Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.28, NO.6, Dec. 2018 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2018.28.6.1401

암호화된 데이터에 대한 프라이버시를 보존하는 k-means 클러스터링 기법*

정 윤 송,† 김 준 식, 이 동 훈‡
고려대학교 정보보호대학원

Privacy-Preserving k-means Clustering of Encrypted Data*

Yunsong Jeong,† Joon Sik Kim, Dong Hoon Lee‡
Graduate School of Information Security, Korea University

요 약

k-means 클러스터링 알고리즘은 주어진 데이터를 비슷한 k개의 그룹으로 묶어서 시장 세분화나 의료연구 등의 다양한 분야에서 활용되고 있다. 본 논문에서는 다수의 사용자 데이터를 노출하지 않고 암호화하여 외부 서버에 저장하는 환경에서 프라이버시를 보존하는 클러스터링 알고리즘을 제안한다. 분산된 서버에 평문으로 저장된 데이터를 다자간 계산프로토콜을 기반으로 수행된 기존 클러스터링 알고리즘 연구와 비교했을 때 제안하는 기법은 모든 데이터를 안전하게 암호문으로 저장할 수 있다는 뚜렷한 장점이 있다. 데이터 간의 거리를 측정하고 비교하기 위해서 덧셈과 곱셈 연산이 가능한 완전동형암호로 데이터를 암호화한다. 프로토콜 수행과정에서 사용자 데이터의 안전성을 분석하고 통신량과 연산량을 다른 연구들과 비교한다.

ABSTRACT

The k-means clustering algorithm groups input data with the number of groups represented by variable k. In fact, this algorithm is particularly useful in market segmentation and medical research, suggesting its wide applicability. In this paper, we propose a privacy-preserving clustering algorithm that is appropriate for outsourced encrypted data, while exposing no information about the input data itself. Notably, our proposed model facilitates encryption of all data, which is a large advantage over existing privacy-preserving clustering algorithms which rely on multi-party computation over plaintext data stored on several servers. Our approach compares homomorphically encrypted ciphertexts to measure the distance between input data. Finally, we theoretically prove that our scheme guarantees the security of input data during computation, and also evaluate our communication and computation complexity in detail.

Keywords: Privacy-preserving clustering, Fully homomorphic encryption, k-means clustering

I. 서 론

현대의 마케팅과 의료연구는 서버에 수집된 수많은 고객이나 환자의 데이터에서 가치 있는 정보를 추출하여 활용한다. 마케터는 방대한 데이터에서 추출된 정보를 기준으로 시장 세분화와 표적 시장을 선정하여 소비자 행동을 예측하는 효과적인 시장 전략을 설계한다. 한편으로는 기존 환자의 정보를 활용해서

Received(10. 31. 2018), Modified(11. 27. 2018), Accepted(12. 11. 2018)
* 이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (No. 2016-6-00599, 함수서명 설계기법 및 응용기술 연구)
† 주저자, song91415@naver.com
‡ 교신저자, donghlee@korea.ac.kr(Corresponding author)

유사한 징후를 가지는 환자의 질병을 예방할 수 있다. 이러한 데이터 구획화에는 레이블(label)이 없는 데이터 집합을 서로 유사한 특성을 가지는 부분집합으로 분류하는 클러스터링(clustering) 알고리즘이 사용된다. 클러스터링 알고리즘을 이용한 시장 세분화[1,2]와 의료 연구[3,4] 등의 사회공학 연구는 활발히 이루어져 왔다. 그 예로 환자의 나이, 성별, 체중, 흡연 여부 등을 변수로 클러스터링된 결과가 당뇨병 발병 확률을 예측할 수 있다는 연구 결과[4]가 있다. 시장 분석이나 발병원인의 연구는 여러 출처(source)에서 수집되는 다수의 샘플 데이터로 클러스터링을 수행해야 유의미한 결과를 얻을 수 있다. 카드사는 다양한 온라인 상점으로부터 고객의 구매패턴뿐만 아니라 방문기록이나 주소, 직업, 나이 등을 포함하는 개인정보를 수집해서 세분화한 군집마다 고객의 요구사항을 예측할 수 있다. 한편으로는 중앙병원이나 외부 기관에서 다양한 샘플 채취를 위해서 여러 개인병원이나 종합병원에서 수집된 환자의 개인정보를 연계하여 분석결과의 정확도를 높일 수 있다. 그러나 개인정보나 의료기록과 같은 민감한 정보를 외부 서버에서 학습하면 프라이버시를 침해할 우려가 있다. 따라서 구매고객이나 환자 등의 데이터 소유주의 프라이버시를 보호하는 동시에, 외부 서버에서 의미 있는 데이터를 추출할 수 있는 클러스터링 알고리즘이 필요하다. 이때, 사용자의 프라이버시를 보호하기 위해서는 민감한 정보는 외부 서버에서 암호화하여 관리해야 한다. 또한, 암호화된 정보들을 연산하기 위해서는 완전동형암호 기법이 필요하다. 본 논문에서는 다수의 사용자 개인정보를 완전동형암호로 암호화하여 안전하게 저장하고 클러스터링을 수행하는 기법을 제안하고자 한다.

1.1 관련 연구

외부 서버에서 데이터를 노출하지 않고 데이터 마이닝을 수행하는 방법은 크게 두 가지 접근법으로 연구되었다. 첫 번째 방식은 분산된 데이터베이스가 각각의 서버에 수집된 데이터를 서로 공개하지 않고 비밀 공유(secret sharing)로 집계(aggregation)된 데이터베이스에서 클러스터링 결과를 얻는 방법이다. 둘 이상의 서버로 구성된 수평 분산 혹은 수직 분할 데이터베이스에서 각각 데이터를 제공하며 협력적으로 연산하여 각자의 데이터는 상대방에게 노출하지 않고 클러스터링된 결과만을 얻는다[5,6,7]. 클러스터링 기법 중에서 k-means 알고리즘이 가장 활발하게 연구된 주제이다. Jha 등[7]은 수평 분산 데이터베이스 환경에서 불확정 다항식 평가(oblivious polynomial evaluation)와 동형암호를 기반으로 양자 간의 k-means 클러스터링 프로토콜을 설계하였다. 이후, Yao의 가블드(garbled) 회로 평가 프로토콜[10]을 이용해 임의로 분산된 데이터베이스에 대한 클러스터링 기법이 제안되어 수평 분산 데이터베이스와 수직 분할 데이터베이스에 대한 클러스터링 연산이 일반화되었다[6]. 2007년에는 Bunn 등[5]이 Yao의 가블드 회로를 사용하지 않고 임의로 분산된 데이터베이스에 대한 k-means 클러스터링 알고리즘을 설계하고 호기심을 가진(honest-but-curious) 공격자에 대한 프로토콜의 안전성을 증명했다. 이전 연구[5,6,7]에서는 암호화되지 않은 평문 데이터가 서버에 저장된 환경이므로 사용자의 데이터 프라이버시를 보존하지 않는다. 그러므로 분산된 데이터베이스 모델은 여러 출처에서 수집된 사용자의 개인정보가 포함된 데이터를 학습하는 클러스터링 알고리즘에 적절하지 않다.

두 번째 방식은 Jaschke 등[8]의 변형 k-means 알고리즘과 같이 완전동형암호로 암호화된 상태로 외부 서버에 저장된 데이터에 대한 클러스터링 연산을 수행한다. 그러나 하나의 공개키로 암호화된 데이터에 대한 연산만 가능하므로 1명의 사용자가 외부 서버에 클러스터링 연산을 아웃소싱(outsourcing)하는 환경에 적합하다. 이는 여러 명의 사용자로부터 수집된 데이터를 학습하기에는 적절하지 않은 모델이다. 또한, 현재까지 연구된 동형암호시스템에서는 클러스터링을 위해 필요한 암호문 간의 나눗셈 연산이 실용적이지 않다. 따라서 Jaschke 등[8]은 군집의 평균을 계산하는 과정에서 암호문을 정해진 상수로 나누는 회로를 미리 만들어서 암호문 간의 나눗셈 연산 없이 변형하여 설계하였다. 이 과정에서 군집의 평균을 근삿값으로 계산하므로 기존 알고리즘보다 반복하는 횟수가 훨씬 증가한다. 이 기법은 서버나 사용자 간의 온라인 연결이 필요가 없지만, 총 수행시간이 길어서 실제로 활용하기는 힘들다.

