

# 위치 기반 서비스를 위한 계층 클러스터 기반 Cloaking 알고리즘

이재흥\*

Hierarchical Clustering-Based Cloaking Algorithm for Location-Based Services

Jae-Heung Lee\*

요 약

최근 스마트 폰 이용자 수가 증가하면서 다양한 위치 기반 서비스들이 주목을 받고 있다. 위치 기반 서비스는 사용자의 위치와 시스템이 가지고 있는 다양한 정보를 결합하여 사용자에게 유용한 정보를 전달해 주기도 하지만 이로 인한 개인 정보의 침해 가능성 역시 높은 것이 사실이다. 최근의 위치 기반 서비스에서의 프라이버시 관련 연구는 K-anonymity를 만족하는 Cloaking 영역 생성에 중점을 두고 있다. 본 논문에서는 위치 기반 서비스를 위한 계층 클러스터 기반 Cloaking 알고리즘을 제안한다. 제안 기법은 약간 변형된 응집 계층 클러스터링 기법을 사용해서 트리를 생성한 뒤, Reciprocity 성질을 만족시키는 Cloaking 영역을 생성한다. 제안 기법은 Reciprocity 성질을 만족시키며, Hilbert Cloak보다 작고 RC-AR과 비슷한 크기의 영역을 생성하며, 생성 속도는 Hilbert Cloak과 비슷하며 RC-AR보다는 훨씬 빠르다.

ABSTRACT

The rapid growth of smart phones has made location-based services (LBSs) widely available. However, the use of LBS can raise privacy issues, as LBS can allow adversaries to violate the location privacy of users. There has been a considerable amount of research on preserving user location privacy. Most of these studies try to preserve location privacy by achieving what is known as location K-anonymity. In this paper, we propose a hierarchical clustering-based spatial cloaking algorithm for LBSs. The proposed algorithm constructs a tree using a modified version of agglomerative hierarchical clustering. The experimental results show, in terms of the ASR size, that the proposed algorithm is better than Hilbert Cloak and comparable to RC-AR (R-tree Cloak implementation of Reciprocal with an Asymmetric R-tree split). In terms of the ASR generation time, the proposed algorithm is much better in its performance than RC-AR and similar in performance to Hilbert Cloak.

키워드

Privacy, Anonymity, Location-Based Services, Spatial Cloaking, Hierarchical Clustering  
프라이버시, 익명성, 위치 기반 서비스, 공간 클로킹, 계층 클러스터링

## 1. 서 론

최근 Apple의 iPhone을 포함한 다양한 스마트 폰들이 나오면서 위치 기반 서비스(Location-Based

Service : LBS)들이 주목을 받고 있다[1, 2, 3]. 이러한 위치 기반 서비스는 사용자의 위치와 시스템이 가지고 있는 다양한 정보를 결합하여 사용자에게 유용한 정보를 전달한다. 예를 들어 사용자는 “현재 위치

\* 교신저자(corresponding author) : 서울대학교 컴퓨터연구소(jhlee@os.snu.ac.kr).  
접수일자 : 2013. 06. 20

심사(수정)일자 : 2013. 07. 22

게재 확정일자 : 2013. 08. 23

에서 가장 가까운 커피 전문점은 어디지?”라든지, “현재 내가 달리고 있는 도로의 상태는 어떻게?” 같은 질의를 LBS 서버에게 요청할 수 있다. 그러면 LBS 서버는 질의에 포함된 사용자의 위치와 자신이 가지고 있는 정보를 활용해 알맞은 응답을 생성해 사용자에게 되돌려 준다.

올바른 응답을 받기 위해 사용자는 자신의 정확한 위치를 LBS 서버에게 전달해야 한다. 하지만 이렇게 서버에게 전달되는 위치 정보를 통해 심각한 개인 정보 침해가 이루어질 수 있다. 만약 LBS 서버에 보내지는 사용자들의 위치 정보가 해커나 서버 관리자 또는 국가에 의해 악용될 경우, 개인의 생활 습관, 질병 정보, 정치나 종교적 신념 같은 사적 영역에 대한 침해가 이루어질 수 있다. 따라서 사용자가 편하고도 안전하게 사용할 수 있는 개인 정보 보호 방법이 요구된다.

