



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2021-0083465  
(43) 공개일자 2021년07월07일

(51) 국제특허분류(Int. Cl.)

G06F 9/455 (2018.01)

(52) CPC특허분류

G06F 9/45558 (2013.01)

G06F 2009/4557 (2019.08)

(21) 출원번호 10-2019-0175556

(22) 출원일자 2019년12월26일

심사청구일자 2019년12월26일

(71) 출원인

연세대학교 산학협력단

서울특별시 서대문구 연세로 50 (신촌동, 연세대학교)

(72) 발명자

김경섭

서울특별시 서대문구 연세로 50 4공학관 919호 (신촌동, 연세대학교)

윤호영

서울특별시 서대문구 연세로 50 4공학관 909호 (신촌동, 연세대학교)

(74) 대리인

특허법인우인

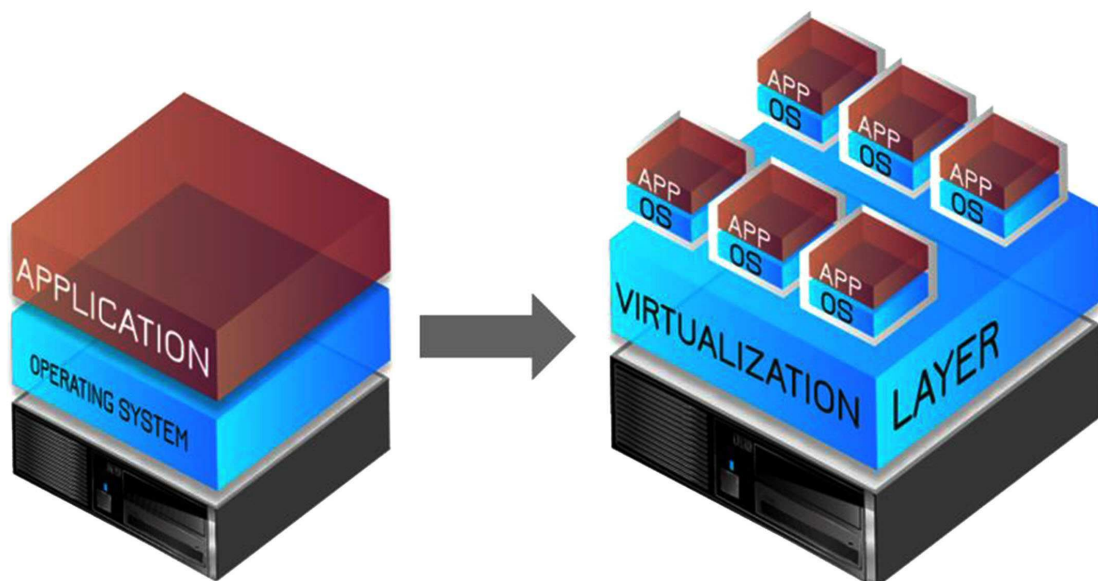
전체 청구항 수 : 총 5 항

(54) 발명의 명칭 가상머신 워크로드 클러스터링 예측을 활용한 높은 전력 효율성을 제공하는 다중 서버 관리 방법

(57) 요약

본 실시예들은 가상머신을 클러스터링하고 클러스터별로 학습된 인공지능을 활용한 워크로드 예측을 통해, 다중 서버 자원을 효율적이고 안정적으로 운영하는 다중 서버 관리 방법 및 장치를 제공한다.

대표도 - 도1



(52) CPC특허분류

G06F 2009/45591 (2019.08)

이 발명을 지원한 국가연구개발사업

|             |   |
|-------------|---|
| 과제고유번호      | 2019R1F1A1059276                        |
| 부처명         | 과학기술정보통신부                               |
| 과제관리(전문)기관명 | 한국연구재단                                  |
| 연구사업명       | 일반연구자지원사업                               |
| 연구과제명       | 클라우드 데이터센터의 최적 운영을 위한 의사결정지원 모형 연구(1/3) |
| 기 여 율       | 1/1                                     |
| 과제수행기관명     | 연세대학교 산학협력단                             |
| 연구기간        | 2019.06.01 ~ 2020.02.29                 |

---

## 명세서

### 청구범위

#### 청구항 1

복수의 서버에 혼재하여 위치하는 복수의 가상머신을 상기 가상머신의 워크로드의 패턴에 따라 복수의 클러스터로 클러스터링하는 단계;

상기 클러스터에 속하는 가상머신의 현재 워크로드로부터 워크로드 예측 모델을 통해 상기 가상머신의 다음 워크로드를 예측하는 단계;

상기 예측한 가상머신의 다음 워크로드를 기준으로 상기 복수의 가상머신을 가상머신 세트로 조합하는 단계; 및  
상기 조합한 가상머신 세트를 상기 복수의 서버에 배치하는 단계를 포함하는 다중 서버 관리 방법.

#### 청구항 2

제1항에 있어서,

상기 가상머신의 워크로드는 시계열 데이터이며,

상기 조합한 가상머신 세트를 상기 복수의 서버에 배치하는 단계는 상기 복수의 가상머신의 워크로드의 패턴을 상쇄하는 방식으로 상기 조합한 가상머신 세트의 워크로드의 분포를 최소화하고 상기 복수의 서버의 워크로드의 분포를 최소화하는 것을 특징으로 하는 다중 서버 관리 방법.

#### 청구항 3

제1항에 있어서,

상기 조합한 가상머신 세트를 상기 복수의 서버에 배치하는 단계 이후에,

상기 예측한 가상머신의 다음 워크로드의 변화가 기준 범위를 넘으면 하나의 서버에 위치한 가상머신을 다른 서버로 마이그레이션하여 상기 복수의 가상머신을 재배치하는 단계를 추가로 포함하는 것을 특징으로 하는 다중 서버 관리 방법.

#### 청구항 4

제1항에 있어서,

상기 워크로드 예측 모델은 상기 클러스터마다 독립적으로 병렬적으로 학습되며,

상기 워크로드 예측 모델은 장단기 메모리(Long Short Term Memory, LSTM) 구조의 순환신경망(Recurrent Neural Network, RNN)을 기반으로 상기 다음 워크로드를 예측하는 것을 특징으로 하는 다중 서버 관리 방법.

#### 청구항 5

제1항에 있어서,

상기 클러스터링하는 단계는 메타 데이터를 기준으로 상기 클러스터를 분류하며,

상기 메타 데이터는 상기 가상머신의 명칭, 애플리케이션 리스트, 워크로드의 유형, 및 사용 자원의 비율을 포함하며,

상기 클러스터링하는 단계는 상기 가상머신의 명칭으로부터 키워드를 추출하고 상기 키워드가 스톱워드(Stopword)에 해당하면, 상기 가상머신에 대응시킬 복수의 클러스터의 가중치를 균일하게 설정하고,

상기 클러스터링하는 단계는 상기 애플리케이션 리스트 중에서 기 설정된 사용 빈도 범위에 속하는 애플리케이션의 정보를 추출하여 상기 애플리케이션의 정보를 기준으로 상기 클러스터를 분류하고,

상기 워크로드의 유형은 (i) 제1 시간 구간에서 점진적 증가하는 형태인 제1 유형, (ii) 상기 제1 시간 구간보다 좁은 제2 시간 구간에서 급격히 증가하는 형태인 제2 유형, 및 (iii) 시간에 상관없이 일정한 형태인 제3 유형으로 구분되며,

상기 사용 자원의 비율은 프로세서의 사용률 및 메모리의 사용률로 구분되는 것을 특징으로 하는 다중 서버 관리 방법.