1.2 기여도

본 논문에서는 외부 서버에 다수의 사용자 데이터를 암호화하여 저장한 환경에서 서버로부터 사용자의

프라이버시를 보존하는 클러스터링 연산을 수행하는 사용자 프라이버시 보존 k-means 클러스터링 프로토콜(User Privacy-preserving k-means Clustering Protocol, UP-kCP)을 제안한다. 다수의 사용자에 대한 암호화된 데이터의 클러스터링 기법을 설계하기 위해서는 사용자마다 개별적으로 키를 발급하는 방법과 신뢰하는 3자(Trusted Third Party, TTP)의 키를 이용하는 방법으로 두 가지 방식이 가능하다. 첫 번째로는 여러 명의 사용자로부터 받은 암호문 간의 연산을 지원하는 멀티 키 완전동형암호(Multi-key Fully Homomorphic Encryption, MK- FHE)[20]을 사용하여 데이터를 암호화하는 방식이다. 그러나 MK-FHE 기법을 사용하면 다른 사용자의 키로 연산할 때 키의 소유자와 매번 다자간 계산 프로토콜(Multi-party Computation, MPC)을 수행하여 통신량에서 오버헤드(overhead)가 발생한다. 따라서 본 논문에서는 MK-FHE를 사용하지 않고, Nikolanko 등[9]의 암호문에 대한 선형 능선 회귀를 위한 시스템모델을 인용하여 평가 서버(evaluating server)와 상호작용하는 CSP(Cryptographic Service Provider)를 추가한 시스템모델을 제안한다. UP-kCP에서는 CSP의 공개키를 다수의 사용자가 사용한다.

Fig 1.과 같이 제안하는 시스템모델에서 평가 서버와 CSP는 암호화된 사용자 데이터에서 군집 중심(centroid)을 계산한다. 신뢰하는 3자인 CSP는 완전동형암호시스템에서 비효율적으로 구현되는 암호문 간의 크기 비교와 나눗셈 연산을 라그랑주 보간 다항식과 이진 게이트로 수행하면서 Jaschke 등[8]의 변형 k-means 알고리즘과 비교해 효율적인 클러스터링이 가능하다.

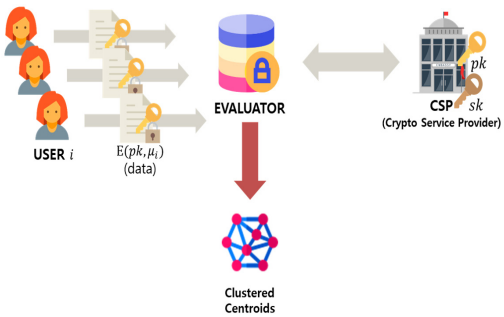

Fig. 1. System model of User Privacy-preserving k-means Clustering Protocol

사용자는 암호화된 개인정보를 제공하고 취합된 데이터를 활용하는 서비스에 동의하는 경우에 자신의 데이터를 평가 서버라는 외부 서버에 저장한다. 각각의 사용자는 CSP의 공개키로 암호화된 데이터를 평가 서버에 업로드한다. 이후의 모든 연산은 평가 서버와 CSP 사이에서 이루어져서 사용자의 추가적인 통신과 연산이 필요하지 않다. 평가자는 CSP와의 상호작용과정을 통해서 클러스터링을 수행하고 그 결과로 출력한 군집의 대푯값을 서비스에 활용한다. UP-kCP는 사용자 데이터의 집합과 군집화 중심에 대한 프라이버시를 보장한다. CSP는 입력 데이터 집합과 군집화 결과에 대해서 어떠한 정보도 알아낼 수 없고, 평가 서버는 임곗값 이하의 사용자와 공모하는 경우에 군집화 중심에 대한 제한된 정보 외에는 어떠한 정보도 알아낼 수 없다. 다시 말해서, 제안하는 기법은 외부 서버에서 프라이버시를 침해하지 않고 암호화된 여러 명의 개인정보를 활용해서 클러스터링 결과를 추출할 수 있다.

본 논문의 구성은 다음과 같다. II장에서는 제안하는 기법의 이해에 필요한 배경지식을 설명하고, III장에서는 제안하는 시스템모델과 위협모델을 정의한다. IV장에서는 제안하는 UP-kCP의 기법을 단계별로 설명한다. V장에서는 제안하는 기법을 위협모델에서 안전성을 분석하고, 기존 연구와 UP-kCP의 연산량 및 통신량을 비교 분석하여 VI장에서 결론을 맺는다.

II. 배경지식

2.1 k-means 클러스터링

k-means 클러스터링은 d차원 벡터 데이터의 집합을 상수 k개의 클러스터(cluster)라는 부분집합으로 분할하는 비지도 학습 알고리즘이다. 각각의 데이터는 클러스터 내 거리의 합을 최소화하는 가장 가까운 클러스터에 속한다. 본 논문에서는 두 벡터 간의 거리를 측정하고 거리의 최솟값을 계산할 때 맨해튼 거리(Manhattan distance)를 사용한다. 평문 공간의 크기가 암호문의 크기 비교의 시간 복잡도에 영향을 주기 때문에 거듭제곱한 유클리드 거리의 크기보다 값이 작은 맨해튼 거리를 선택했다. 클러스터링 연산을 할 때 초기 중심은 입력 데이터 중에서 랜덤으로 선택한다고 가정한다. 여기서는 간단하게 k-means 클러스터링 알고리즘의 개념을 설명하고

자세한 내용은 MacQueen[19]의 연구를 따른다.

2.2 라그랑주 보간다항식(Lagrange Interpolating Polynomial)

라그랑주 다항식 보간법은 어떠한 x_i도 서로 같지 않은 $(n+1)$개의 점 (x_i,y_i)을 보간하는 최대 n차 다항식 P를 계산한다. 그 결과로 얻어지는 다항식은 $P(x)=(a_nx^n+\cdots+a_1x+a_0)\bmod p$ 의 형태이다.

정의 1. [라그랑주 보간다항식] 어떠한 x_i도 서로 같지 않으면서 $y_i=P(x_i)\bmod p$를 만족하는 $n+1$개의 점 $(x_0,y_0),\cdots(x_n,y_n)$이 주어질 때, 라그랑주 보간다항식은 $P(x)=\sum_{i=0}^{n}\prod_{\substack{j\neq i\\0\le j\le n}}\frac{x-x_j}{x_i-x_j}y_i(\bmod p)$ 로 정의한다.

2.3 Ring Learning with Errors(RLWE) 문제

RLWE 문제는 Lyubashevsky 등[21]이 고안한 문제로 벡터 대신 환(Ring)의 원소를 이용하는 LWE(Learning with Errors)의 대수적 변종이다. $Z_p[x]$가 소수 p에 대해 다항식의 모든 계수가 모듈로 p를 만족하는 다항식 환이고, $\Phi_m(x)$가 m차 원분다항식(cyclotomic polynomial)일 때, RLWE 문제는 다음과 같이 정의한다.

정의 2. [RLWE 문제] 보안상수 λ와 m차 원분다항식 $\Phi_m(x)$에 대해서 환 $R_p=Z_p[x]/\Phi_m(x)$로 정의한다. $p=p(\lambda)\ge 2$를 만족하는 정수와 랜덤한 원소 $\boldsymbol{s}\in R_p$, R_p에 대한 분포 $\chi=\chi(\lambda)$에 대해서 $A_{\boldsymbol{s},\chi}^{(p)}$를 균등하게 랜덤한 원소 $\boldsymbol{a}\leftarrow R_p$와 $\boldsymbol{e}\leftarrow\chi$를 선택하여 만든 $(\boldsymbol{a},[\boldsymbol{a}\cdot\boldsymbol{s}+\boldsymbol{e}]_p)$ 의 분포로 정의한다. RLWE 문제는 $A_{\boldsymbol{s},\chi}^{(p)}$와 균등 분포 $U(R_p^2)$를 구분하는 문제이다.

