

엣지 컴퓨팅에서 트래픽 분산을 위한 흐름 예측 기반 동적 클러스터링 기법

이 창 우^{*}

Flow Prediction-Based Dynamic Clustering Method for Traffic Distribution in Edge Computing

Chang Woo Lee[†]

ABSTRACT

This paper is a method for efficient traffic prediction in mobile edge computing, where many studies have recently been conducted. For distributed processing in mobile edge computing, tasks offloading from each mobile edge must be processed within the limited computing power of the edge. As a result, in the mobile nodes, it is necessary to efficiently select the surrounding edge server in consideration of performance dynamically. This paper aims to suggest the efficient clustering method by selecting edge servers in a cloud environment and predicting mobile traffic. Then, our dynamic clustering method is to reduce offloading overload to the edge server when offloading required by mobile terminals affects the performance of the edge server compared with the existing offloading schemes.

Key words: Mobile Edge Computing, Traffic Flow, Clustering, Machine Learning, AI

1. 서 론

최근에 여러 산업 분야에서 인공지능(AI), IoT(Internet of Thing), 자율 주행, MR, XR, 메타버스 등 IT 관련 서비스의 확대와 다양한 산업 분야에 적용되면서 유무선 네트워크를 통해 전달되는 서비스와 관련된 데이터의 양도 매우 증가하고 있다. 무엇보다, 온프레미스 형태인 로컬로 구축하고 있던 서버 및 관련된 서비스들이 온라인 서비스의 증가와 함께, 클라우드 기반의 서비스가 우리 일상에서 편리하고 다양한 서비스로 제공이 되고 있다. 이러한 환경은 중앙 서버에서 클라우드가 처리할 데이터 및 트래픽이 예전보다 매우 증가하고 있는 상황이다. 이렇게 증가된 데이터 트래픽과 처리는 서비스에 따라서 실시간으로 제공할 경우에 중앙 서버에서 제공하는 클라우

드 기반 서비스는 처리해야 할 로드를 분산하는 분산 처리 기법이 매우 중요하다[1,2].

중앙 서버에서 데이터를 처리하는 클라우드의 제한적인 처리 용량과 처리 속도를 개선하기 위한 방안으로 엣지 컴퓨팅관련 연구가 활발하게 진행 중이다. 이러한 클라우드의 분산 기법은 무선 이동통신 환경에서 기지국(Base Station)에 적용하여 각 단말에서 오프로딩한 태스크를 처리해주는 모바일 엣지 컴퓨팅관련 된 기술은 MEC(Mobile Edge Computing)라고 한다. 모바일 엣지 컴퓨팅(MEC)은 사용자와 가장 가까이에서 엣지 서버가 있는 환경으로 발생하는 데이터를 처리하는 분산 환경에 맞는 기술로 볼 수 있다. 모바일 엣지 컴퓨팅은 전송할 데이터의 전송 시간 단축과 사용자의 레벨에 맞는 요구 사항과 서비스를 제공한다. 최근 5G 네트워크와 이동 통신

* Corresponding Author : Chang Woo Lee, Address: (02 707) 77, Jeongneung-ro, Seongbuk-gu, Seoul, Republic of Korea, TEL : +82-2-910-6339, FAX : +82-2-910-4868,

E-mail : leecw05@kookmin.ac.kr

Receipt date : Aug. 14, 2022, Approval date : Aug. 24, 2022

[†] The School of Software, Kookmin University

에서는 중요 기술로 연구가 진행되고 있다[3]. Fig. 1은 기존 논문에는 모바일 엣지 컴퓨팅에서 사용자 및 서비스의 트래픽이 많아지면 엣지 클라우드의 태스크(Work Load)를 분산하기 위해서 주변에 있는 모바일 엣지 서버를 검색하여 선택하게 되고 이런 서버가 여러 개로 구성이 되는 경우에는 계층적 엣지 클라우드를 구성하여 오버로드를 분산시키는 방법이 제안되었다[4,5]. 무선 이동통신 환경에서는 사용자가 요구하는 태스크를 분산하는 과정에서 최적화를 위해 담금질(Simulated Annealing) 기법을 사용하여 최적화 값(Optimization Value)을 구하는 방안을 제시한다[5].

