와 ST_y에 대해 코사인 유사도를 계산 후, 해당 결과를 내림차순 정렬한다. 정렬된 결과인 I에 대해 상위 k개의 결과를 선정한다. 하지만 4.4절의 실험 결과와 같이 코사인 유사도를 사용하여도 유사도 간의 동점이 존재할 가능성이 존재한다. 따라서, 코사

Contract 2 : Return Top-k Records

Input : $ST_y = \{Q_y^{'}M_1^{-1}, Q_y^{''}M_2^{-1}\}, k$
Output : $\{record_{i,j} \mid sim(I_{i,j}, ST)\ is\ in\ Top-k\}$
where sim is cosine similarity

1. Create $I = \{\ \}$, $S = \{\ \}$, $S' = \{\ \}$, $R = \{\ \}$
2. For $I_{i,j}$ in total blocks in CB :
3. Add $sim(I_{i,j}, ST_y)$ to I
4. End For
5. Sort I in descending order
6. Search idx that $I[idx] > I[k]$ and closest to k
7. For $t = 1$ to idx :
8. Add t to S
9. End For
10. For $t = idx + 1$ to $\| I \|$:
11. If $I[t] < I[k]$: Break loop
12. Add t to S'
13. End For
14. Select distinct t in S' randomly for $(k - idx)$ times, and add selected t to S
15. For t in S :
16. Add $record_{i,j}$ corresponding to $I[t]$ to R
17. End For
18. Return R

(그림 11) Top-k 레코드 반환
(Figure 11) Return Top-k records

인 유사도가 동점인 경우, 해당하는 $I_{i,j}$ 중 랜덤하게 선택한다. 최종적으로 선택된 상위 k개의 $I_{i,j}$에 상응하는 $record_{i,j}$를 검색 결과인 Top-k 레코드로 반환한다. 검색 결과는 코사인 유사도를 기반으로 Top-k 레코드를 선정하므로, [4]의 연구보다 더욱 정확한 결과를 도출한다.

5.3 제안 시스템 안전성 분석

제안 시스템은 인증 기관으로부터 발급받은 인증서를 기반으로 데이터 제공자, 서비스형 블록체인, 데이터 요청자가 상호 인증을 진행한다. 따라서 악의적인 데이터 제공 및 요청 시, 해당 인증서를 바탕으로 악의적인 데이터 제공자, 요청자를 추적할 수 있다.

또한, 제안 시스템은 클라우드 환경의 컨소시엄 블록체인 기반 서비스형 블록체인을 운영한다. 그러므로 다수 기업이 블록체인 노드로 참여 시, 합의를 통해 올바르지 않은 인덱스 생성 및 Top-k 레코드 반환을 수행하는 악의적인 노드 검출이 가능하다. 그리고 블록체인을 통해 레코드와 인덱스 정보에 대한 변조 불가능성을 보장한다. 또한 클라우드를 통한 동적인 블록체인 저장 공간 획득 및 자동화된 블록체인 서버 관리 등이 가능하다.

마지막으로 사용자의 검색 토큰 생성 시, 안전성에 따른 충분히 큰 랜덤값 r, ϵ을 사용함으로써 검색 토큰과 검색 결과 간의 비연결성을 보장한다. r, ϵ이 각 η_r, η_ϵ비트, 인증 기관으로부터 발급받은 암호키의 S에 존재하는 0의 개수를 α, 검색 가능한 키워드 개수가 n일 때, 동일한 검색 토큰을 생성할 확률은 $2^{-(\eta_r + \eta_\epsilon + \alpha(n+1))}$이다. 따라서 충분히 큰 r, ϵ를 이용하여 검색 토큰 생성 시, 검색 토큰과 검색 결과 간의 비연결성을 보장한다. 이를 통해 서비스형 블록체인과 클라우드 서비스 제공자는 사용자의 검색 토큰을 통한 사용자 추적이 불가능하다.