개인 정보 보호를 위해 생각할 수 있는 가장 간단한 방법은 사용자가 LBS 서버에게 직접 질의를 보내지 않도록 하는 것이다. 대신 사용자는 질의를 신뢰할 수 있는 필명 서비스(pseudonym service)를 거치도록 함으로써 자신의 신원을 숨긴다. 이렇게 하면 사용자의 신원 관련 정보들이 드러나지 않으므로 질의를 보낸 사용자가 누구인지 알 수 없을 것 같지만 사실은 그렇지 않다. 질의에 포함된 사용자의 위치 정보와 이미 공개되어 있는 다양한 외부 정보들을 결합함으로써 그 질의를 보낸 사용자가 누구인지 추론할 수 있다. 예를 들어 질의에 포함된 위치가 혼자 살고 있는 누군가의 집 내부라면 그 질의를 보낸 사람은 그 집 주인일 확률이 높다. 그러므로 질의를 보낸 사람이 자신의 사용자 ID나 IP 주소를 드러내지 않더라도 위치 정보만으로 질의를 보낸 사람을 예측하는 것이 가능하다.

최근의 위치 기반 서비스에서의 프라이버시 관련 연구는 K-anonymity[4]를 만족하는 Cloaking 영역 생성에 중점을 두고 있다. K-anonymity는 원래 관계형 데이터베이스에서 나온 개념으로 테이블에 저장된 각각의 데이터 레코드가 quasi-identifier라 불리는 배포된 외부 정보와 연결할 수 있는 속성들의 집합에 대해 서로 구별이 불가능한 데이터 레코드를 적어도 K-1개 가지도록 하는 것을 의미한다.

위치 기반 서비스 프라이버시와 관련해서

K-anonymity를 보장하기 위한 다양한 기법들이 제안되었다[5, 6, 7]. 이 기법들은 사용자의 정확한 위치를 주변의 K명 이상의 사용자를 포함하는 Cloaking 영역으로 변환하여 LBS 서버에게 전달한다. 하지만 K명 이상의 사용자를 포함하는 Cloaking 영역이 K-anonymity를 무조건 보장해주는 것은 아니다.

Reciprocity 특성은 이러한 단점을 보완하기 위해 도입되었다[6]. Reciprocity 특성을 만족시킬 경우 K-anonymity를 만족시킨다는 것이 이미 증명되었다. Hilbert Cloak[6]와 Reciprocal[7]이 Reciprocity 특성을 만족시키도록 제안되었다. Hilbert Cloak은 2차원인 전체 공간 영역을 Hilbert 커브[8]를 통해 1차원 공간 영역으로 변환한 뒤, 이 1차원 공간 영역에서의 각 노드의 순서를 계산하고, 이렇게 계산된 노드들의 순서를 통해 전체 노드들을 Reciprocity 성질을 가지는 노드 집합들의 모임으로 분할한다. Reciprocal은 R\*-tree나 Quad-tree와 같은 계층 공간 색인을 이용해서 전체 공간을 Reciprocity 성질을 가지는 공간들로 나눈다. Hilbert Cloak은 Reciprocal의 한 구현인 RC-AR (R-tree Cloak implementation of Reciprocal with Asymmetric R-tree split)보다 더 빠르게 Cloaking 영역을 생성하지만, 생성되는 영역의 크기가 크다.

본 논문에서는 위치 기반 서비스를 위한 계층 클러스터 기반 Cloaking 알고리즘을 제안한다. 제안 기법은 약간 변형된 응집 계층 클러스터링 기법을 사용해서 트리를 생성한 뒤, Reciprocity 성질을 만족시키는 Cloaking 영역을 생성한다. 제안 기법은 Reciprocity 성질을 만족시키며, Hilbert Cloak보다 작고 RC-AR과 비슷한 크기의 영역을 생성하며, 생성 속도는 Hilbert Cloak과 비슷하며 RC-AR보다는 훨씬 빠르다.

본 논문의 구성은 다음과 같다. 2장에서는 응집 계층 클러스터링 기법을 설명하고, 3장에서 제안 기법을 소개한다. 기존 연구들과의 비교 성능 평가 결과는 4장에서 제시되며, 5장에서 결론을 맺는다.

## II. 응집 계층 클러스터링

클러스터링은 데이터를 같은 그룹에 있는 데이터가

다른 그룹에 있는 데이터보다 더 밀접하게 관련되도록 클러스터라 불리는 그룹들로 나누는 과정을 말한다.

응집 계층 클러스터링에서는 데이터들이 연속적으로 결합됨으로써 클러스터들이 생성된다. 클러스터링 될 N개의 데이터가 존재할 때, 기본적인 응집 계층 클러스터링의 과정은 다음과 같다[9].