다만, 기존의 논문은 모바일 엣지 컴퓨팅 환경에서 태스크를 분산시에 모바일 단말의 사용 패턴 및 모바일 엣지 서버의 처리 성능과 데이터 처리량에 따른 클러스터링으로 분류하는 방안의 고려는 없다. 이러한 기존 논문의 고려하지 않은 부분을 엣지 컴퓨팅에서 본 논문은 모바일 엣지 컴퓨팅에서 태스크(Peak Load)를 분산하는 방안으로 개선된 동적 클러스터링 방식을 제시하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서 본 논문의 모바일 엣지 컴퓨팅의 클러스터링과 트래픽 플로우 예측의 기본 이론이 되는 알고리즘을 정의하고 3장에서 제안하는 트래픽 플로우 예측에서 사용할 클러스터링 기법을 설명한다. 여기에서는 제안한 클러스터링 기법의 전체적인 개요와 구성과 적용되는 절차를 기술한다. 그리고 4장에서는 본 논문에서 제안하는 기법으로 실험한 결과를 객관적인 결과 값으로 비교하여 성능을 평가하고 5장에서 결론을 맺는다.

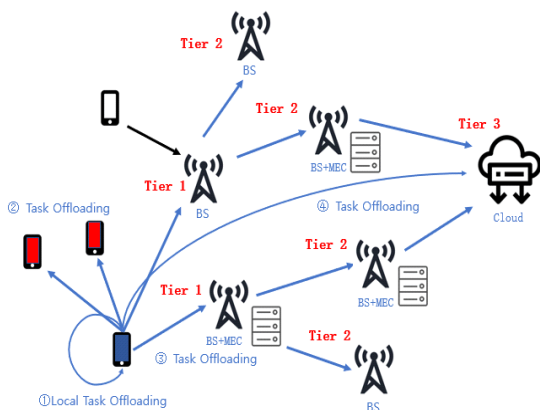


Fig. 1. The flow of the hierarchical clustering method.

2. 모바일 엣지 컴퓨팅

2.1 클러스터링 방식

모바일 엣지 컴퓨팅의 응용 서버로서 중요한 역할은 컴퓨팅 자원의 저장과 데이터 통신 능력, 사용자의 트래픽, 무선 네트워크 액세스 등을 제공하는 것이다. Fig. 2는 무선 이동통신 구성으로 RAN(Radio Access Network)의 내부에서 제공하는 다양한 응용 서비스의 효율적인 통합을 만든다. 모바일 엣지 컴퓨팅에서 엣지 서버는 매크로 기지국(Macro Base Station)이나 RNC(Radio Network Controller)에 있다. 모바일 단말은 오프로드(Self Offload)할 태스크를 해당 모바일 단말에서 바로 처리하거나 주변에 엣지 서버를 선택하여 태스크를 오프로드를 진행한다. 엣지 서버로 오프로드하는 경우는 전체 태스크 중 일부 부분적으로 오프로드(Partial Offload) 혹은 전체 태스크를 오프로드(Full Offload) 모드 중에 선택한 엣지 서버의 상태에 따라 선택하여 진행한다.

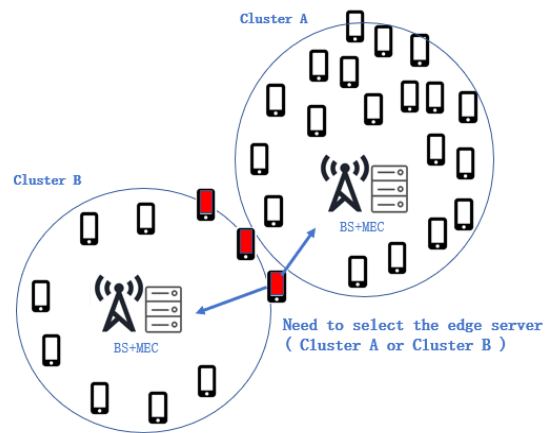


Fig. 2. The edge server selection for the clustering.