5.4 제안 시스템 기대 효과

위와 같은 설계를 통해 안전한 검색이 가능한 서비스형 블록체인을 구성한다. 인증 기관의 인증서를 바탕으로 서비스형 블록체인을 운영하는 기업과 데이터 제공자, 데이터 요청자 간의 올바름을 인증할 수 있으며, 인증 기관이 관리하는 검색 가능 암호키를 바탕으로 데이터 요청자인 사용자는 자신의 검색에 대한 프라이버시를 보호한다. 따라서, 사용자의 위치 정보, 개인 식별 정보 등 프라이버시 보호가 필요한 정보에 대한 검색 서비스를 서비스형 블록체인을 통해 제공하고자 하는 기업은 위 기법을 바탕으로 사용자의 검색 프라이버시를 보호하며, 신뢰성 있는 서비스를 제공할 수 있다.

6. 결 론

클라우드는 기업의 요구사항에 따른 동적 자원 할당, 인프라 구축 등의 장점이 있다. 블록체인은 단일 서버의 단일 장애 지점 문제 해결, 데이터 무결성 등의 장점이 존재한다. 이러한 클라우드와 블록체인의 장점을 바탕으로 여러 클라우드 서비스 제공자는 블록체인 운용에 필요한 자원과 API 등을 제공하는 서비스형 블록체인 기능을 제공한다. 하지만 사용자의 검색 이력, 입력 정보 등을 바탕으로 상품 추천, 검색 결과 제공 등의 서비스를 서비스형 블록체인 환경에서 운영할 시, 검색 및 입력 내용에 의한 서비스 사용자의 개인 정보 유출, 검색 내용에 따른 사용자 추적 등의 프라이버시 문제가 발생할 수 있다.

[4]의 연구는 검색 가능 암호를 사용하여 컨소시엄 블록체인 기반의 이커머스 서비스 환경에서 사용자의 검색에 대한 프라이버시를 보호하며, 검색 내용과 관련성이 높은 Top-k 레코드를 반환한다. 하지만 내적 유사도를 기반으로 검색 결과인 Top-k 레코드를 반환함에 따라 검색 쿼리와 레코드의 내적 연산에 따른 동점 문제가 발생한

다. 따라서 해당 기법을 서비스형 블록체인에 적용하기 위해 코사인 유사도 기반의 Top-k 레코드 반환이 적합함을 보였으며, 이를 바탕으로 사용자의 검색에 대한 프라이버시를 보호하는 서비스형 블록체인 환경을 설계하였다. 또한, 서비스형 블록체인에 저장된 데이터의 검색 가능 암호화와 정확한 Top-k 레코드 반환을 위한 스마트 컨트랙트를 제안하였다. 본 연구를 바탕으로 사용자의 검색 프라이버시를 보호하며 신뢰성 있는 서비스형 블록체인 구축에 활용될 것으로 기대한다.

참고문헌(Reference)

[1] S. H. Yang, H. Y. Park, I. Y. Choi, and J. Liu, “How Korean Air succeeded in managing the vaccine cold chain with Amazon Managed Blockchain,” https://aws.amazon.com/ko/solutions/case-studies/innovators/korean-airlines/

[2] B. King, “Faster invoicing resolutions build stronger relationships”. https://www.ibm.com/case-studies/the-home-depot

[3] N. Andola, R. Gahlot, V. K. Yadav, S. Venkatesan, and S. Verma, “Searchable encryption on the cloud: a survey,” The Journal of Supercomputing, Vol.78, No.7, pp.9952-9984, 2022. https://doi.org/10.1007/s11227-022-04309-6

[4] Z. Guan, N. Wang, X. Fan, X. Liu, L. Wu, and S. Wan, “Achieving Secure Search over Encrypted Data for e-Commerce: A Blockchain Approach,” ACM Transactions on Internet Technology, Vol.21, No.1, pp.1-17, 2020. https://doi.org/10.1145/3408309

[5] U. Varri, S. Pasupuleti, and K. V. Kadambari, “A scoping review of searchable encryption schemes in cloud computing: taxonomy, methods, and recent developments,” The Journal of Supercomputing, Vol.76, No.4, pp.3013-3042, 2020. https://doi.org/10.1007/s11227-019-03087-y