- ① 각각의 데이터를 하나의 클러스터에 대응시킨다. 그러므로 처음에는 N개의 클러스터가 존재하게 된다. 그리고 각각의 클러스터 사이의 거리를 측정한다.
- ② 그 중 가장 가까운 두 개의 클러스터를 찾아 합병한다.
- ③ 합병되어 새롭게 생성된 클러스터와 다른 클러스터 사이의 거리를 측정한다.
- ④ 크기 N인 하나의 클러스터만 남을 때까지 두 번째와 세 번째 과정을 계속 반복한다.

### III. 제안 기법

이번 장에서는 위치 기반 서비스를 위한 계층 클러스터 기반 Cloaking 알고리즘을 기술한다. 제안 기법은 약간 변형된 응집 계층 클러스터링 기법을 사용해서 트리를 생성한 뒤, Reciprocity 성질을 만족시키는 Cloaking 영역을 생성한다.

3.1 절에서는 트리 생성 알고리즘을 기술하고, 3.2 절에서는 이를 바탕으로 한 Cloaking 영역 생성 알고리즘을 기술한다.

#### 3.1 트리 생성 알고리즘

표 1은 제안 기법의 트리 생성 알고리즘으로써 응집 계층 클러스터링을 기반으로 한다. 이 알고리즘은 각각 한 명의 사용자만 포함하는 N개의 서로 다른 트리로 시작해서 (라인: 1-2), 가장 가까운 두 개의 트리를 찾아 합병하는 과정을 반복함으로써 N명의 모든 사용자를 포함하는 하나의 트리를 생성한다 (라인: 5-11). 두 트리 사이의 거리는 각 트리에 있는 사용자들 간의 거리를 평균해서 계산한다.

여기서 가장 많은 시간을 소비하는 부분이 가장 가

까운 두 트리를 찾는 부분인데, 이는 dynamic closest pair 문제와 관련된 다른 연구 결과들을 이용해 O(N) 시간 안에 가능하다[10]. 표 1에서는 각 트리가 자신과 가장 가까운 트리의 정보를 유지하면서 (라인: 3-4), 트리 관련 정보들이 바뀔 때마다 관련 정보들을 업데이트 함으로써 (라인 10) 효율적으로 가장 가까운 두 트리를 찾도록 하였다.

표 1에서 사용한 MergeTrees 함수는 두 트리를 합병하면서, 서로 가까운 사용자들이 트리 안에서도 가까이 위치하도록 필요에 따라 트리의 위치를 바꾸거나 뒤집는 일을 한다.

표 1. 트리 생성 알고리즘  
Table 1. Tree generation algorithm

Input: $U_1, U_2, U_3, \dots, U_N$ (N Users)
01: for (i = 1; i < N; i++)
02: $T_i = \text{CreateTreeNode}(U_i)$
03: for (i = 1; i < N; i++)
04: $\text{FindNearestNeighbor}(T_i)$
05: while (N != 1) {
06:   (i, j) = $\text{FindClosestPair}(T_1, T_2, \dots, T_N)$
07: $T_i = \text{MergeTrees}(T_i, T_j)$
08: $T_j = T_N$
09: $N = N - 1$
10: $\text{UpdateNearestNeighbor}(T_1, T_2, \dots, T_N)$
11: }

#### 3.2 Cloaking 영역 생성 알고리즘

Cloaking 영역 생성 알고리즘은 표 2와 같다. 표 2는 앞에서 만들어진 트리를 root 노드부터 아래로 탐색하면서 질의자를 포함한 K명 이상의 사용자를 포함하는 가장 작은 Reciprocal 집합을 찾는다. 여기서 L과 R은 이전에 방문한 노드에서 아직 Reciprocal 집합을 만드는데 사용되지 않은 사용자들을 보관하기 위해 사용된 집합이다. 생성되는 Cloaking 영역은 해당 집합의 사용자들을 포함하는 가장 작은 직사각형 영역이다.