정의 3. [RLWE 가정] 다항식 시간 안에 RLWE 문제를 풀 수 있는 알고리즘 D가 존재할 때, D는 $A_{\boldsymbol{s},\chi}^{(p)}$와 $U(R_p^2)$의 분포가 같으면 1을 출력하고 다르면 0을 출력한다. D의 문제를 풀어내는 이점이 충분히 작다면(negligible) RLWE 문제는 풀기 어렵다고 정의하고 이때 D의 이점을 다음과 같이 정의한다.

$$Adv_D^{RLWE}=\left|\Pr\left[U(R_p^2),U(R_p^2)\right]=1-\Pr\left[D(A_{\boldsymbol{s},\chi}^{(p)}),U(R_p^2)\right]=1\right|\le\epsilon$$

2.4 완전동형암호(Fully Homomorphic Encryption, FHE)

완전동형암호는 다음과 같이 네 개의 알고리즘 $(Setup,Enc,Dec,Eval)$으로 구성되어 있다.

- $Setup(1^\lambda)\rightarrow(pp,pk,sk)$: 보안상수 1^λ를 입력받고 시스템 전체에 사용되는 공개파라미터 pp와 공개키 pk, 개인키 sk를 출력한다.
- $Enc(pk,\boldsymbol{\mu})\rightarrow C$: 메시지 $\boldsymbol{\mu}$와 공개키 pk를 입력받고 암호문 C를 출력한다.
- $Dec(sk,C)\rightarrow\boldsymbol{\mu}$: 암호문 C와 개인키 sk를 입력받고 복호화한 메시지 $\boldsymbol{\mu}$를 출력한다.
- $Eval(P,C)\rightarrow C_p$: 평가할 연산 P와 암호문 C를 입력받아 $P(\boldsymbol{\mu})\leftarrow Dec(sk,C_p)$를 만족하는 평가한 암호문 C_p를 출력한다.

본 논문에서는 평문 메시지가 정수 또는 비트로 인코딩된 두 가지 방식으로 클러스터링 기법을 제안한다.

- 정수기반 완전동형암호

정수기반 데이터는 레벨을 가지는(leveled) 완전동형암호인 BGV 동형암호시스템[12]을 사용한다. BGV 암호시스템은 RLWE(Ring Learning with Error) 가정하에 의미론적으로 안전(semantically secure)하다고 증명되었다. HElib[13]은 재선형화, 부트스트래핑(bootstrapping), 암호문 포장기법으로 BGV 암호시스템을 구현하고 최적화한 암호라이브러리로 동형암호구현에서 가장 일반적으로 활용되고 있다.

$R_p:=Z_p[x]/\Phi_m(x)$의 원소 평문 $\boldsymbol{\mu_1},\boldsymbol{\mu_2}$에 대한 덧셈과 곱셈 연산과 HElib에서의 다항식 P에 대한 평문 $\boldsymbol{\mu}$의 평가를 다음과 같이 정의한다.

$$Dec(sk,Enc(pk,\boldsymbol{\mu}_1)+Enc(pk,\boldsymbol{\mu}_2))=\boldsymbol{\mu}_1+\boldsymbol{\mu}_2$$
$$Dec(sk,Enc(pk,\boldsymbol{\mu}_1)*Enc(pk,\boldsymbol{\mu}_2))=\boldsymbol{\mu}_1{}^*\boldsymbol{\mu}_2 \quad (1)$$

$$Dec(sk, Eval(P,(Enc(pk, \boldsymbol{\mu})) = Eval(P, \boldsymbol{\mu}) \quad (2)$$

• 비트기반 완전동형암호

비트기반 데이터는 Chillotti 등이 제안한 완전동형암호시스템을 사용한다[14]. Chillotti 등[14]은 LWE 가정하에 의미론적으로 안전하다고 증명된 GSW(Gentri-Sahai-Waters) 암호시스템[15]을 기반으로 외적 연산으로 빠르게 부트스트래핑할 수 있는 완전동형암호알고리즘을 제안했다. TFHE (Torus Fully Homomorphic Encryption) [16]은 이 연구를 평문이 비트 문자열일 때 구현한 라이브러리이다. TFHE 라이브러리는 원본 데이터에서 어떠한 정보도 노출하지 않고 임의의 이진 회로의 평가를 지원한다. NAND, OR, AND, XOR 등의 이진 게이트와 NOT, MUX 게이트를 지원한다.

2.5 완전동형암호로 암호화된 데이터의 정렬

Harika 등[18]은 동형암호로 암호화된 데이터의 정수 간 비교 혹은 비트 간 비교의 두 가지 방법으로 데이터를 정렬하는 방법을 제안했다. 정수 간 암호문 정렬은 미리 계산한 라그랑주 보간다항식을 이용하여 두 암호문의 순서를 교환하는 방식으로 설계했다. l 비트 정수의 암호문에서 정수 간 크기 비교는 한 번의 교환이 필요하고, 비트 간 크기 비교는 l번의 교환이 필요하다.

2.5.1 정수 간 크기 비교

평문이 l비트 크기의 정수인 암호문의 크기 비교는 라그랑주 보간다항식을 이용하여 설계되었다[18]. 암호문의 크기 비교를 위해서는 $0 \le x_i \le 2^l-1$이면 $y_i = 0$, $-2^l+1 \le x_i \le -1$이면 $y_i = 1$를 만족하는 $2^{l+1}-1$개의 점 (x_i, y_i)를 보간하는 라그랑주 다항식 $P(x)$가 필요하다. 이때의 위수 p는 $p > 2^{l+1}$을 만족하는 소수이다. 암호문을 오름차순으로 정렬하는 알고리즘 $SORT(C_a, C_b)$는 다음과 같이 정의한다.

• $SORT(C_a, C_b) \rightarrow (C_x, C_y)$:

① 두 암호문의 차 $C_{a-b} = C_a - C_b$ 를 계산한다.

② 라그랑주 보간다항식 $P(x)$에 대해 암호문 C_{a-b}를 평가한 $C_z \leftarrow Eval(P, C_{a-b})$를 계산한다.

③ $C_x := C_b + C_z \cdot C_{a-b}$, $C_y := C_a + C_b - C_x$를 계산하고 (C_x, C_y)를 출력한다.

두 암호문 C_a, C_b가 오름차순으로 정렬되어 $Dec(sk, C_a) \le Dec(sk, C_b)$를 만족하면 $Dec(sk, C_z) = 0$이고, 두 암호문이 내림차순으로 정렬되어 $Dec(sk, C_a) > Dec(sk, C_b)$일 때 $Dec(sk, C_z) = 1$을 만족한다. 따라서 출력한 두 암호문은 $Dec(sk, C_x) \le Dec(sk, C_y)$를 만족한다. $SORT$는 위수 p가 작을수록 보간할 점의 개수가 작으므로 라그랑주 보간다항식의 차수가 낮아서 효율적으로 수행된다.

2.5.2 비트 간 크기 비교

이진수로 표현된 평문을 각각의 비트에 대응하도록 암호화한 경우 비트 간의 크기 비교를 한다. 두 개의 l비트 암호문 C_a와 C_b가 주어졌을 때, 두 암호문의 차 $C_{sub} = C_a - C_b$는 2의 보수법으로 계산한다. C_{sub}의 최상위비트를 c_{msb}라고 할 때 c^i를 C의 i번째 비트라고 한다. 두 암호문의 차가 양수면 $D(c_{msb}) = 1$이고 음수면 $Dec(sk, c_{msb}) = 0$을 만족한다. 정수 간의 비교와 마찬가지로 각 비트마다 교환기법으로 방식으로 정렬함수 $SORT(C_a, C_b)$라고 정의한다.

• $SORT(C_a, C_b) \rightarrow (C_x, C_y)$:

① 2의 보수에서 뺄셈법으로 두 암호문의 차 $C_{sub} = C_a + (2's\, complement\, C_b)$ 를 계산하고 최상위비트를 c_{msb}로 한다.

② 정수 $i (0 \le i < l)$에 대해 $c_x^i = c_{msb} \cdot c_a^i + (1 - c_{msb}) \cdot c_b^i$와 $c_y^i = (1 - c_{msb}) \cdot c_a^i + c_{msb} \cdot c_b^i$를 계산하고 $C_x = (c_x^0, \cdots, c_x^{l-1})$, $C_y = (c_y^0, \cdots, c_y^{l-1})$를 출력한다.