## 발명의 설명

### 기술 분야

[0001] 본 발명이 속하는 기술 분야는 가상머신 워크로드 클러스터링 예측을 활용한 다중 서버 관리 방법 및 장치에 관한 것이다.

### 배경 기술

[0002] 이 부분에 기술된 내용은 단순히 본 실시예에 대한 배경 정보를 제공할 뿐 종래기술을 구성하는 것은 아니다.

[0003] 가상화 기술은 컴퓨터 운영체제(OS)를 시스템 구조나 하드웨어에 영향받지 않고 설치 및 사용할 수 있는 기술이다. 하나의 물리머신(예컨대, 서버)을 가상으로 여러 조각으로 나누어, 물리머신을 효율적으로 사용할 수 있게 한다. 수십 개의 가상머신이 서버의 프로세서와 메모리 자원을 할당받는다.

[0004] 데이터센터에는 수 많은 서버들이 가상화 환경으로 설정되어 있다. 수십 내지 수백, 수천 내지 수만 대의 서버가 있는 데이터센터는 서버의 수보다 많은 가상머신의 수가 존재한다. 데이터 센터는 기본적으로 높은 계산 복잡도를 보이는 특성이 있으며, 이는 관리자의 직관으로 운영할 수 있는 수준을 훨씬 초월한다. 데이터센터는 설비 비용보다 운영 비용이 훨씬 높은 에너지 고밀도 건물로 단일건물로는 전력 사용량이 가장 많다.

[0005] 데이터센터는 365일 24시간 동안 끊임없이 가동되어야 하는 조건때문에 관리자들은 보수적인 운영방안을 채택하고 있다. 서버 운영이 중지되는 시간이 길어질수록, 피해 비용이 발생하게 된다. 데이터센터의 규모가 커지고 있으나, 여전히 관리자의 직관으로 데이터센터를 운영되는 것이 실정이다.

### 선행기술문헌

#### 특허문헌

[0006] (특허문헌 0001) 한국공개특허공보 10-2018-0105874 (2018.10.01)

## 발명의 내용

### 해결하려는 과제

[0007] 본 발명의 실시예들은 가상머신을 클러스터링하고 클러스터별로 학습된 인공지능을 활용한 워크로드 예측을 통해, 다중 서버 자원을 효율적이고 안정적으로 운영하는데 발명의 주된 목적이 있다.

[0008] 본 발명의 명시되지 않은 또 다른 목적들은 하기의 상세한 설명 및 그 효과로부터 용이하게 추론할 수 있는 범위 내에서 추가적으로 고려될 수 있다.

### 과제의 해결 수단

[0009] 본 실시예의 일 측면에 의하면, 복수의 서버에 혼재하여 위치하는 복수의 가상머신을 상기 가상머신의 워크로드의 패턴에 따라 복수의 클러스터로 클러스터링하는 단계, 상기 클러스터에 속하는 가상머신의 현재 워크로드로부터 워크로드 예측 모델을 통해 상기 가상머신의 다음 워크로드를 예측하는 단계, 상기 예측한 가상머신의 다음 워크로드를 기준으로 상기 복수의 가상머신을 가상머신 세트로 조합하는 단계, 및 상기 조합한 가상머신 세트를 상기 복수의 서버에 배치하는 단계를 포함하는 다중 서버 관리 방법을 제공한다.

[0010] 상기 가상머신의 워크로드는 시계열 데이터일 수 있다.

[0011] 상기 조합한 가상머신 세트를 상기 복수의 서버에 배치하는 단계는 상기 복수의 가상머신의 워크로드의 패턴을

상쇄하는 방식으로 상기 조합한 가상머신 세트의 워크로드의 분포를 최소화하고 상기 복수의 서버의 워크로드의 분포를 최소화할 수 있다.

- [0012] 상기 다중 서버 관리 방법은 상기 조합한 가상머신 세트를 상기 복수의 서버에 배치하는 단계 이후에, 상기 예측한 가상머신의 다음 워크로드의 변화가 기준 범위를 넘으면 하나의 서버에 위치한 가상머신을 다른 서버로 마이그레이션하여 상기 복수의 가상머신을 재배치하는 단계를 추가로 포함할 수 있다.
- [0013] 상기 워크로드 예측 모델은 상기 클러스터마다 독립적으로 병렬적으로 학습될 수 있다.
- [0014] 상기 워크로드 예측 모델은 장단기 메모리(Long Short Term Memory, LSTM) 구조의 순환신경망(Recurrent Neural Network, RNN)을 기반으로 상기 다음 워크로드를 예측할 수 있다.
- [0015] 상기 클러스터링하는 단계는 메타 데이터를 기준으로 상기 클러스터를 분류할 수 있다.
- [0016] 상기 메타 데이터는 상기 가상머신의 명칭, 애플리케이션 리스트, 워크로드의 유형, 및 사용 자원의 비율을 포함할 수 있다.
- [0017] 상기 클러스터링하는 단계는 상기 가상머신의 명칭으로부터 키워드를 추출하고 상기 키워드가 스톱워드(Stopword)에 해당하면, 상기 가상머신에 대응시킬 복수의 클러스터의 가중치를 균일하게 설정할 수 있다.
- [0018] 상기 클러스터링하는 단계는 상기 애플리케이션 리스트 중에서 기 설정된 사용 빈도 범위에 속하는 애플리케이션의 정보를 추출하여 상기 애플리케이션의 정보를 기준으로 상기 클러스터를 분류할 수 있다.
- [0019] 상기 워크로드의 유형은 (i) 제1 시간 구간에서 점진적 증가하는 형태인 제1 유형, (ii) 상기 제1 시간 구간보다 좁은 제2 시간 구간에서 급격히 증가하는 형태인 제2 유형, 및 (iii) 시간에 상관없이 일정한 형태인 제3 유형으로 구분될 수 있다.
- [0020] 상기 사용 자원의 비율은 프로세서의 사용률 및 메모리의 사용률로 구분될 수 있다.

### 발명의 효과

- [0021] 이상에서 설명한 바와 같이 본 발명의 실시예들에 의하면, 복수의 가상머신을 클러스터링하고 클러스터별로 학습된 인공지능을 활용한 워크로드 예측을 통해, 다중 서버 자원을 효율적이고 안정적으로 운영할 수 있는 효과가 있다.
- [0022] 여기에서 명시적으로 언급되지 않은 효과라 하더라도, 본 발명의 기술적 특징에 의해 기대되는 이하의 명세서에서 기재된 효과 및 그 잠정적인 효과는 본 발명의 명세서에 기재된 것과 같이 취급된다.