[6] R. Zhang, R. Xue, and L. Lium “Searchable Encryption for Healthcare Clouds: A Survey,” IEEE Transactions on Services Computing, Vol.11, No.6, pp.978-996, 2018. https://doi.org/10.1109/TSC.2017.2762296

[7] J. Bader and A. L. Michala, “Searchable Encryption with Access Control in Industrial Internet of Things (IIoT),” Wireless Communications & Mobile Computing, Vol.2021, pp.1-10, 2021. https://doi.org/10.1155/2021/5555362

[8] H. B. How and S. H. Heng, “Blockchain-Enabled Searchable Encryption in Clouds: A Review,” Journal of Information Security and Applications, Vol.67, 2022. https://doi.org/10.1016/j.jisa.2022.103183

[9] H. Guo and X. Yu, “A survey on blockchain technology and its security”, Blockchain: Research and Applications, Vol.3, No.2, 2022. https://doi.org/10.1016/j.bcra.2022.100067

[10] A. H. Mohammed, A. A. Abdulateef, and I. A. Abdulateef, “Hyperledger, Ethereum and Blockchain Technology: A Short Overview,” 2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications, pp.1-6, 2021. https://doi.org/10.1109/HORA52670.2021.9461294

[11] Hyperledger Fabric, A Blockchain Platform for the Enterprise. 2020. https://hyperledger-fabric.readthedocs.io

[12] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, “Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data,” IEEE Transactions on Parallel and Distributed Systems, Vol.25, No.1, pp.222-233, 2014. https://doi.org/10.1109/TPDS.2013.45

[13] L. Chen, W. K. Lee, C. C. Chang, K. K. R. Choo, and N. Zhang, “Blockchain based searchable encryption for electronic health record sharing,” Future Generation Computer Systems, Vol.95, pp.420-429, 2019. https://doi.org/10.1016/j.future.2019.01.018

[14] A. Bas, “Market Basket Analysis Data,” https://www.kaggle.com/datasets/ahmtcnbs/datasets-for-appiori

[15] M. B. Shagor, “Market Basket Analysis Dataset,” https://www.kaggle.com/datasets/mobasshir/market-basket-analysis-dataset?select=movielens_movies.csv

◐ 저 자 소 개 ◑

장 호 빈(Hobin Jang)
2022년 서울시립대학교 수학과(이학사)
2022년~현재 고려대학교 정보보호대학원 융합보안학과(공학석사)
관심분야 : 정보보호, 데이터 보안, 프라이버시 향상 기술
E-mail : jhb0858@korea.ac.kr

천 지 영(Ji Young Chun)
1997년 이화여자대학교 수학과(이학사)
2006년 고려대학교 정보보호학과(공학석사)
2011년 고려대학교 정보경영공학과(공학박사)
2021년~현재 서울사이버대학교 빅데이터·정보보호학과 조교수
2022년~현재 서울사이버대학교 빅데이터·AI센터 부센터장
관심분야 : 데이터 보안, 인공지능, 연합학습, 프라이버시 향상 기술
E-mail : jychun@iscu.ac.kr

정 익 래(Ik Rae Jeong)
1998년 고려대학교 전산학과(공학사)
2000년 고려대학교 전산학과(공학석사)
2004년 고려대학교 정보보호학과(공학박사)
2006년~2008년 한국전자통신연구원 암호기술연구팀 선임연구원
2008년~현재 고려대학교 정보보호대학원 교수
관심분야 : 암호 이론, 프라이버시 향상 기술, 데이터베이스 보안, 생체인증
E-mail : irjeong@korea.ac.kr

노 건 태(Geontae Noh)
2008년 고려대학교 산업시스템정보공학과(공학사)
2010년 고려대학교 정보경영공학과(공학석사)
2014년 고려대학교 정보보호학과(공학박사)
2014년~2017년 고려대학교 정보보호연구원 박사후 연구원, 연구교수
2017년~현재 서울사이버대학교 빅데이터·정보보호학과 조교수
2020년~현재 서울사이버대학교 빅데이터·AI센터 센터장
관심분야 : 프라이버시 향상 기술, 데이터 보안, 블록체인, 인공지능
E-mail : gnoh@iscu.ac.kr