III. 시스템모델 및 위협모델

3.1 시스템모델

Nikolanko 등의 프라이버시를 보존하는 선형 능선회귀 학습[9]의 시스템모델에서 CSP의 개념을 인용해서 UP-kCP를 위한 시스템모델을 제안한다.

각각의 사용자 u는 d차원의 데이터 $\boldsymbol{\mu}_u \in R_p^d$를

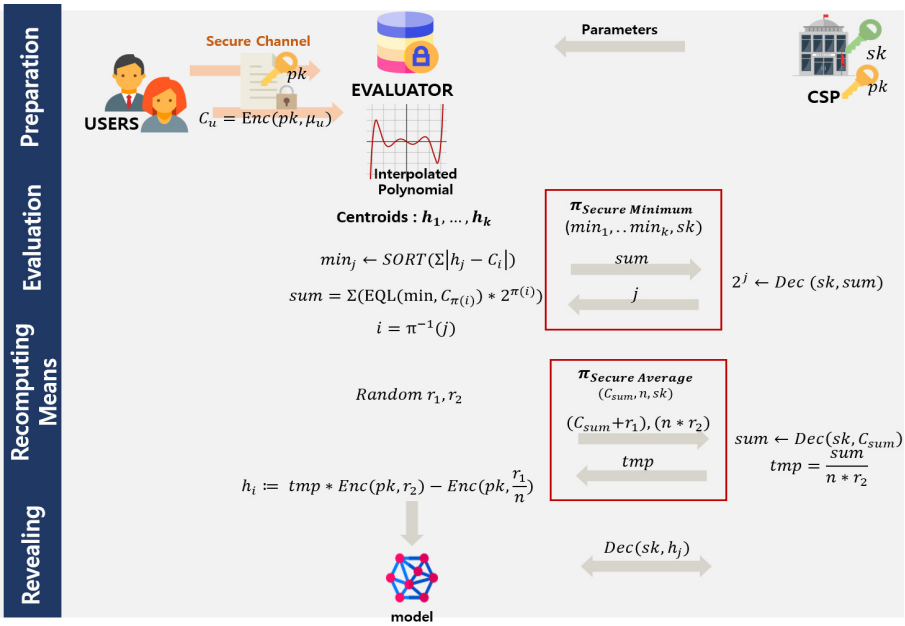

Fig. 2. User Privacy-preserving k-means Clustering Protocol

암호화하고 평가자는 각각의 샘플에서 가장 가까운 군집의 중심 $\boldsymbol{h}_1,\cdots,\boldsymbol{h}_k$를 연산한다. CSP와 평가자 사이의 상호작용으로 클러스터링 연산이 이루어지므로 사용자는 데이터를 평가자 서버에 처음 업로드할 때를 제외하고는 온라인 상태를 유지할 필요가 없다. 시스템에 포함된 참가자는 다음과 같다.

• 사용자 : 사용자 u는 개인정보가 포함된 데이터 $\boldsymbol{\mu}_u$를 가지고 있다. CSP의 공개키로 평가자 서버에 자신의 데이터를 암호화하여 전송한다.

• 평가자 : 평가자는 암호화된 데이터 학습의 결과로 각 데이터의 레이블과 군집의 대푯값을 얻는다. 각 사용자의 프라이버시를 침해하지 않고 전체 데이터의 특성을 파악하는 것이 목표이다.

• CSP : CSP는 완전동형암호시스템을 위한 공개 파라미터를 설정하고 사용자와 평가자에게 그 값을 전송한다. CSP는 평가자가 모델을 학습하는 동안 프라이버시 침해 우려 없이 완전동형암호시스템에서 비효율적인 연산을 대신하는 서비스를 제공한다.

UP-kCP는 다음과 같은 네 단계의 순서로 진행된다. 전체적인 과정은 Fig 2.에서와 같다.

1. **준비단계** : CSP는 암호화에 필요한 시스템 파라미터를 설정하고 다른 참여자에게 그 값을 전송한다. 사용자는 CSP가 제공하는 시스템 파라미터에 따라 데이터를 암호화하여 평가자에게 전송한다
2. **평가단계** : 평가자는 보로노이(Voronoi) 다이어그램에 따라 각각의 관측값(observation)을 분할하고 클래스 레이블을 계산하기 위해서 CSP에 복호화 질의를 한다.
3. **중심 재계산단계** : 평가자는 각 레이블에 속하는 암호화된 데이터의 합을 계산하고 평균값을 얻기 위해서 CSP에 복호화 질의를 한다. 모든 관측값의 레이블이 변하지 않을 때까지 2~3단계를 반복한다.
4. **노출단계** : 평가자가 정확한 분석을 위해서 각 군집의 중심을 알고 싶을 때 선택적으로 수행하는 단계이다. 평가자는 CSP에게 복호화 질의를 해서 클러스터 중심의 값을 출력으로 가진다.

3.2 위협모델

평가자와 CSP는 모두 호기심을 가진(honest-but-curious, or semi-honest) 공격자라고 가정한다. 호기심을 가진 공격자는 기능성을 제공하기 위해서 프로토콜을 충실히 따르지만, 추가로 가능한 정보를 얻기 위한 시도를 하는 공격자이다. 또한, 평가자와 CSP는 서로 공모하지 않는다고 가정한다. 이러한 가정은 이전 연구[17]에서도 클라우드 컴퓨팅 비즈니스 업계에서 합리적이라고 고려되어왔다. 평가

자가 일부의 사용자와 공모하더라도 클러스터링 결과를 제외하고는 사용자의 데이터에 대한 어떠한 정보도 알 수 없어야 한다. 호기심을 가진 공격자 모델에서 다음 두 가지 보안 요구사항을 충족해야 한다.

• **사용자 데이터 집합 $D=\{\boldsymbol{\mu}_u | u \in [1,n]\}$ 에 대한 프라이버시** : 평가자와 CSP 모두 사용자의 데이터에 대해 클러스터링 결과를 제외하고는 어떠한 정보도 알 수 없다. 평가자가 일부의 사용자와 공모하더라도 나머지 사용자의 데이터에 대한 정보를 알아낼 수 없다.

• **클래스 레이블에 대한 프라이버시** : CSP는 클러스터링된 각각의 데이터의 레이블에 대한 정보를 알 수 없다.

더 나아가서는 CSP가 완전동형암호시스템 파라미터 설정 이후에 사용자와 평가자에게 정당한 공개키와 인증서가 전송되었다고 가정한다.

IV. 사용자 프라이버시를 보존하는 k-means 클러스터링 프로토콜(UP-kCP)

4.1 준비 단계

1) CSP는 완전동형암호시스템의 파라미터와 학습할 데이터 정보에 대한 파라미터를 설정한다.
CSP는 데이터의 차원 d, 클러스터의 개수 k, 보안상수 λ와 소수 p를 선택한다. 집합 S의 크기를 $|S|$라 할 때, 환 R_p에 대해서 평문 데이터의 최댓값 M과 평문 데이터의 최대 개수 N을 $N>M$, $N \cdot M < |R_p|$를 만족하는 정수 범위에서 선택한다. 비트기반 인코딩을 사용하고자 할 때는 추가적으로 평문 데이터의 비트 수 $l = \lfloor \log p \rfloor + 1$을 선택한다. CSP는 보안상수 λ를 입력으로 완전동형암호시스템의 $(pp, pk, sk) \leftarrow Setup(1^{\lambda})$ 에서 출력된 공개키 pk와 비밀키 sk를 생성하고 공개파라미터를 (d, k, p, N, M, pp, pk)로 한다.

2) CSP는 참여자와 보안 채널(secure channel)을 생성해서 공개파라미터와 CSP의 인증서를 모든 참여자에게 전송한다.