### 도면의 간단한 설명

- [0023] 도 1은 서버에 설치된 가상머신을 예시한 도면이다.
- 도 2 및 도 3은 본 발명의 실시예들에 따른 다중 서버 관리 방법의 전체 동작을 예시한 흐름도이다.
- 도 4는 본 발명의 일 실시예에 따른 다중 서버 관리 방법이 신규 가상머신을 클러스터링하는 동작을 예시한 흐름도이다.
- 도 5는 본 발명의 일 실시예에 따른 다중 서버 관리 방법이 처리하는 메타 데이터를 예시한 도면이다.
- 도 6은 본 발명의 일 실시예에 따른 다중 서버 관리 방법이 처리하는 클러스터 가중치를 예시한 도면이다.
- 도 7은 본 발명의 일 실시예에 따른 다중 서버 관리 방법이 클러스터별 워크로드 예측 모델을 학습하는 동작을 예시한 도면이다.
- 도 8은 본 발명의 일 실시예에 따른 다중 서버 관리 방법이 복수의 서버에 복수의 가상머신을 배치한 상태를 예시한 도면이다.
- 도 9는 본 발명의 다른 실시예에 따른 다중 서버 관리 장치를 예시한 도면이다.

### 발명을 실시하기 위한 구체적인 내용

- [0024] 이하, 본 발명을 설명함에 있어서 관련된 공지기능에 대하여 이 분야의 기술자에게 자명한 사항으로서 본 발명의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명을 생략하고, 본 발명의 일부 실시예들

을 예시적인 도면을 통해 상세하게 설명한다.

[0025] 도 1은 가상머신을 예시한 도면이다.

[0026] 가상머신(Virtual Machine, VM)은 컴퓨팅 환경을 소프트웨어로 구현한 것이다. 가상머신은 하드웨어와 독립적인 프로그램 실행 환경과 추상화를 제공하도록 설계된다. 가상머신에 운영체제 또는 애플리케이션을 설치하고 실행할 수 있다.

[0027] 수 많은 서버는 가상화 환경으로 설정되어 있다. 서버에 설치된 가상머신은 365일 24시간 가동되면서 CPU, Memory, Disk I/O, Network I/O 등의 자원을 사용한다. 이러한 자원 사용을 워크로드라고 한다. 데이터센터 관리자는 분당 발생하는 워크로드를 모니터링할 수 있는 관리자 도구를 활용하여, 데이터센터의 수 많은 자원들이 정상적으로 작동되고 있는지 모니터링한다. 모니터링 결과 서버에 이상이 발생하면 관리자는 대응 조치를 취한다.

[0028] 가상화 기술은 마이그레이션 기능을 지원한다. 마이그레이션은 한 서버에 위치한 가상머신을 다른 서버로 이동하는 기능이다. 서버에 할당되어 있는 가상머신을 다른 서버로 모두 옮기면, 서버는 휴지(Idle) 상태가 되고, 절전 모드로 전환하거나 전원을 끌 수 있다. 가상머신 재배치를 활용하여, 서버를 절전하는 것을 가상머신 통합(Virtual Machine Consolidation)이라고 한다. 가상머신 통합은 어떤 서버에 위치한 어떤 가상머신을 다른 어떤 서버로 마이그레이션할 건지를 결정하는 문제이다.

[0029] 가상머신을 마이그레이션하는 것은 시간이 소요되며, 마이그레이션하는 작업 자체가 과부하가 발생하는 작업이다. 마이그레이션하는 동안에도 서비스는 중단없이 제공되지만, 시간 간격이 존재하기 때문에 서비스의 품질이 저하될 수 있다. 관리자는 어떤 가상머신이 어떤 서버에 위치했는지 어느 정도 파악을 할 수 있어야 하는데, 마이그레이션이 자주 발생하게 된다면 운영 관점에서도 관리 포인트가 증가하게 된다. 가상머신 통합은 가상머신 마이그레이션이 최소화 되어야 한다는 조건이 고려되어야 한다.

[0030] 기존의 가상머신 통합 연구들은 서버의 환경을 24시간 모니터링하면서, 과부하가 발생하였거나, 저부하 상태의 서버가 감지되었을 때 가상머신을 마이그레이션한다. 기존의 방식은 어느 정도의 임계치(Threshold)를 여유있게 설정하긴 하지만 과부하가 발생하였을 때, 가상머신을 어디로 마이그레이션할 것인지에 대한 연산을 수행해야 하고, 과부하가 발생한 서버에서 과부하를 일으키는 가상머신 마이그레이션 작업을 무리하게 수행해야 한다.

[0031] 본 발명의 실시예들에 따른 다중 서버 관리 방법은 인공 지능(딥 러닝) 기술을 적용하여 서버 자원이 사용하고 있는 시계열적인 워크로드 데이터를 예측하여, 데이터 센터의 서버 운영에 영향을 주는 큰 문제가 발생하기 전에 사전 조치를 취한다. 다중 서버 관리 방법은 복수의 가상머신을 클러스터링하고 클러스터별로 학습된 인공지능을 활용한 워크로드 예측을 통해, 다중 서버 자원을 효율적이고 안정적으로 운영할 수 있다.

[0032] 도 2 및 도 3은 본 발명의 실시예들에 따른 다중 서버 관리 방법의 전체 동작을 예시한 흐름도이다.

[0033] 다중 서버 관리 방법은 다중 서버 관리 장치에 의해 수행될 수 있다. 다중 서버 관리 장치의 구성은 도 9에 예시적으로 도시되어 있다.

[0034] 단계 S210에서 프로세서는 복수의 서버에 혼재하여 위치하는 복수의 가상머신을 가상머신의 워크로드의 패턴에 따라 복수의 클러스터로 클러스터링한다. 가상머신의 워크로드는 시계열 데이터이며, 가상머신의 자원 사용 정보이다.

[0035] 클러스터링하는 단계(S210)는 메타 데이터를 기준으로 클러스터를 분류할 수 있다. 메타 데이터는 가상머신의 명칭, 애플리케이션 리스트, 워크로드의 유형, 및 사용 자원의 비율을 포함할 수 있다.

[0036] 클러스터링하는 단계(S210)는 가상머신의 명칭으로부터 키워드를 추출하고 키워드가 스톱워드(Stopword)에 해당하면, 가상머신에 대응시킬 복수의 클러스터의 가중치를 균일하게 설정할 수 있다.

[0037] 클러스터링하는 단계(S210)는 애플리케이션 리스트 중에서 기 설정된 사용 빈도 범위에 속하는 애플리케이션의 정보를 추출하여 애플리케이션의 정보를 기준으로 상기 클러스터를 분류할 수 있다.

[0038] 단계 S220에서 프로세서는 클러스터에 속하는 가상머신의 현재 워크로드로부터 워크로드 예측 모델을 통해 가상머신의 다음 워크로드를 예측한다. 워크로드 예측 모델은 클러스터마다 독립적으로 병렬적으로 학습될 수 있다. 워크로드 예측 모델은 장단기 메모리(Long Short Term Memory, LSTM) 구조의 순환신경망(Recurrent Neural Network, RNN)을 기반으로 다음 워크로드를 예측할 수 있다.