2.2 분산 처리 방식

여러 종류의 무선 기술과 이동 통신 기지국을 제어하는 모바일 엣지 컴퓨팅의 구조는 멀티 밴더 환경에서 기능 요소들을 포함한다. 사용자의 다양한 기능들은 트래픽이 증가되고 태스크가 몰려 있는 곳에 인접한 엣지 서버는 동적인 클러스터링으로 집중된 부하를 분산할 수 있다[6,7]. 한 개의 엣지 서버에 단말의 태스크가 집중되면 오버로드가 발생하게 되고 해당 엣지 서버에서 실시간으로 데이터 처리가 되지

않고 데이터 처리에 지연이 발생하게 된다. 엣지 서버의 큐에 저장되는 TASK 정보와 큐에 각 엣지 서버별로 큐에 저장되는 패턴에 따른 엣지 서버의 재선택은 엣지 서버에서 처리해야 할 TASK를 여러 개의 다른 엣지 서버로 분산시킬 수 있다.

3. 제안한 방법

3.1 동적 클러스터링 기법의 개요

동적 클러스터링 방법에서는 엣지 서버의 개수를 결정하고 각 클러스터링의 중심에서 각 에지 서버의 거리를 우선 계산하여 가장 가까운 에지 서버와 데이터 사용 패턴과 데이터 처리량을 고려하여 클러스터링으로 구성한다. 모바일 단말에서 보낸 TASK의 처리는 Work Load 단위마다 각 클러스터링에 대한 새로운 중심값을 계산하고 새로운 중심값에서 다시 동적 클러스터링을 진행한다. 클러스터링은 K-means Clustering 알고리즘을 기반으로 성능 비교를 진행하고 최적화 값(Optimization Value)을 구하는 과정을 반복하여 진행한다[8,9].

특히, Fig. 3의 경우로 특정 엣지 서버로 TASK가 집중된 경우에는 이로 인한 서비스 지연이 발생하므로 개선된 분산 클라우드 기술 적용이 중요하고 서비스 제공에 적용이 필요하다. 기본적인 k-means 알고리즘은 엣지 서버의 갯수인 k 클러스터와 중심 벡터인 μ 값을 시작으로 반복적으로 수행하며 모든 입력 데이터 값은 x 가장 가까운 중심 벡터 값에 설정한다 [8,9].

$$\mu = [\mu_{k0}, \mu_{k1}] (k = 0, 1, 2) \quad (1)$$

여러 개의 중앙 벡터는 입력 데이터에서 같은 거리

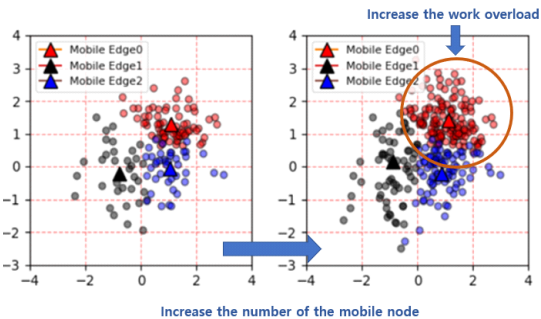


Fig. 3. The distribution of mobile nodes around edge servers.