3) 사용자 u는 공개파라미터에 따라 자신의 데이터 $\boldsymbol{\mu}_u$를 암호화하여 평가자에게 보안 채널로 전송한다. 사용자는 정수단위 인코딩을 선택하는 경우에는 HElib[13]로, 비트 단위 인코딩($p=2$)을 선택하는 경우에는 TFHE[16]으로 암호화해서 그 암호문을 보안 채널을 사용해서 평가자에게 전송한다. 데이터의 각 차원은 특징(feature) 벡터에 대응하고 개별적으로 암호화한다. 따라서 데이터 $\boldsymbol{\mu}_u = (\mu_{u,1}, \cdots, \mu_{u,d}) \in R_p^d$는 d차원의 암호문 $\boldsymbol{C}_u = (c_{u,1}, \cdots c_{u,d})$에 대응한다. 학습하려는 데이터가 n개일 때, 정수기반 인코딩에서는 암호문은 $n \cdot d$개이고 비트 기반 인코딩에서는 $n \cdot d \cdot l$개이다. 평가자가 공개파라미터에 해당하는 라그랑주 보간다항식 P를 미리 계산해두면 모든 참여자는 다음 단계를 위한 준비를 마친다.

4.2 평가 단계

1) 평가자는 주어진 n개의 암호문 $\boldsymbol{C}_1, \cdots \boldsymbol{C}_n$에서 랜덤으로 k개를 선택하여 초기 중심 $\boldsymbol{h}_1, \cdots \boldsymbol{h}_k$으로 한다.

2) 평가자는 n개의 암호문에 대해서 암호문 $\boldsymbol{C}_u = (c_{u,1}, \cdots, c_{u,d})$과 각 중심 $\boldsymbol{h}_j = (h_{j,1}, \cdots, h_{j,d})$ 사이의 맨해튼 거리 $C_{\triangle_{u,j}} = \sum_{i=1}^{d} |h_{j,i} - c_{u,i}|$를 계산한다. 두 암호문의 차의 절댓값의 연산은 Harika 등[18]의 정렬알고리즘을 이용한다. 정수단위의 암호문의 절댓값과 비트단위의 암호문의 절댓값은 다음과 같이 계산한다.

• 정수단위의 암호문의 절댓값

$[0, \frac{p}{2}]$ 범위에 속하는 정수는 환 R_p에서 계수 $[0, \frac{p}{2}]$에 대응하고, $[-\frac{p}{2}, 0)$ 범위에 속하는 정수는 환 R_p에서 계수 $[\frac{p}{2}+1, p-1)$로 대응한다. 따라서 데이터의 최댓값인 M이 $N \cdot M < |R_p|$, $N > M$라는 조건을 만족하므로 음의 정수보다 양의 정수의 암호문이 R_p위에서 더 작은 값을 가진다. 따라서 암호문 C의 평문의 절댓값에 대응하는 암호문은 $(C_x, C_y) \leftarrow SORT(C, -C)$에서 C_x와 같다. 평문의 값이 더 작은 C_x가 C의 절댓값에 해당한다.

• 비트단위의 암호문의 절댓값

비트단위의 암호문의 절댓값은 암호문 C의 최상위비트 c_{msb}를 이용해서 다음과 같이 계산할 수 있다. 부호가 있는(signed) 이진수 표기에서 최상위비트는 양수이면 0으로, 음수라면 1로 부호를 표시한

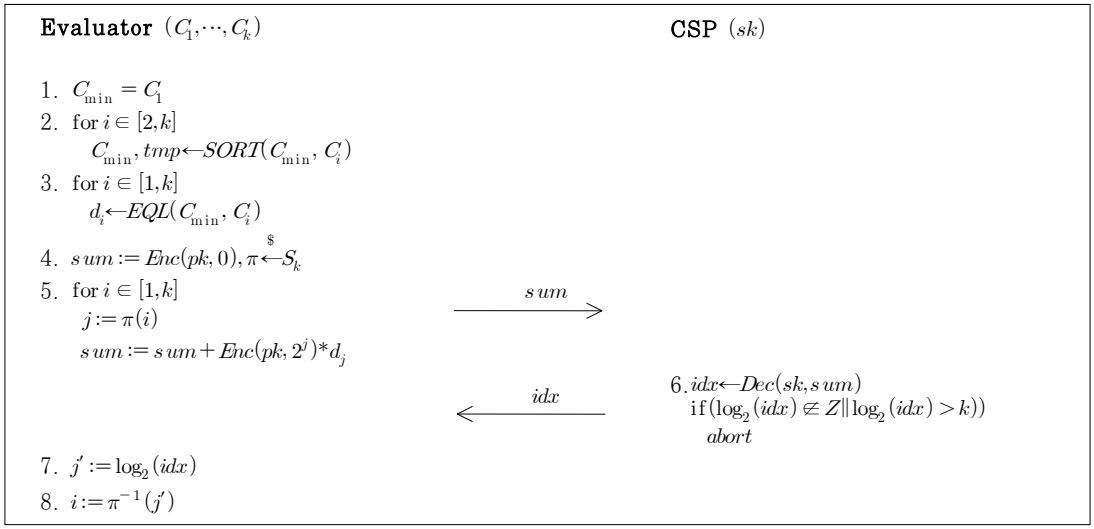

Fig. 3. Secure Minimum Protocol

다. 따라서 암호문 $C=(c^0,\cdots c^{l-1})$에 대해서 각 비트에 대해 $c_{msb}\times(-c^i)+(1-c_{msb})\times c^i$를 연산하면 $2l$번의 AND 연산과 l번의 OR 연산으로 암호문의 절댓값을 구할 수 있다. 이에 따라 Jaschke 등[8]의 변형 k-means 알고리즘에서 절댓값을 구하기 위해 사용한 멀티플렉서보다 효율적인 절댓값 연산 회로를 구현할 수 있다.

3) 평가자와 CSP는 $1\le u\le n$에 대해 각각 $C_{\triangle_{u,1}},\cdots,C_{\triangle_{u,k}}$와 개인키 sk를 입력으로 최소거리 검색 프로토콜(Secure minimum protocol)을 수행하여 각 데이터에 클래스 레이블을 붙인다. Fig 3.은 안전한 최소거리 검색 프로토콜을 설명한다. 프로토콜 수행 결과로 평가자는 주어진 k개의 암호문 중에서 최솟값의 평문을 가지는 암호문에 해당하는 인덱스 i를 C_{μ}에 해당하는 클래스 레이블로 결정한다.

최소거리 검색 프로토콜은 정렬함수 $SORT$를 이용해서 k개의 맨해튼 거리 $C_{\triangle_{u,1}},\cdots C_{\triangle_{u,k}}$중에서 가장 짧은 $C_{\min_u}$를 검색한다. 그러나 정렬함수 $C_x, C_y \leftarrow SORT(C_a, C_b)$에서 C_x, C_y가 C_a, C_b와 다른 평가된 암호문이므로 평가자는 결과로 얻은 $C_{\min_u}$가 몇 번째 맨해튼 거리에 해당하는 $C_{\triangle_{u,j}}$인지 알 수 없다. 따라서 평가자는 CSP에 복호화 질의를 통해서 몇 번째 군집 중심까지 거리가 최솟값에 해당하는지 확인하는 과정이 필요하다. 이 과정에서 평가자는 질의를 통해서 몇 번째 인덱스에 속하는지의 정보를 제외하고는 어떠한 추가적인 정보도 얻지 못해야 한다. 또한, CSP는 이 상호작용을 통해서 사용자의 데이터에 대한 어떠한 정보도 알 수 없다. 최소거리 검색 프로토콜은 다음과 같이 단계적으로 이루어진다. 이 때, k개의 맨해튼 거리는 모두 서로 다르다고 가정한다.

① 첫 번째 암호문 C_1을 최솟값의 평문을 가지는 암호문 $C_{\min}$이라고 한다.

② 정렬함수 $SORT(C_{\min}, C_i)$를 $k-1$번 반복해서 최솟값의 평문을 가지는 암호문을 찾는다.

③ 검색한 $C_{\min}$이 i번째 암호문과 일치하면 1을 출력하는 함수 $EQL(\min, C_i)$의 출력 d_i를 구한다. 일치함수 $d_i \leftarrow EQL(C_a, C_b)$는 정수단위의 암호문과 비트단위의 암호문에 대해 각각 다음과 같이 정의한다.

- 정수단위의 암호문의 일치함수

Harika 등[18]의 라그랑주 다항식 보간법을 따른다. 라그랑주 보간다항식 $P(x)$는 모듈로 p상에서 $0<x_i\le 2^l-1$ 또는 $-2^l+1\le x_i<0$이면 $y_i=0$, $x_i=0$이면 $y_i=1$를 만족하는 $2^{l+1}-1$개의 점 $(x_0,y_0),\cdots(x_n,y_n)$를 보간한다. 두 암호문의 평문값이 일치하는지 확인하는 일치함수 $EQL(C_a, C_b)$는 다음을 만족한다.