- [0039] 단계 S230에서 프로세서는 예측한 가상머신의 다음 워크로드를 기준으로 복수의 가상머신을 가상머신 세트에 포함한다.
- [0040] 단계 S240에서 프로세서는 조합한 가상머신 세트를 상기 복수의 서버에 배치한다. 조합한 가상머신 세트를 상기 복수의 서버에 배치하는 단계(S240)는 복수의 가상머신의 워크로드의 패턴을 상쇄하는 방식으로 조합한 가상머신 세트의 워크로드의 분포를 최소화하고 복수의 서버의 워크로드의 분포를 최소화할 수 있다.
- [0041] 단계 S250에서 프로세서는 조합한 가상머신 세트를 상기 복수의 서버에 배치하는 단계 이후에, 예측한 가상머신의 다음 워크로드의 변화가 기준 범위를 넘으면 하나의 서버에 위치한 가상머신을 다른 서버로 마이그레이션하여 복수의 가상머신을 재배포할 수 있다.
- [0042] 단계 S310에서 다중 서버 관리 장치는 가상머신 메타 데이터 데이터베이스로부터 메타 데이터를 독출한다. 메타 데이터를 분석하여, 데이터 유형에 맞게 정의한다.
- [0043] 단계 S320에서 다중 서버 관리 장치는 가상머신을 클러스터링한다. 다중 서버 관리 장치는 가상머신 클러스터 정보를 기반으로 워크로드 데이터 학습 데이터 베이스를 구축한다.
- [0044] 단계 S330에서 다중 서버 관리 장치는 워크로드 데이터 학습 데이터 베이스를 이용하여 가상머신 워크로드 클러스터링을 예측한다. 가상머신이 어떤 클러스터로 분류되는지 파악하고, 클러스터의 학습파일을 토대로 CPU, Memory 등의 워크로드를 시계열적으로 예측한다.
- [0045] 단계 S340에서 다중 서버 관리 장치는 가상머신 별 워크로드 예측값에 따라 가상머신 조합을 탐색한다. 가상머신 워크로드 예측값을 토대로 서버의 워크로드가 안정적으로 유지되고, 각 서버별로 워크로드 균형이 맞는 가상머신 조합을 탐색한다.
- [0046] 단계 S350에서 다중 서버 관리 장치는 가상머신 마이그레이션을 실행한다. 도출된 가상머신 조합이 이전의 조합과 다르다면, 새롭게 도출된 가상머신 조합에 따라 가상머신을 마이그레이션한다.
- [0047] 도 4는 본 발명의 일 실시예에 따른 다중 서버 관리 방법이 신규 가상머신을 클러스터링하는 동작을 예시한 흐름도이다.
- [0048] 단계 S410에서 프로세서는 신규 가상머신의 명칭 키워드를 추출한다.
- [0049] 단계 S420에서 프로세서는 키워드가 스톱워드에 해당하는지 여부를 판단한다. 가상머신의 이름이 직관적으로 작성되지 않았는지를 검사한다. 예컨대, test123, aaa123, abc 등의 명칭이 있을 수 있다.
- [0050] 단계 S430에서 프로세서는 키워드가 스톱워드에 해당하면, 클러스터 가중치를 균일하게 분배한다. 가상머신을 분류할 수 없기 때문에 Cluster의 weight를 균일하게 분배한다. 예컨대, New 가상머신 #100은 Cluster #1은 0.25, Cluster #2는 0.25, Cluster #3은 0.25, Cluster #4는 0.25의 가중치를 갖도록 설정할 수 있다.
- [0051] 단계 S440에서 프로세서는 키워드가 스톱워드에 해당하지 않으면, 가상머신 클러스터링을 분류한다(S440).
- [0052] 도 5는 본 발명의 일 실시예에 따른 다중 서버 관리 방법이 처리하는 메타 데이터를 예시한 도면이다.
- [0053] 시계열 데이터의 클러스터링은 시간대별로 구분하여 진행한다. 가상머신의 워크로드는 시계열 데이터이므로 시간에 따라 고유한 패턴을 보인다. 주말보다는 평일에 높은 워크로드를 보이는지, 새벽보다는 일과시간에 높은 워크로드를 보이는지 등 시간대별로 패턴을 분류할 수 있다. 가상머신 통합의 목적 자체가 워크로드를 안정하게 만드는 가상머신 조합을 탐색하는 것이므로 시간대별로 구분이 가능하다.
- [0054] 가상머신의 패턴을 시간대별로만 구분하는 것은 간단하지만 정확도를 높이기 위해 별도의 정보들을 추가적으로 고려한다. 도 5를 참조하면, 가상머신을 클러스터링하는데 고려되는 정보가 예시되어 있다.
- [0055] 메타 데이터는 가상머신의 명칭, 애플리케이션 리스트, 워크로드의 유형, 및 사용 자원의 비율을 포함할 수 있다. 서버와 관련된 프로그래밍 인터페이스를 통해 메타 데이터를 추출할 수 있다.
- [0056] 관리자가 가상머신을 처음 생성할 때 가상머신의 명칭을 직관적으로 작성하므로, 어떤 가상머신인지 파악하는데 단서를 제공한다.
- [0057] 애플리케이션, 즉, 어떤 프로세스를 사용하는지를 파악하면 해당 가상머신이 어떤 특징이 있는지 알 수 있다.
- [0058] 워크로드의 유형은 (i) 제1 시간 구간에서 점진적 증가하는 형태인 제1 유형, (ii) 제1 시간 구간보다 좁은 제2 시간 구간에서 급격히 증가하는 형태인 제2 유형, 및 (iii) 시간에 상관없이 일정한 형태인 제3 유형으로 구분

될 수 있다. 예컨대, 제1 유형은 업무 시간대를 기준으로 서버의 활용률이 점진적으로 증가하는 형태인 일반(Normal) 유형으로, 제2 유형은 특정 시간대에 활용률이 급격히 증가하는 형태인 경사(Ramp) 유형으로, 제3 유형은 시간에 관계없이 일정한 형태인 안정(Steady) 유형으로 설정할 수 있다.