에 있다면, 임의의 중앙 벡터 값이 선택된다. 만약 y 값이 1이라면, 해당 값은 중앙 벡터 값으로 설정한다.

$$\gamma_k = [\gamma_{k0}, \gamma_{k1}, \gamma_{k2}] \quad (2)$$

개별 데이터와 각 중심 벡터 값과의 거리를 계산하고 최소 거리를 갖는 중심 벡터 값을 클러스터링으로 정한다. 이때 거리는 유클리디안(Euclidean Distance)을 활용한다.

$$\|x_n - \mu_k\| = x_{n0} - \mu_{k0}^2 + x_{n1} - \mu_{k1}^2 (k=0,1,2) \quad (3)$$

이때, 제안하는 동적인 클러스터링 방식은 유클리디안 방식으로 최소 거리를 구한 다음에 바로 엣지 서버로 선택하지 않고 엣지 서버의 성능과 단말로부터 받은 트래픽 처리량을 고려하여 엣지를 선택하도록 개선하였다. 또한, 이 과정에서 단말로부터 받은 데이터의 패턴과 오프로드한 처리량을 분석하여 트래픽 플로우를 예측하는 방식을 적용하여 기존 방식보다 분산 처리 방식에서 효과적으로 진행되도록 고려하였다. 사용자와 엣지 서버의 데이터 사용 패턴을 가우시안 분포로 분석해보면 인접한 엣지 서버를 클러스터링으로 분류할 수 있다. 단말에서 발생하는 TASK에 따른 Work Load를 확인하고 단말이 이동하는 경우는 클러스터링의 경계 구역의 데이터 처리도 고려를 해야한다. 결국, 단말에서 발생하는 TASK를 처리하는 가운데 단말이 다음 클러스터링으로 넘어갈 경우는 모바일 단말의 핸드 오버(Hand Over) 관련 부분도 고려하여 알고리즘에 반영한다.

4. 실험 결과 및 고찰

4.1 동적 클러스터링 기법의 환경 구성

실험 환경은 각 단말과 엣지 서버의 데이터 통신 지연은 일정하다고 가정하고 단말과 엣지 서버간 데이터 통신 지연은 엣지 서버와 중앙 클라우드 사이의 지연보다는 작다고 가정한다. 엣지 서버의 성능은 지연 시간, 데이터 처리 완료 시간으로 분류하여 처리한다. Table 1은 제안하는 동적 클러스터링 기법의 실험을 위한 파라미터를 정리하였다. 엣지 서버의 3개, 모바일 단말은 100개~300개 사이로 랜덤하게 설정하였다. 또한, 각 엣지 서버의 성능은 동일하지 않고 처리 속도가 다르게 설정하였다. 모바일 단말에서 보내는 TASK의 데이터 사이즈는 100MB 로 설정하고 엣지 서버에서 모바일 단말로부터 받은 TASK는

Table 1. Simulation Parameter Definition.

Notation	Definition
n	Number of Edge : $n = 3$
C_{1-n}	Edge cloud with computation capacity : Total 50GHz, each Edge cloud 10GHz
E_j	Edge that task i is placed on
$P_{E_j}^{F_i}$	Percentage of Edge e_j 's computational Capacity being allocated for task i : $0 \sim 100\%$
TS_j	Set of computing tasks placed at Edge $j \rightarrow \{T_1, T_2 \dots T_k\}$
$W_i / P_{E_j}^{F_i} C_{e_j}$	Computation delay of task i 's execution
DS_i	Task i 's data size : 100 MB
B_{e_j}	Network bandwidth allocated to e_j : 10 MB/s
$D_{tr} = DS_i / B_{e_j}$	Communication delay of transmitting program i 's state to edge E_j
EC_j	Energy Consumption for the MEC is given as 1W/GHz
BTD	Backhaul time delay 0.0001 sec/KB
Distance (E -> C)	Distance (edge to core) : 10km

엣지 서버의 큐에 저장하고 순차적으로 처리한다.