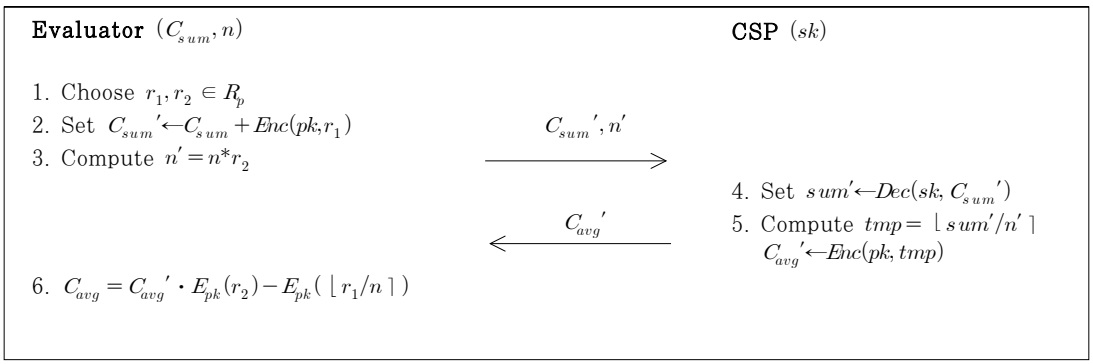

Fig. 4. Averaging Protocol

$$Dec(sk, EQL(C_a, C_b)) = Dec(sk, Eval_P(C_a - C_b)) = \begin{cases} 1 \ (\text{if } D(C_a) = D(C_b)) \\ 0 \ (o.w.) \end{cases} \quad (3)$$

• 비트단위의 암호문의 일치함수

비트단위의 암호문 일치함수는 라그랑주 보간다항식과는 다른 방법으로 설계하였다. 자세한 알고리즘은 Table 1.과 같다. $C_{\min}$과 C의 j번째 비트의 평문이 같다면, $c_{\min}^j$과 c^j를 XOR한 결과는 0이고 다를 경우에 1이다. 각 비트에 두 암호문을 XOR 연산한 결과를 OR 연산하면 모든 비트에 해당하는 평문이 같을 때 0이고, 나머지 경우에는 1을 출력한다. 그 결과를 부정하면 두 암호문의 평문이 일치할 경우 1에 해당하는 암호문을 출력하고, 일치하지 않는 경우 0에 해당하는 암호문을 출력한다.

④ 인덱스의 합을 나타내는 변수 sum으로 둔다. 평가자는 레이블 인덱스의 집합을 $I_k = \{1, \cdots, k\}$로 정의할 때, 조합함수 $\pi : I_k \rightarrow I_k$를 랜덤으로 선택한다.

⑤ 모든 인덱스 i에 대해서 $j = \pi(i)$일 때, 일치함수의 출력 d_j와 2의 거듭제곱의 합 $sum = \sum_{j=1}^{k}(d_j * Enc(pk, 2^j))$을 계산한다. 비트단위의 암호문에서는 2의 거듭제곱을 쉬프트(shift) 연산으로 간단하게 표현할 수 있다. 평가자는 sum을 CSP에게 복호화 질의를 한다.

⑥ CSP는 sum을 복호화한 idx를 평가자에게 전송한다. $\log_2(idx)$가 인덱스의 최댓값 k 초과이거나 $\log_2(idx)$가 정수가 아닌 경우에는 idx가 2의 거듭제곱의 꼴이 아니므로 정당한 복호화 질의가 아니므로 CSP는 idx를 전송하지 않고 프로토콜을 중단한다.

⑦ CSP는 $j' := \log_2(idx)$를 계산한다. idx는 최소 맨해튼 거리 $C_{\min}$과 일치하는 암호문의 인덱스 j'로 거듭제곱한 값 $2^{j'}$와 같다.

⑧ 평가자는 j'의 조합의 역 $\pi^{-1}(j')$을 계산하여 인덱스 i를 출력한다.

Table 1. Bit-wise equality check algorithm

Algorithm EQL$(C_{\min}, C)$
Input: $C_{\min} = (c_{\min}^1, \cdots, c_{\min}^l), C = (c^1, \cdots c^l)$
Output: d_i $\begin{cases} Dec(sk, d_i) = 1 \ (\text{if } Dec(sk, C_{\min}) = Dec(sk, C)) \\ Dec(sk, d_i) = 0, (o.w) \end{cases}$
1 $tmp := Enc(pk, 0);$
2 for $j \in [1, l]$
3 $tmp := tmp \ OR (c_{\min}^j \ XOR c^j);$
4 $return \ NOT(tmp);$

위의 과정은 n개의 데이터에 대해 병렬적으로 수행할 수 있으므로 CSP에게 복호화 질의는 n번 온라인 상태를 유지하여 수행하면 된다.

4.3 중심 재계산 단계

1) 평가자는 각 클래스 j에 속한 데이터의 개수를 n_j라고 한다.

2) 평가자는 각 클래스 j에 속하는 모든 암호문의 합을 구해서 C_{sum_j}라고 한다. 평가자는 C_{sum_j}과 n_j, CSP는 개인키 sk를 입력으로 평균계산 프로토콜(Averaging protocol)에서 군집의 중심의 값

C_{avg_j}를 출력한다.

새로운 군집의 중심 평균은 C_{sum_j}/n_j와 같이 암호문의 나눗셈 과정을 포함한다. 하지만 완전동형암호에서 나눗셈은 이론적으로 회로로 구현되나 비효율적인 연산이다. 또한, n_j의 R_p상의 역을 곱하는 것은 평문이 나누어떨어지지 않으면 정수 상에서 나눗셈과 전혀 다른 결과를 출력한다. 따라서 이 과정에서 평균을 정수로 반올림한 값을 구하기 위해서는 CSP에 복호화 질의가 필요하다. 평균계산 프로토콜은 Fig 4.에서와 같이 다음의 순서로 진행한다.

① 평가자는 $0 \le r_1 < M, 0 < n \times r_2 < N$를 만족하는 난수 $r_1, r_2 \in R_p$를 선택한다.
② 평가자는 입력받은 C_{sum}에 암호화한 난수 r_1을 더해 C_{sum}'으로 한다.
③ 평가자는 암호문의 개수 n에 난수 r_2를 곱한 값을 n'으로 하고 CSP에게 C_{sum}'과 n'을 전송한다.
④ CSP는 C_{sum}'을 개인기 sk로 복호화한 메시지를 sum'으로 한다.
⑤ CSP는 sum'/n'을 계산한 후 반올림하여 암호화한 C_{avg}'를 평가자에게 전송한다.
⑥ 평가자는 C_{avg}'에서 자신이 선택한 난수 r_1, r_2를 제거한 C_{avg}를 평균값으로 출력한다.

모든 암호문의 레이블이 변화하지 않을 때까지 4.2의 평가단계와 4.3의 중심 재계산 단계를 반복한다. 군집 내 거리의 합이 최소화될 때 각 데이터는 가장 가까운 군집에 할당된다. 이 단계까지 평가자는 연산 결과로 얻은 데이터의 레이블을 제외하고는 데이터에 대한 어떠한 정보도 알 수 없다.

4.4 노출 단계

평균노출 프로토콜은 평가자의 입력 $h_1, \cdots h_k$과 CSP의 입력 sk로 다음과 같이 수행하여 평가자는 복호화된 군집 중심 $Dec(sk, h_1), \cdots, Dec(sk, h_k)$를 출력한다.

1) 평가자는 CSP에 임의의 k개의 $r_j \in R_p$를 선택하여 랜덤 마스크(mask)를 더한 암호문 $h_1 + Enc(pk, r_1), \cdots, h_k + Enc(pk, r_k)$을 전송한다.

2) CSP는 전송받은 암호문을 복호화해서 평가자에게 전송한다.

3) 평가자는 랜덤 마스크 r_j를 제거하여 실제 군집 중심을 출력한다.