- [0059] 사용 자원의 비율은 프로세서의 사용률 및 메모리의 사용률로 구분될 수 있다. 예컨대, CPU와 Memory의 항목의 합은 1이 되어야 하며, 두 항목이 0.5이면, 균일하게 사용한다는 의미이다.
- [0060] 도 6은 본 발명의 일 실시예에 따른 다중 서버 관리 방법이 처리하는 클러스터 가중치를 예시한 도면이다.
- [0061] 클러스터링 기법은 퍼지 클러스터링(Fuzzy Clustering)을 적용하여, 하나의 가상머신이 여러 개의 클러스터에 포함될 수 있으며, 각 클러스터에 얼마나 밀접한지 가중치 형태로 도출된다. 퍼지 클러스터링은 각 관측치가 하나의 클러스터에만 속하는 것이 아니라 여러 클러스터에 속할 수 있으며, 각 군집에 속할 가능성(Possibility) 또는 확률(Probability)을 제시한다.
- [0062] 도 6을 참조하면, 가상머신 #1은 클러스터 #1에 0.82의 가중치를 가지며, 클러스터 #2와 클러스터 #5에 각각 0.14, 0.04의 가중치를 갖는다. 이는 가상머신 #1은 클러스터 #1, #2, #5에 해당한다는 의미이며, 예측을 할 때, 클러스터 #1, #2, #5의 학습 결과를 사용하여, 가중치를 기반으로 예측한다.
- [0063] 도 7은 본 발명의 일 실시예에 따른 다중 서버 관리 방법이 클러스터별 워크로드 예측 모델을 학습하는 동작을 예시한 도면이다.
- [0064] 가상머신 통합에 워크로드 패턴을 고려하며 각 가상머신별로 워크로드 패턴을 예측하고, 가상머신 재배치를 결정한다.
- [0065] 인공지능 기반의 전형적인 예측 모형은 여러 가지 패턴들을 수집하여, 인공지능 알고리즘을 통해 학습을 진행한다. 해당 학습 모델에 예측할 데이터를 입력하면, 예측 데이터가 출력되는 형식이다.
- [0066] VM #1 내지 VM #10은 학습에 사용할 수집 데이터이다. LSTM 알고리즘을 통해 학습하게 되면, 학습 모델이 생성된다. 예측하려는 객체(Target VM)의 기존 값을 생성된 학습 모델에 입력하면, 기존 학습 모델에 저장된 패턴에 따라 객체의 값이 예측(Predicted VM)된다.
- [0067] RNN은 연속성을 지니고 앞뒤의 순서가 중요한 패턴을 표현하고 학습하는 데 유용한 신경망이고, LSTM 모델은 시간적 순서에 따른 데이터를 인코딩을 위해 사용된다. LSTM은 데이터를 분석하고 벡터를 생성한다.
- [0068] RNN은 은닉 레이어의 결과가 다시 같은 은닉 레이어의 입력으로 들어가도록 연결된 루프가 있고, 과거의 데이터가 미래에 영향을 주는 구조를 갖는다. 즉, RNN은 히든 노드가 방향을 가진 엣지로 연결되어 순환구조를 이룬다.
- [0069] LSTM 모델은 은닉 레이어에 여러 개의 게이트가 연결된 셀을 추가한 구조이다. 은닉 레이어는 입력 게이트(Input Gate), 출력 게이트(Output Gate), 망각 게이트(Forget Gate)를 포함하는 셀, 즉 메모리 블록(Memory Block)을 갖는다. 망각 게이트는 과거 정보를 잊기를 위한 게이트이고, 입력 게이트는 현재 정보를 기억하기 위한 게이트이다. 게이트는 각각 세기 및 방향을 가진다. 셀은 컨베이어 벨트 역할을 하고, 상태가 오래 경과하더라도 그래디언트가 비교적 전파를 유지할 수 있다. 망각 게이트는 이전 상태의 정보를 얼마나 기억할 것인지를 결정하는 단계로 0에서 1 사이 값이 출력되고 0이면 이전 상태의 정보를 완전히 잊는 것이고 1이면 이전의 정보를 온전히 기억하는 것이 될 수 있다.
- [0070] 본 실시예에 따른 다중 서버 관리 방법은 학습할 데이터들의 패턴을 분석하여, N(자연수) 개의 클러스터로 구분하고, 해당 클러스터에 속하는 데이터들끼리 학습을 진행한다. 즉, N 번을 진행한다. 예컨대, 가상머신 데이터들을 4 개의 패턴으로 나누었고, 각 패턴에 속하는 가상머신 데이터들을 모아서 학습을 진행하여, 학습 모델을 관리한다. 패턴의 개수는 요구되는 설계사항에 따라 다양한 수치로 설정될 수 있다.
- [0071] 이러한 분할 클러스터링 학습은 전형적인 예측 모델에 비해, 패턴의 특수성을 유지할 수 있으며, 학습을 병렬적으로 진행한다면, 학습 시간을 최소화할 수 있다.
- [0072] 도 8은 본 발명의 일 실시예에 따른 다중 서버 관리 방법이 복수의 서버에 복수의 가상머신을 배치한 상태를 예시한 도면이다.
- [0073] 도 8의 좌측을 보면, 서버 #1에 가상머신 #1이 할당되고, 서버 #2에 가상머신 #2, 가상머신 #3, 가상머신 #4이 할당되고, 서버 #3에 가상머신 #5, 가상머신 #6이 할당된 것으로 가정하면, 서버 #1은 저부하 상태로 해석될 수



있고, 서버 #2는 과부하 상태에 가깝다고 해석할 수 있다.

- [0074] 본 실시예에 따른 다중 서버 관리 방법은 서버의 시계열적인 워크로드 데이터를 감시하여 서버의 워크로드가 계속해서 안정적으로 운영한다. 즉, 가상머신의 마이그레이션을 최소화한다
- [0075] 도 8의 우측을 보면, 서버 #1과 서버 #2, 서버 #3이 저부하, 과부하 상태가 아닌 비교적 비슷한 수준을 유지하고 있고, 시간에 따라 안정적인 모습을 확인할 수 있다. 즉, 조합한 가상머신 세트의 워크로드의 분포를 최소화하고 복수의 서버의 워크로드의 분포를 최소화할 수 있다.
- [0076] 서버의 워크로드가 최대한 안정적으로 유지하여 마이그레이션을 최소화한다. 한 서버의 워크로드의 표준편차가 0에 가까울수록 안정한 상태라고 정의할 수 있고, 각 서버의 워크로드 표준편차를 최소화한다.
- [0077] 서버는 부하가 높아질수록 온도가 상승하여, 장비의 수명에 영향을 준다. 한 서버에 가상머신이 몰려 부하 수준이 높다면, 해당 서버의 수명은 줄어들고, 고장률이 높아지게 된다. 각 서버별로 워크로드의 균형을 맞춘다. 각 서버별 워크로드의 표준편차를 최소화한다.
- [0078] 본 실시예에 따른 가상머신 통합은 서로 상반되는 패턴을 보이는 가상머신들을 조합하여, 패턴을 상쇄시켜 워크로드의 표준편차를 0에 가깝게 하고, 각 서버별 워크로드 편차도 0에 가깝게 한다.
- [0079] 도 9는 본 발명의 다른 실시예에 따른 다중 서버 관리 장치를 예시한 도면이다.
- [0080] 다중 서버 관리 장치(12)는 적어도 하나의 프로세서(14), 컴퓨터 판독 가능한 저장매체(16) 및 통신 버스(18)를 포함한다.
- [0081] 프로세서(14)는 다중 서버 관리 장치(12)로 동작하도록 제어할 수 있다. 예컨대, 프로세서(14)는 컴퓨터 판독 가능한 저장 매체(16)에 저장된 하나 이상의 프로그램들을 실행할 수 있다. 하나 이상의 프로그램들은 하나 이상의 컴퓨터 실행 가능 명령어를 포함할 수 있으며, 컴퓨터 실행 가능 명령어는 프로세서(14)에 의해 실행되는 경우 다중 서버 관리 장치(12)로 하여금 예시적인 실시예에 따른 동작들을 수행하도록 구성될 수 있다.
- [0082] 컴퓨터 판독 가능한 저장 매체(16)는 컴퓨터 실행 가능 명령어 내지 프로그램 코드, 프로그램 데이터 및/또는 다른 적합한 형태의 정보를 저장하도록 구성된다. 컴퓨터 판독 가능한 저장 매체(16)에 저장된 프로그램(20)은 프로세서(14)에 의해 실행 가능한 명령어의 집합을 포함한다. 일 실시예에서, 컴퓨터 판독한 가능 저장 매체(16)는 메모리(랜덤 액세스 메모리와 같은 휘발성 메모리, 비휘발성 메모리, 또는 이들의 적절한 조합), 하나 이상의 자기 디스크 저장 디바이스들, 광학 디스크 저장 디바이스들, 플래시 메모리 디바이스들, 그 밖에 다중 서버 관리 장치(12)에 의해 액세스되고 원하는 정보를 저장할 수 있는 다른 형태의 저장 매체, 또는 이들의 적합한 조합일 수 있다.
- [0083] 통신 버스(18)는 프로세서(14), 컴퓨터 판독 가능한 저장 매체(16)를 포함하여 다중 서버 관리 장치(12)의 다른 다양한 컴포넌트들을 상호 연결한다.
- [0084] 다중 서버 관리 장치(12)는 또한 하나 이상의 입출력 장치를 위한 인터페이스를 제공하는 하나 이상의 입출력 인터페이스(22) 및 하나 이상의 통신 인터페이스(26)를 포함할 수 있다. 입출력 인터페이스(22) 및 통신 인터페이스(26)는 통신 버스(18)에 연결된다. 입출력 장치는 입출력 인터페이스(22)를 통해 다중 서버 관리 장치(12)의 다른 컴포넌트들에 연결될 수 있다.
- [0085] 서버는 하드웨어, 펌웨어, 소프트웨어 또는 이들의 조합에 의해 로직회로 내에서 구현될 수 있고, 범용 또는 특정 목적 컴퓨터를 이용하여 구현될 수도 있다. 서버는 고정배선형(Hardwired) 기기, 필드 프로그램 가능한 게이트 어레이(Field Programmable Gate Array, FPGA), 주문형 반도체(Application Specific Integrated Circuit, ASIC) 등을 이용하여 구현될 수 있다. 또한, 서버는 하나 이상의 프로세서 및 컨트롤러를 포함한 시스템온칩(System on Chip, SoC)으로 구현될 수 있다.
- [0086] 서버는 하드웨어적 요소가 마련된 컴퓨팅 디바이스에 소프트웨어, 하드웨어, 또는 이들의 조합하는 형태로 탑재될 수 있다. 서버는 각종 기기 또는 유무선 통신망과 통신을 수행하기 위한 통신 모듈 등의 통신장치, 프로그램을 실행하기 위한 데이터를 저장하는 메모리, 프로그램을 실행하여 연산 및 명령하기 위한 마이크로프로세서 등을 전부 또는 일부 포함한 다양한 장치를 의미할 수 있다.
- [0087] 도 2 내지 도 4에서는 각각의 과정을 순차적으로 실행하는 것으로 기재하고 있으나 이는 예시적으로 설명한 것에 불과하고, 이 분야의 기술자라면 본 발명의 실시예의 본질적인 특성에서 벗어나지 않는 범위에서 도 2 내지 도 4에 기재된 순서를 변경하여 실행하거나 또는 하나 이상의 과정을 병렬적으로 실행하거나 다른 과정을 추가