4.2 동적 클러스터링 기법의 성능

제안하는 동적 클러스터링 기법은 분산 처리 환경에서 매우 성능이 높다고 볼 수 있다. Fig. 4(a)는 엣지 서버별로 오버로드 분포를 보여주는 것으로 엣지 서버 0번이 모바일 단말로부터 태스크를 가장 많이 받아서 오버로드가 높다. 이 경우에는 엣지 서버 2번은 오프로드 받은 태스크가 많지 않으므로 데이터 처리가 가능하나 유희한 상태로 유지되므로 자원 활용도가 낮다고 볼 수 있다. Fig. 4(b)에서 첫 번째 바차트는 특정 엣지 서버로 태스크가 몰렸을때에 클러스터링 기법을 적용하지 않은 경우로 분산 처리가

되지 않는 상태를 확인할 수 있다. Fig. 4(b)에 두 번째 바차트는 클러스터링이 적용된 경우로 첫 번째 바차트 보다는 클러스터링 기법으로 분산 처리를 진행하여 엣지 서버 0에 몰렸던 태스크를 분산 처리하였으나 시간이 지나면서 엣지 서버 2번으로 다시 태스크 할당이 몰리는 현상을 확인할 수 있다. 결국 모바일 단말의 데이터 발생 패턴과 트래픽 플로우에 따른 엣지 서버 선택을 재선택함으로써 분산 처리를 개선하게 된다. Fig. 4(b)에서 클러스터링을 하지 않은 방식과 엣지의 상태를 고려하지 않은 클러스터링 상태와 비교해서 본 논문에서 제시하는 동적인 클러스터링 방식의 성능이 우수함을 보여주고 있다.

결국, 동적 클러스터링 기법은 특정 엣지 서버로

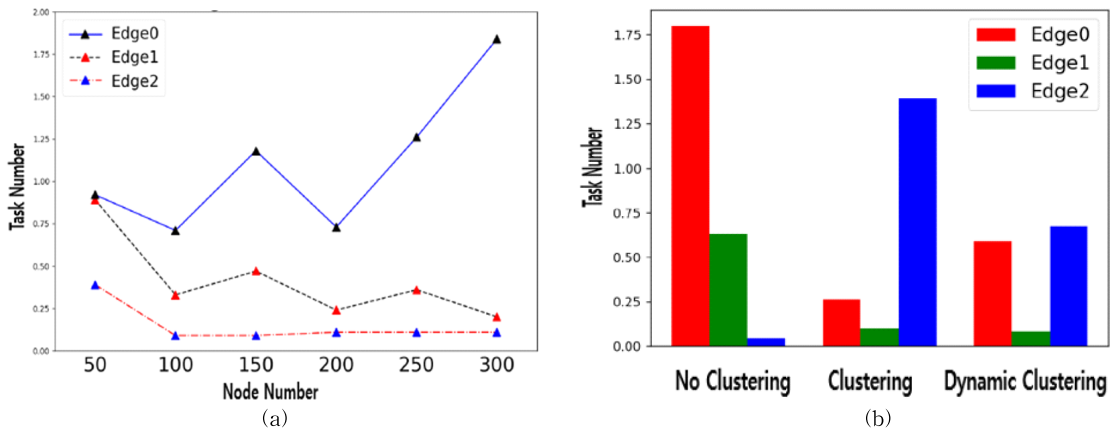


Fig. 4. The performance of the clustering methods, (a) Each task performance and (b) each method performance.

오프로딩의 태스크가 몰려서 엣지 서버의 성능과 모바일 단말의 태스크를 처리하는데 지연되는 문제점을 엣지 서버에 부하가 증가할 때 해당 엣지 서버의 성능을 체크하여 모바일 단말이 엣지 서버를 선택하는 과정을 다시 수행함으로써 기존 방식보다는 부하를 분산할 수 있고 이러한 클러스터링을 활용하면 각 클러스터링별 트래픽 프로우를 각 서버의 큐에 있는 태스크 숫자와 패턴으로 분석할 수 있는 장점이 있다. 트래픽 플로우 분석을 통해서 계속 증가되는 트래픽을 예측하여 몰려 있는 클러스터링에 있는 모바일 단말은 주변에 엣지 서버를 재탐색하여 엣지 서버를 다시 선택하도록 하여 로드 분산이 진행되어 엣지 서버의 성능을 개선할 수 있다.