V. 분 석

5.1 안전성 분석

5.1.1 최소거리 검색 프로토콜의 안전성

• 사용자 데이터 집합에 대한 프라이버시 : 평가자의 뷰에서 $C_1, \cdots C_k$와 최소의 평문 값을 가지는 암호문의 인덱스 i를 볼 수 있다. RLWE 가정이 유효할 때 IND-CPA에서 안전하다고 증명된 BGV 암호시스템[14]과 Chilotti 등[16]의 동형암호시스템에 의해서 평가자는 주어진 암호문에서 평문에 대한 어떠한 정보도 알 수 없다. 평가자에게 프로토콜을 수행하면서 노출된 정보는 사용자의 평문 데이터에 대한 정보를 포함하지 않는 군집의 인덱스 i뿐이다. 평가자가 CSP에 sum 대신에 임의의 암호문에 대한 복호화 질의에 성공하기 위해서는 전체 평문 공간 R_p에서 2^k 미만의 2의 거듭제곱의 꼴의 평문을 가지는 암호문을 선택해야 한다. 이러한 경우의 수는 $\{2^1, \cdots 2^k\}$의 k개뿐이므로 평가자가 최소거리 검색 프로토콜에서 복호화 질의를 통해 얻는 이점은 $Adv_{Eval}^{SMP} = \frac{k}{|R_p|} = \frac{k}{p^m}$로 정의된다. 위수 p와 다항식 환의 차수 m에 대하여 평가자가 얻는 이점 Adv_{Eval}^{SMP}은 충분히 작다(negligible). 또한, 평가자는 각 군집에 포함된 데이터의 개수를 알 수 있는데 이 정보는 클러스터링 알고리즘에서 일반적으로 각 군집에 포함된 원소의 개수가 비슷하므로 전체 데이터의 개수 n에서 추측할 수 있는 값이다. CSP의 뷰에서 i가 암호문에 속하는 인덱스라고 할 때, $\sum 2^{\pi(i)} d_{\pi(i)}$의 평문 값을 볼 수 있다. 그러나 $\pi(i)$는 평가자가 S_k에서 임의로 선택한 조합 π의 출력이므로 실제 데이터에 대한 정보는 포함하지 않는다.

• 클래스 레이블에 대한 프라이버시 : CSP의 뷰에서 복호화된 $\sum 2^{\pi(i)} d_{\pi(i)}$를 볼 수 있다. $\pi(i)$는 평가자가 S_k에서 임의로 선택한 조합 π의 출력이므

로 CSP는 클래스 레이블에 대한 어떠한 정보도 알 수 없다.

5.1.2 평균계산 프로토콜의 안전성

• 사용자 데이터 집합에 대한 프라이버시 : 평가자의 뷰에서는 C_{sum}, n을 입력으로 C_{avg}를 계산한다. RLWE 가정에 의해 IND-CPA에서 안전성이 증명된 완전동형암호[14]에 의해서 평가된 암호문 C_{avg}의 평문에 대한 어떠한 정보도 노출하지 않는다. 한편, CSP의 뷰에서는 평가자가 선택한 난수 $r_1, r_2 \in R_p$가 섞인 M이하의 $n+1$개의 데이터의 합인 sum'과 클래스에 속하는 데이터의 개수 n'을 볼 수 있다. CSP는 랜덤 마스크를 더한 sum'이 메시지 공간 R_p에서 균일하게 분포되어 있고, 상수 $0 < n' < N$가 난수가 곱해져 있으므로 각 사용자의 데이터에 대한 어떠한 정보도 알아낼 수 없다.

• 클래스 레이블에 대한 프라이버시 : CSP는 각 레이블에 속하는 데이터의 합과 데이터의 개수를 랜덤 마스크가 더해진 형태 외에 각 데이터에 대한 어떠한 레이블 정보도 알 수 없다.

5.1.3 평균노출 프로토콜의 안전성

• 사용자 데이터 집합에 대한 프라이버시 : 평가자의 뷰에서는 $h_1, \cdots h_k$을 입력으로 $Dec(sk, h_1), \cdots, Dec(sk, h_k)$를 계산한다. RLWE 가정에 의해 IND-CPA에서 안전성이 증명된 완전동형암호[14]에 의해서 암호문에서 평문에 대한 어떠한 정보도 노출하지 않는다. 평가자가 n명의 사용자 중 $t = \frac{n}{k} - 2$명 이하의 사용자와 공모한 경우 평가자는 공모하지 않은 사용자들의 평균값 외에 각 평문의 정보는 노출하지 않는다. 한편, CSP의 뷰에서는 k개의 평가된 암호문 $h_j + Enc(pk, r_j)$을 입력받는다. 완전동형암호[14]로 암호화된 암호문 $h_j + Enc(pk, r_j)$는 RLWE 가정에 의해 평문에 대한 어떠한 정보도 노출하지 않는다.

• 클래스 레이블에 대한 프라이버시 : 평균노출 프로토콜에서는 데이터의 합과 데이트의 개수를 랜덤 마스크가 더해진 형태 외에 CSP는 각 데이터에 대한 어떠한 레이블 정보도 알 수 없다.

5.2 비교·분석

본 장에서는 Table 2.와 같이 제안하는 프로토콜의 정수단위 표현과 비트단위 표현에서의 연산량과 통신량을 단계별로 비교 분석한다. 여기서 d는 평문 데이터의 차원의 크기, n은 데이터의 개수, k는 클러스터로 구분할 레이블의 개수, p는 완전동형암호에서 선택한 소수이다. κ_p는 정수단위로 암호화했을 때 암호문 하나의 크기를 가리키고, κ_2는 비트단위로 암호화했을 때의 암호문의 크기를 가리킨다. 단계별 연산의 시간복잡도는 동형암호에서 가장 중요한 연산인 곱셈과 평가연산의 횟수를 기준으로 비교하였다. 정수단위의 비교 연산은 비트단위 연산과 비교해 곱셈횟수는 $O(\log p)$배 적지만 곱셈연산 평가의 연산량이 이진게이트 평가의 연산량보다 많으므로 총 연산시간은 비트단위 연산이 더 효율적이다.

Table 3.은 제안하는 프로토콜과 Bunn 등[5]의 연구, Jaschke 등[8]의 연구와 전체 연산량과 통신량의 시간복잡도 및 지원하는 성질을 비교했다. m은 평가단계와 중심 재계산단계 반복 횟수를 가리키고, m'은 변형된 k-means 클러스터링에서의 평가단계와 중심 재계산단계 반복 횟수를 말한다. UP-kCP는 정수단위로 표현했을 때 Bunn 등[5]의 연구에 비해 연산 시간복잡도가 $O(m'dnk^2)$만큼 작다. 완전동형암호시스템의 암호문과 다자간 계산 프로토콜에서 주고받는 메시지의 크기를 직접 비교할 수는 없다. 따라서 같은 보안상수에서 암호문의 크기

Table 2. Computational and communicational complexity for each phase

Phase	Computation Complexity		Communication Complexity	
	Integer-wise	Bit-wise	Integer-wise	Bit-wise
Preparation	$O(dn)$	$O(dn\log p)$	$O(dn\kappa_p)$	$O(dn\log p\kappa_2)$
Evaluation	$O(dnk)$	$O(dnk\log p)$	$O(n\kappa_p)$	$O(n\log p\kappa_2)$
Recomputing Mean	$O(dk)$	$O(dk\log p)$	$O(d\kappa_p)$	$O(d\log p\kappa_2)$

Table 3. Complexity comparison with previous works

	Computation Complexity	Communication Complexity	Data Privacy	Number of Users
[5]	·	$O(mndk\lambda)+O(mnk\lambda^2)+O((m+d)k\lambda^3\log\lambda)$	X	2
[8]	$O(m'dnk^2)+O(m'dk)$	·	O	1
Ours (Integer-wise)	$O(mdnk)+O(mdk)$	$O(dn\kappa_p)+O(mn\kappa_p)+O(md\kappa_p)$	O	n
Ours (Bit- wise)	$O(mdnk\log p)+O(mdk\log p)$	$O(dn\log p\kappa_2)+O(mn\log p\kappa_2)+O(d\log p\kappa_2)$	O	n

가 $\kappa_2 \ll \kappa_p \ll O(\lambda^2)$을 만족하므로 제안하는 기법과 Jaschke 등[10]의 연구의 통신량의 공간복잡도를 비교하면 $O(dn\kappa_p)+O(mn\kappa_p)+O(md\kappa_p) < O(dn\lambda^2)$이다. 그러므로 UP-kCP는 Jaschke 등[8]의 연구에 비해 $O((m+d)k\lambda^3\log\lambda)$ 만큼 더 효율적이다. Bunn 등[5]의 연구에서 다자간 계산프로토콜로 통신하는 반면 UP-kPC는 암호문을 전송하기 때문에 통신량이 비교적 적다는 것을 알 수 있다. 또한, 2개의 서버에 평문으로 저장된 데이터를 활용하여 데이터 프라이버시를 보장하지 않는 Bunn 등[5]의 기법과 한 명의 사용자에 대한 클러스터링을 수행하는 Jaschke 등[8]의 연구와 비교해 제안하는 UP-kPC는 다수의 사용자에 대한 프라이버시를 보존하는 클러스터링을 수행한다는 장점이 있다.