표 2. Cloaking 영역 생성 알고리즘  
Table 2. Cloaking area generation algorithm

```

Input: T (Root Node), U (Querying User), K
01: L =  $\emptyset$ , R =  $\emptyset$ 
02: UCL = false, UCR = false
03: while (1) {
04: LC  $\leftarrow$  GetUserSet(T.left_child)
05: RC  $\leftarrow$  GetUserSet(T.right_child)
06: if  $|LC| \geq K$  and  $|RC| \geq K$  then {
07:   if  $U \in LC$  or UCL = TRUE then
08:     R =  $\emptyset$ , T = T.left_child
09:   else
10:     L =  $\emptyset$ , T = T.right_child
11: }
12: else if  $|LC| \geq K$  and  $|RC| < K$  then {
13:   if  $|L| + |RC| \geq K$  then
14:     if  $U \in LC$  or UCL = TRUE then
15:       R =  $\emptyset$ , T = T.left_child
16:     else
17:       return RURC
18:   else
19:     if  $U \in RC$  then
20:       UCR = true;
21:       R = RURC, T = T.left_child
22: }
23: else if  $|LC| < K$  and  $|RC| \geq K$  then {
24:   if  $|L| + |LC| \geq K$  then
25:     if  $U \in RC$  or UCR = TRUE then
26:       L =  $\emptyset$ , T = T.right_child
27:   else
28:     return LULC
29:   else
30:     if  $U \in LC$  then
31:       UCL = true;
32:       L = LULC, T = T.right_child
33: }
34: else {
35:   else if  $|LC| < K$  and  $|RC| < K$  then
36:     if  $U \in LC$  or UCL = TRUE then
37:       return LULC
38:     else
39:       return RURC
40:   else
41:     return LULCURURC
42: }
43: }

```

#### IV. 성능 평가

이 장에서는 제안 기법과 Hilbert Cloak 기법, 그리고 Reciprocal의 한 구현인 RC-AR의 성능을 다양한 관점에서 실험을 통해 비교하였다. 모든 실험은 Intel Core 2 Duo 2.53GHz의 프로세서와 2GB의 메모리를 가진 Windows Vista 시스템에서 수행되었다.

Reciprocal은 R\*-tree(RC)나 Quad-tree(QC)와 같은 다양한 공간 인덱스 기법들을 사용해서 구현할 수 있다. 우리는 제안 기법을 오직 RC와만 비교했는데 이는 RC가 QC나 다른 공간 인덱스 기법들보다 빠르고 생성되는 Cloaking 영역의 크기도 비슷하기 때문이다[7]. 제안 기법을 RC-GH가 아닌 RC-AR과만 비교한 것도 마찬가지 이유이다.

성능 평가를 위해 데이터 집합에 존재하는 모든 사용자가 각각 질의를 요청하도록 하고 그 결과들을 평균하여 비교를 수행하였다. 이때 사용된 데이터 집합으로 <http://www.rtreeportal.org>에 있는 US 데이터 집합을 사용하였다. 이 데이터 집합은 미국 안에 있는 15,206 곳의 유명한 장소를 포함한 자료이다.

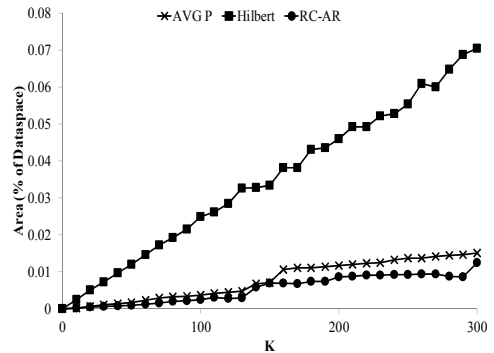


그림 1. Cloaking 영역 크기  
Fig. 1 Cloaking area size

그림 1은 K 값 변화에 따른 Cloaking 영역의 크기를 비교한 것이다. Cloaking 영역의 크기 관점에서 제안 기법은 Hilbert Cloak보다 더 좋은 성능을 보였으며, RC-AR과는 거의 비슷한 성능을 나타냄을 알 수 있다.

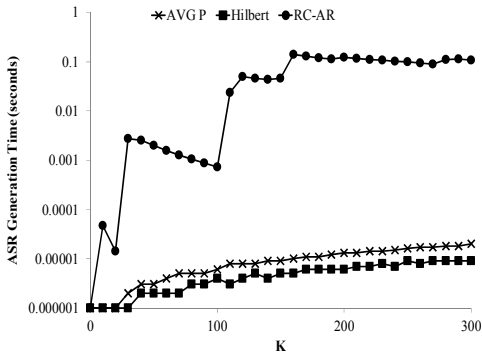


그림 2. Cloaking 영역 생성 시간  
Fig. 2 Cloaking area creation time

그림 2는 K 값 변화에 따른 Cloaking 영역 생성 시간을 비교한 것이다. Y축은 로그 단위로 나타내었으며, Cloaking 영역 생성 시간의 관점에서 제안 기법은 Hilbert Cloak과 비슷한 성능을 보였으며, RC-AR 보다는 훨씬 나은 성능을 보여주었다.