5. 결 론

본 논문에서는 모바일 엣지 컴퓨팅에서 모바일 사용자의 증가에 따른 데이터 트래픽이 높아지는 무선 네트워크의 환경에서 태스크가 집중되는 Peak Load를 분산시키는 방법과 트래픽 플로우를 예측하는 기법으로 동적 클러스터링 방안을 제안하였다. 모바일 단말 사용자의 트래픽의 사용 패턴과 엣지 서버의 데이터 처리 성능을 고려한 부분은 기존 오프로딩 방식에서는 고려하지 않았던 내용으로 모바일 엣지 서버의 성능과 부하 분산에서 개선하였다. 결국, 동적인 클러스터링으로 모바일 엣지 서버에 집중되는 태스크 로드를 분산하고 데이터 처리 속도를 개선하였다. 향후에도 모바일 엣지 컴퓨팅에서 트래픽 로드 분산과 효율적인 데이터 처리를 위해 머신러닝의 모델 활용과 데이터 패턴에 따른 분석으로 모바일 엣지 컴퓨팅에서 협업 모델을 연구하고자 한다.

REFERENCE

[1] W. Yu, F. Liang, X. He, W.G. Hatcher, C. Lu, J. Lin, and X. Yang, "A Survey on the Edge Computing for the Internet of Things," *IEEE Access*, Vol. 6, pp. 6900-6919, 2017.

[2] H. Lin, S. Zeadally, Z. Chen, H. Labiod, and L. Wang, "A Survey on Computation Offloading Modeling for Edge Computing," *Journal of Network and Computer Applications*, Vol. 169, 102781, 2020.

[3] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, "Mobile Edge Computing: A survey," *IEEE Internet of Things Journal*, Vol. 5, Issue 1, pp. 450-465, 2017.

[4] Q. Fan and N. Ansari, "Workload Allocation in Hierarchical Cloudlet Networks," *IEEE Communications Letters*, Vol. 22, Issue 4, pp. 820-823, 2018.

[5] L. Tong, Y. Li, and W. Gao, "A Hierarchical Edge Cloud Architecture for Mobile Computing," *IEEE INFOCOM*, 2016.

[6] Z. Wei, B. Zhao, J. Su, and X. Lu, "Dynamic Edge Computation Offloading for Internet of Things with Energy Harvesting: A Learning Method," *IEEE Internet of Things Journal*, Vol. 6, Issue 3, pp. 4436-4447, 2018.

[7] Y. Mao, J. Zhang, and K.B. Letaief, "Dynamic Computation Offloading for Mobile-Edge Computing with Energy Harvesting DeVices," *IEEE Journal on Selected Areas in Communications*, Vol. 34, Issue 12, pp. 3590-3605, 2016.

[8] A. Likas, N. Vlassis, and J.J. Verbeek, "The Global K-means Clustering Algorithm," *Pattern Recognition*, Vol. 36, No. 2, pp. 451-461, 2003.

[9] F. Yi and I. Moon, "K-means based Clustering Method with a Fixed Number of Cluster Members," *Journal of Korea Multimedia Society*, Vol. 17, No. 10, pp. 1160-1170, 2014.



이 창 우

現) 국민대학교 소프트웨어학부 교수 (2019년 9월.현재)
 前) 삼성전자 Senior Professional (1998년 1월~2017년 3월)
 경희대학교 전자계산기공학 학사 / 고려대학교 컴퓨터공학 석사, 박사 수료

네트워크, 통신, 클라우드, 엣지 컴퓨팅, 인공지능, 웹/앱 프로그래밍