V. 결 론

본 논문에서는 다수의 사용자가 제공하는 암호화된 데이터에 대한 k-means 클러스터링 기법을 수행하는 환경에서 프라이버시를 보존하는 시스템모델과 프로토콜을 제안하였다. 제안하는 UP-kCP는 암호문에 대한 클러스터링 연산을 수행하면서 완전동형암호시스템의 비효율적인 나눗셈 연산과 비교연산을 지원하지 않는 문제점을 개선하였다. 동형암호서비스를 제공하는 CSP의 도움을 받아 Jaschke 등[8]의 완전 동형암호문에 대한 클러스터링 기법과 비교했을 때 최솟값 계산방법이 효율적이고, 여러 명의 사용자가 참여하는 환경에서도 적합하다. 또한, 프라이버시를 보존하는 클러스터링 기법을 설계하는데 고전적인 문제점이었던 암호문의 크기 비교를 라그랑주 보간다항식과 이진게이트로 해결하였다. 그 결과 최소거리 검색 프로토콜과 암호문 일치함수를 설계하여 동형암호문 데이터 간의 거리를 비교하였다. 또한, 평균계산 프로토콜을 통해서 새롭게 만들어진 군집의 중심을 계산해서 평균 연산을 수행했다.

이후의 연구로는 HElib 라이브러리가 제공하는 암호문 포장 기능을 이용해서 연산시간과 전송량을 줄이는 기법을 설계하는 것과 악의적인(malicious) 평가자를 공격자로 고려하는 것도 의미가 있을 것이다.

References

[1] S. Dolnicar, "Using cluster analysis for market segmentation - typical misconceptions, established methodological weaknesses and some recommendations for improvement," Australasian Journal of Market Research, vol. 11, no. 2, pp. 5-12, Nov. 2003.

[2] M.N. Tuma, R. Decker, and S.W. Scholz, "A survey of the challenges and pitfalls of cluster analysis application in market segmentation," International Journal of Market Research, vol. 53, no. 3, pp. 391-414, May 2011.

[3] M. GJ, "Cluster analysis and related techniques in medical research," Statistical Methods in Medical Research. vol. 1, no. 1, pp. 27-48, Mar. 1992.

[4] R. Paul, and A.S.M.L. Hoque, "Clustering medical data to predict the likelihood of diseases," Pro-

ceedings of 2010 Fifth International Conference on Digital Information Management, pp. 44-49, July 2010.
〔5〕 P. Bunn and R. Ostrovsky, "Secure two-party k-means clustering," Proceedings of the 14th ACM conference on Computer and communications security, pp. 486-497, Oct. 2007.
〔6〕 G. Jagannathanand and R.N. Wright, "Privacy-preserving distributed k-means clustering over arbitrarily partitioned data," Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, pp. 593-599, Aug. 2005.
〔7〕 S. Jha, L. Kruger, and P. McDaniel, "Privacy Preserving Clustering," European Symposium on Research in Computer Security, LNCS 3679, pp. 397-417, 2005.
〔8〕 A. Jaschke, and F. Armknecht, "Unsupervised Machine Learning on Encrypted Data," IACR ePrint, Report 2018-411, May 2018.
〔9〕 V. Nikolaenko, U. Weinsberg, S. Ioannidis, M. Joye, D. Boneh, and N. Taft, "Privacy-Preserving Ridge Regression on Hundreds of Millions of Records," Proceedings of 2013 IEEE Symposium on Security and Privacy, pp. 334-348, May 2013.
〔10〕 A.C. Yao, "How to generate and exchange secrets," Proceedings of 27th Annual Symposium on Foundations of Computer Science, pp. 162-167, Oct. 1986.
〔11〕 Y. Lindell and B. Pinkas, "A Proof of Security of Yao's Protocol for Two-Party Computation," Journal of Cryptology, vol. 22, no. 2, pp. 161-188, Apr. 2009.
〔12〕 Z. Brakerski, C. Gentry, and V. Vaikuntanathan, "(Leveled) fully homomorphic encryption without bootstrapping," ACM Transactions on Computation Theory, vol. 6, no. 13, July 2014.
〔13〕 HElib, "HElib", https://github.com/shaih/HElib, Last accessed 13 Dec. 2018.
〔14〕 I. Chillotti, N. Gama, M. Georgieva, and M. Izabachène, "Faster Fully Homomorphic Encryption: Bootstrapping in Less Than 0.1 Seconds," Advances in Cryptology, ASIACRYPT 2016, LNCS 10031, pp. 3-33, 2016.
〔15〕 C. Gentry, A. Sahai, and B. Waters, "Homomorphic Encryption from Learning with Errors: Conceptually - Simpler, Asymptotically-Faster, Attribute-Based," Advances in Cryptology, CRYPTO '13. LNCS 8042, pp. 75-92, 2013.
〔16〕 TFHE:Fast Fully Homomorphic Encryption over the Torus, "TFHE," https://tfhe.github.io/tfhe/, Last accessed 13 Dec. 2018.
〔17〕 B.K. Samanthula, Y. Elmehdwi, and W. Jiang, "k-Nearest Neighbor Classification over Semantically Secure Encrypted Relational Data," IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 5, pp. 1261-1273, May 2015.
〔18〕 H. Narumanchi, D. Goyal, N. Emmadi, and P. Gauravaram, "Performance Analysis of Sorting of FHE Data: Integer-Wise Comparison vs Bit-Wise Comparison," Proceedings of 2017 IEEE 31st International Conference on Advanced Information Networking and Applications, pp. 902-908, Mar. 2017.
〔19〕 J. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations," Proceedings of Fifth Berkeley Symposium on Mathematical Statistics and

Probability, pp. 281-297, Jan. 1967.
[20] C. Peikert and S.Shiehian, "Multi-Key FHE from LWE, Revisited", IACR ePrint 2016-196, Aug. 2016.
[21] V. Lyubashevsky, C. Peikert, and O. Regev, "On Ideal Lattices and Learning with Errors over Rings", Advances in Cryptology, EUROCRYPT 2010, LNCS 6110, pp. 1-23, 2010.

〈저 자 소 개〉

정 윤 송 (Yunsong Jeong) 학생회원
2017년 2월: 고려대학교 컴퓨터학과 졸업
2017년 3월~현재: 고려대학교 정보보호학과 석사과정
〈관심분야〉 암호프로토콜, 암호이론

김 준 식 (Joon Sik Kim) 학생회원
2016년 2월: 고려대학교 물리학과 졸업
2016년 3월~현재: 고려대학교 정보보호학과 석사과정
〈관심분야〉 암호프로토콜, 암호이론, 인증 및 키 교환

이 동 훈 (Dong Hoon Lee) 종신회원
1983년 8월: 고려대학교 경제학사 졸업
1987년 12월: Oklahoma University 전산학과 석사 졸업
1992년 5월: Oklahoma University 전산학과 박사 졸업
1993년 3월~1997년 2월: 고려대학교 전산학과 조교수
1997년 3월~2001년 2월: 고려대학교 전산학과 부교수
2001년 3월~현재: 고려대학교 정보보호대학원 교수
〈관심분야〉 암호 프로토콜, 암호이론, USN이론, 키 교환, 익명성 연구, PET 기술