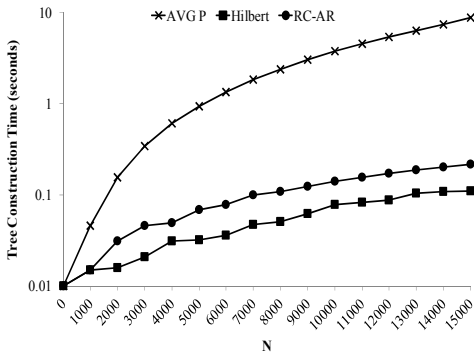


그림 3. 트리 생성 시간  
Fig. 3 Tree creation time

그림 3은 사용자 수 (N 값) 변화에 따른 트리 생성 시간을 비교한 것이다. 역시 Y축은 로그 단위로 나타내었으며, 트리 생성 시간의 관점에서 제안 기법은 기존의 두 기법들보다 좋지 않은 성능을 보이고 있다. 이는 제안 기법이 트리를 생성하는데  $O(N^2)$ 의 시간을 필요로 하는데 비해 다른 두 기법들은  $O(N \log N)$ 의 시간만 필요로 하기 때문이다. 하지만 제안 기법에서 트리 생성은 오직 처음 한 번, 그리고 후에 아주 가끔만 일어나고, 이러한 과정은 백그라운드 프로세스로도 수행 가능하므로, 트리 생성으로 인한

부하는 그다지 많지 않다.

## V. 결론

본 논문에서는 위치 기반 서비스를 위한 계층 클러스터 기반 Cloaking 알고리즘을 제안하였다. 제안 기법은 약간 변형된 응집 계층 클러스터링 기법을 사용해서 트리를 생성한 뒤, Reciprocity 성질을 만족시키는 Cloaking 영역을 생성한다. 제안 기법은 Reciprocity 성질을 만족시키며, Hilbert Cloak보다 작고 RC-AR과 비슷한 크기의 영역을 생성하며, 생성 속도는 Hilbert Cloak과 비슷하며 RC-AR보다는 훨씬 빠르다.

## 참고 문헌

- [1] Moon PJ, "On the Availability of Anti-Forensic Tools for Android Smartphones", The Journal of The Korea Institute of Electronic Communication Sciences, Vol. 8, No. 6, pp. 855 - 862, 2013.
- [2] Lee HJ, Lee JJ, Jung ST, "A Design and Implementation of Digital Textbook for Elementary School Computer Education based on Smart phone", The Journal of The Korea Institute of Electronic Communication Sciences, Vol. 8, No. 5, pp. 687 - 693, 2013.
- [3] Jung SR, Shin SH, "The effect of the user experience of smart phone on satisfaction", The Journal of The Korea Institute of Electronic Communication Sciences, Vol. 7, No. 5, pp. 1087 - 1093, 2012.
- [4] Sweeney L., "k-anonymity : a model for protecting privacy", International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, Vol. 10, No. 5, pp. 557 - 570, 2002.
- [5] Gruteser M. and Grunwald D., "Anonymous usage of location-based services through spatial and temporal cloaking", Proceedings of the 1st international conference on Mobile systems, applications and services, MobiSys '03, New York, NY, USA, pp. 31 - 42, ACM, 2003.
- [6] Kalnis P., Ghinita G., Mouratidis K., and

- Papadias D., "Preventing location-based identity inference in anonymous spatial queries", IEEE Transactions on Knowledge and Data Engineering, Vol. 19, No. 12, pp. 1719 - 1733, 2007.
- [7] Ghinita G., Zhao K., Papadias D., and Kalnis P., "A reciprocal framework for spatial k-anonymity", Information Systems, Vol. 35, No. 3, pp. 299 - 314, 2010.
- [8] Butz A.R., "Alternative algorithm for hilbert's space-filling curve", IEEE Transactions on Computers, Vol. 20, No. 4, pp. 424 - 426, 1971.
- [9] Johnson S., "Hierarchical clustering schemes", Psychometrika, Vol. 32, No. 3, pp. 241 - 254, 1967.
- [10] Eppstein D., "Fast hierarchical clustering and other applications of dynamic closest pairs", Journal of Experimental Algorithmics, Vol. 5, No. 1, pp. 1 - 23, 2000.

## 저자 소개



### **이재흥(Jae-Heung Lee)**

2001년 서울대학교 컴퓨터공학과  
졸업(공학사)

2003년 서울대학교 대학원 전기,  
컴퓨터공학과 졸업(공학석사)

2013년 서울대학교 대학원 전기, 컴퓨터공학과 졸업  
(공학박사)

2013년~현재 서울대학교 컴퓨터연구소 연구원

※ 관심분야 : 컴퓨터보안, 센서 네트워크