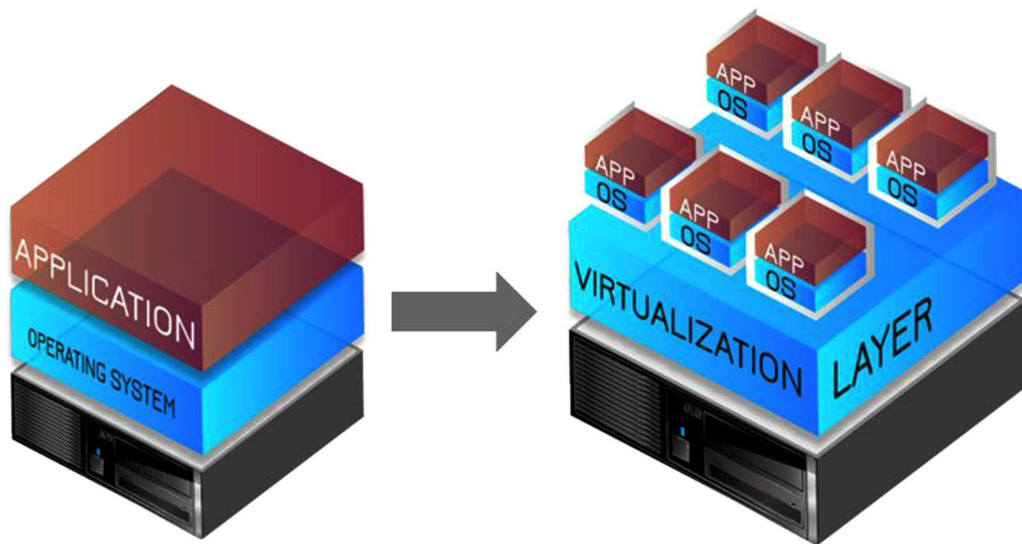
하는 것으로 다양하게 수정 및 변형하여 적용 가능할 것이다.

[0088] 본 실시예들에 따른 동작은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능한 매체에 기록될 수 있다. 컴퓨터 판독 가능한 매체는 실행을 위해 프로세서에 명령어를 제공하는 데 참여한 임의의 매체를 나타낸다. 컴퓨터 판독 가능한 매체는 프로그램 명령, 데이터 파일, 데이터 구조 또는 이들의 조합을 포함할 수 있다. 예를 들면, 자기 매체, 광기록 매체, 메모리 등이 있을 수 있다. 컴퓨터 프로그램은 네트워크로 연결된 컴퓨터 시스템 상에 분산되어 분산 방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수도 있다. 본 실시예를 구현하기 위한 기능적인(Functional) 프로그램, 코드, 및 코드 세그먼트들은 본 실시예가 속하는 기술분야의 프로그래머들에 의해 용이하게 추론될 수 있을 것이다.

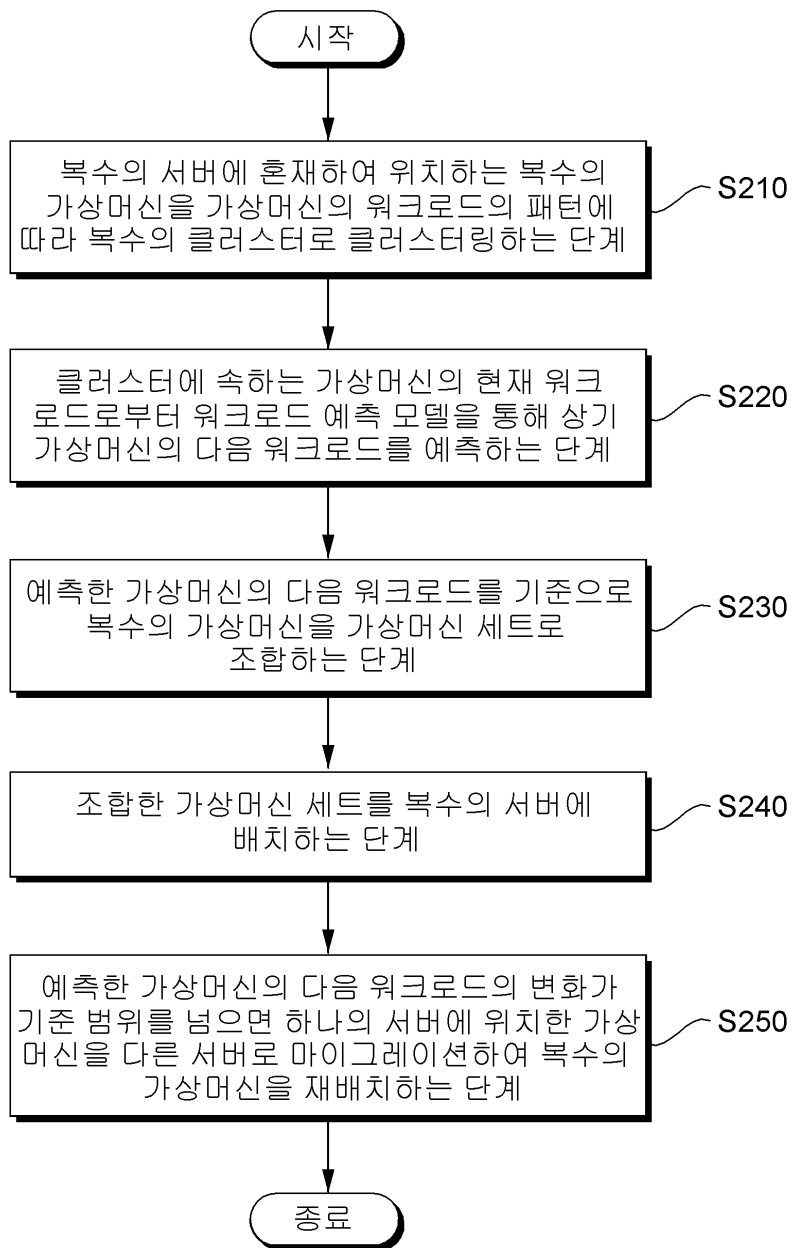
[0089] 본 실시예들은 본 실시예의 기술 사상을 설명하기 위한 것이고, 이러한 실시예에 의하여 본 실시예의 기술 사상의 범위가 한정되는 것은 아니다. 본 실시예의 보호 범위는 아래의 청구범위에 의하여 해석되어야 하며, 그와 동등한 범위 내에 있는 모든 기술 사상은 본 실시예의 권리범위에 포함되는 것으로 해석되어야 할 것이다.

## 도면

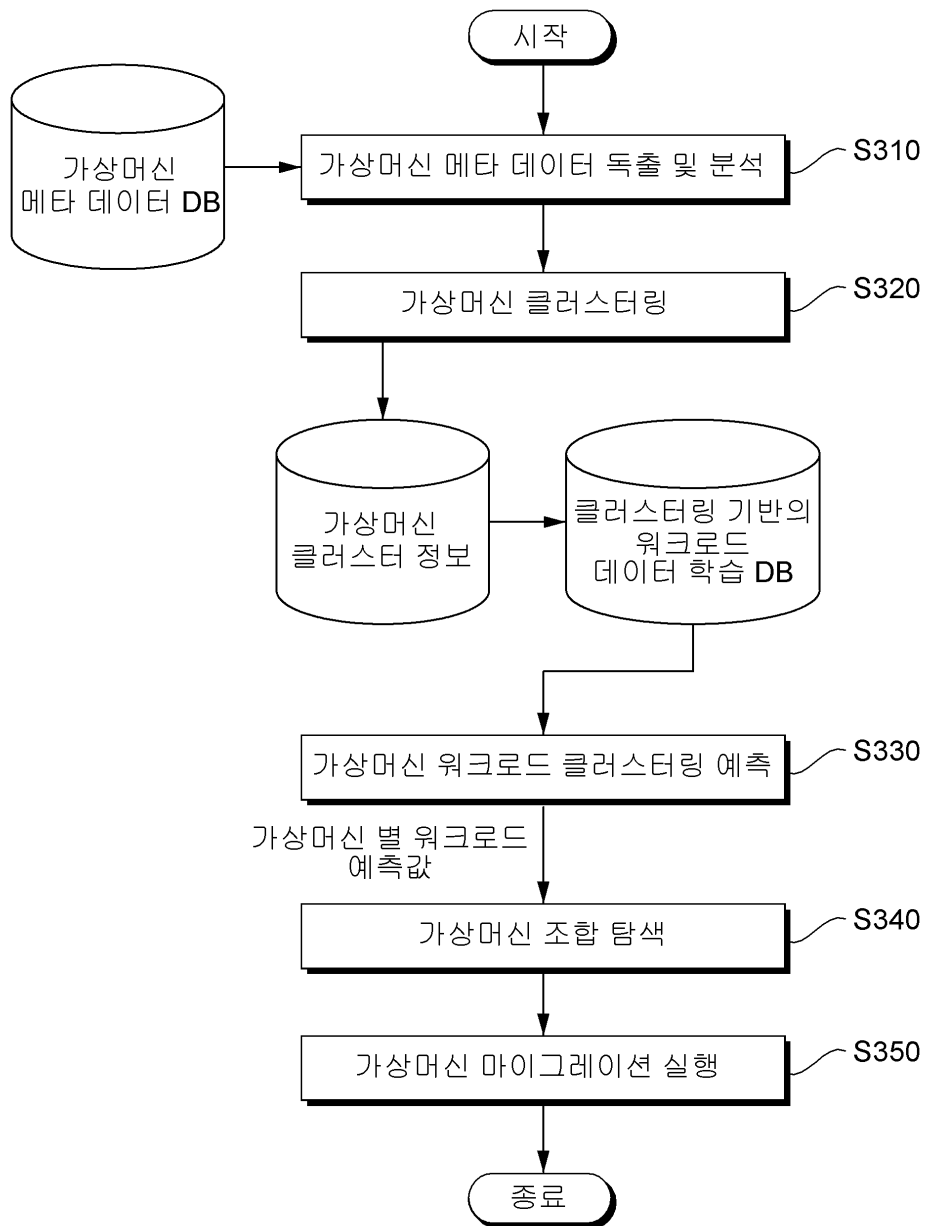
### 도면1



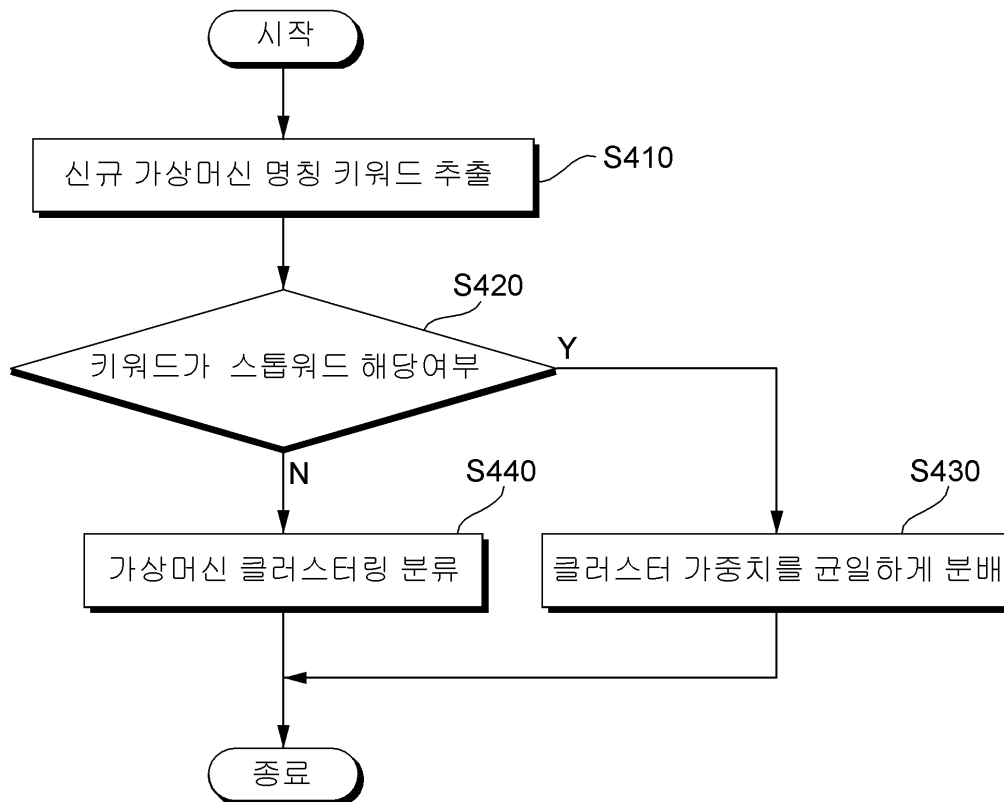
도면2



도면3



도면4



도면5

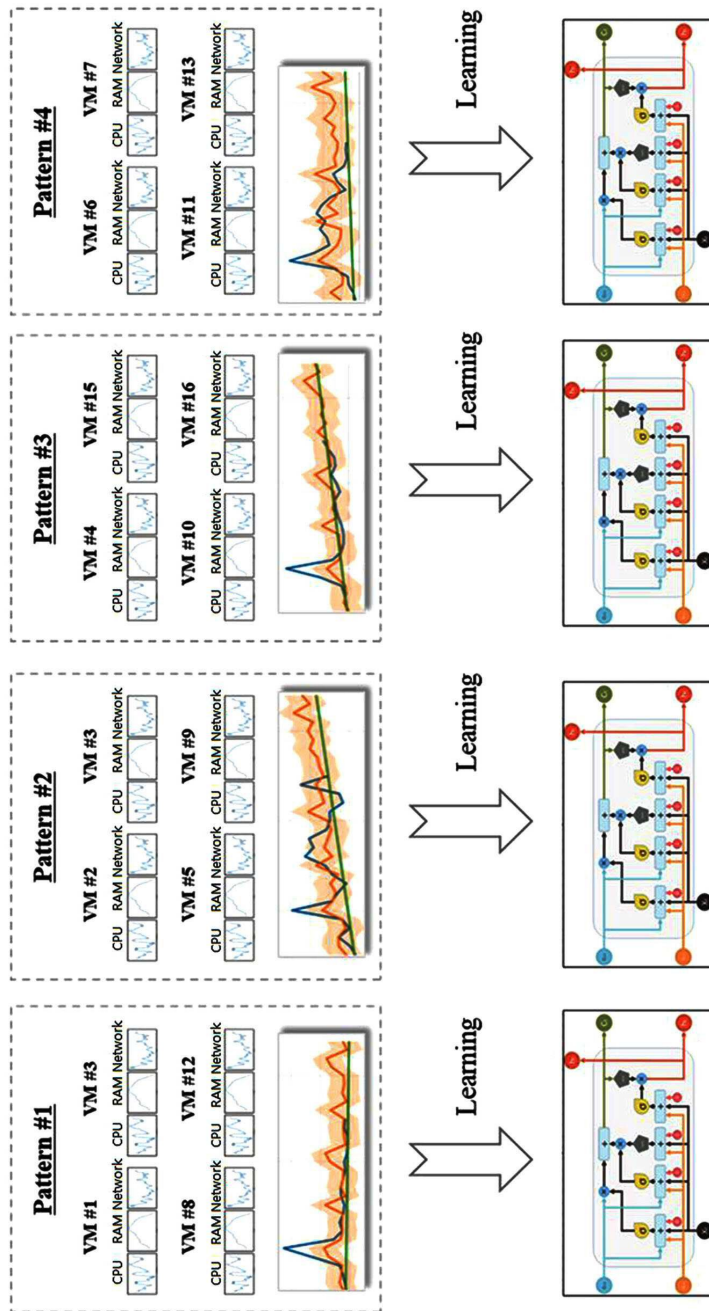
| 메타 데이터               | 설명   | 데이터 유형      |
|----------------------|--|-------------|
| 가상머신 명칭              | 가상머신의 명칭을 키워드로 추출                                      | String      |
| 애플리케이션 리스트           | 사용 빈도가 높은 애플리케이션의 정보를 추출                               | String[]    |
| 가상머신의 워크로드 유형 정의     | 가상머신의 워크로드 유형을 정의<br>(Normal, Ramp, Steady)            | Int (0,1,2) |
| 가상머신의 사용 자원 정의 - CPU | CPU를 많이 쓰는지? Memory를 많이 쓰는지? CPU와 Memory를 균형있게 쓰는지의 비율 | Double      |
| 가상머신의 사용 자원 정의 - MEM |  | Double      |

도면6

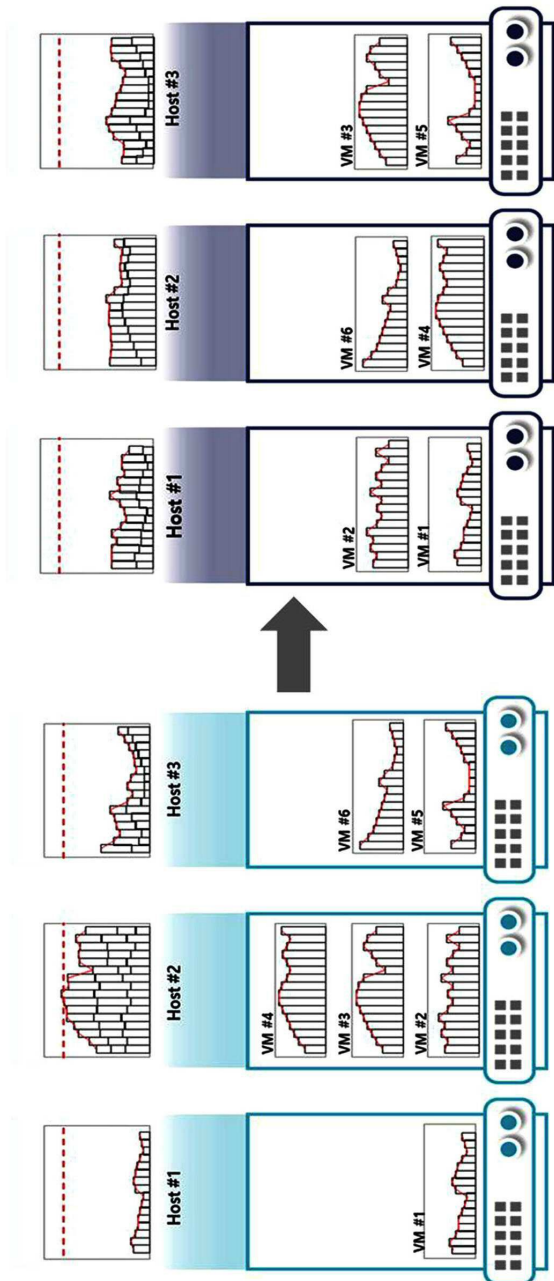
|       | Cluster #1 | Cluster #2 | Cluster #3 | Cluster #4 | Cluster #5 |
|-------|------------|------------|------------|------------|------------|
| VM #1 | 0.82       | 0.14       | 0          | 0          | 0.04       |
| VM #2 | 0.05       | 0.05       | 0.56       | 0.01       | 0.33       |
| VM #3 | 0          | 0.33       | 0.1        | 0          | 0.57       |
| VM #4 | 0.59       | 0          | 0          | 0.41       | 0          |
| VM #5 | 0.7        | 0          | 0.3        | 0          | 0          |



도면7



도면8



도면9

