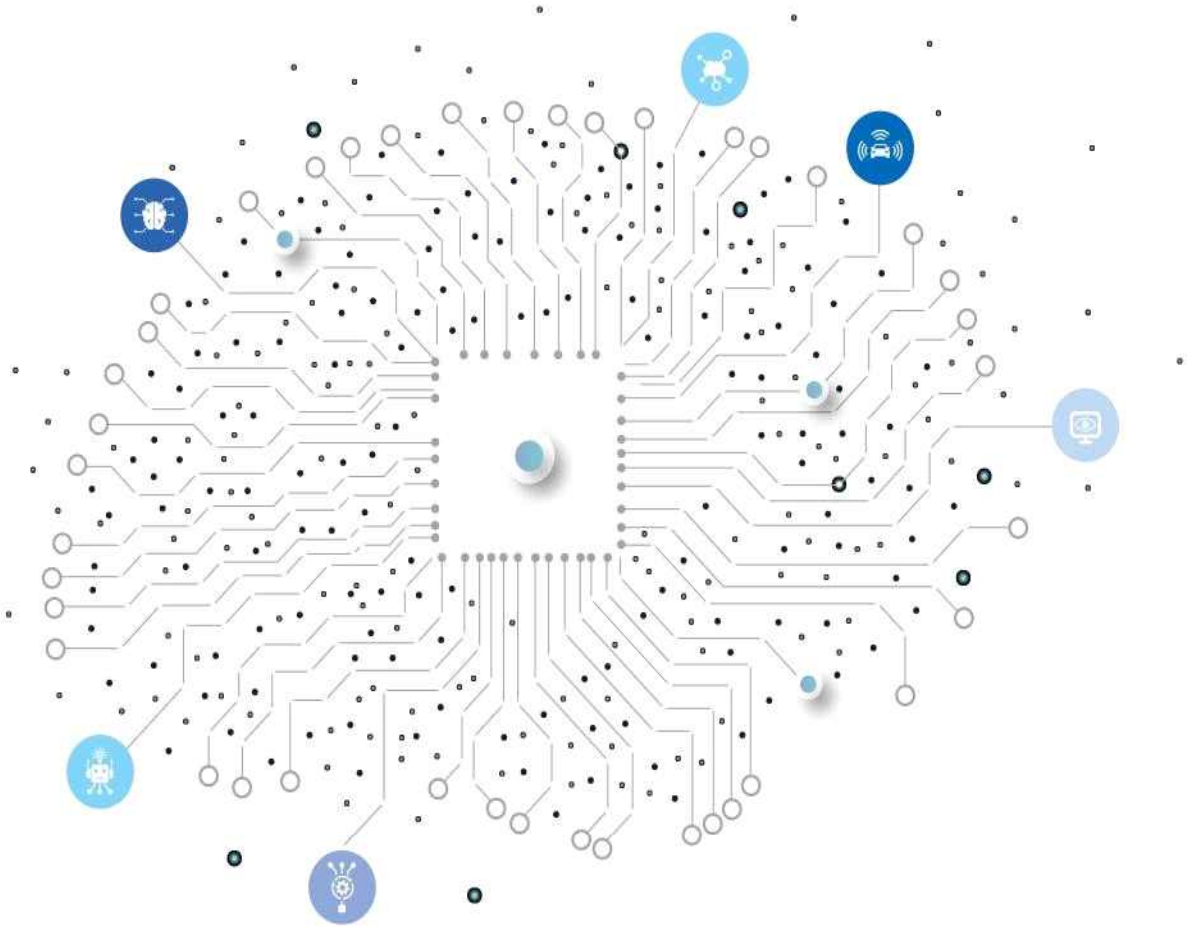
AI Network Lab
INSIGHT
Vol.7
(2021.07)



자율형 네트워크를 위한 강화학습 연구 동향

| 작 성 | 염성웅, 김경백 (전남대학교)

NIA 한국지능정보사회진흥원
NATIONAL INFORMATION SOCIETY AGENCY

o 『AI Network Lab 인사이트』는 인공지능, 클라우드, 5G 등 4차 산업혁명의 핵심인 지능정보기술과 네트워크 신기술에 대한 동향을 간략하고 심도 있게 분석한 보고서입니다.

o 본 연구보고서는 과학기술정보통신부의 방송통신발전기금조성사업, 한국지능정보사회진흥원의 초연결지능형연구개발망 구축·운영사업의 연구과제 결과이며, 한국지능정보사회진흥원/한국능률협회와 공동 기획하였습니다.

o 본 보고서의 내용은 한국지능정보사회진흥원의 공식 견해와 다를 수 있습니다.

발 행 처 한국지능정보사회진흥원

발 행 인 문용식

기 획 한국지능정보사회진흥원 지능형인프라본부 미래네트워크센터

보고서 온라인 서비스 www.nia.or.kr

Contents

개요

네트워크 복잡성이 증가하고 서비스 요구사항이 더욱 엄격해지며 다양해짐에 따라 네트워크 자동화 기술은 여러 기업으로부터 큰 관심을 받고 있다. 이러한 네트워크 자동화는 네트워크로부터 관찰되는 상태 데이터를 수집하고, 의사 결정에 사용되는 지식을 추출하며, 주어진 네트워크 자원으로 요구되는 네트워크 서비스를 관리하기 위한 제어하는 기능의 정의가 필요하다. 이러한 기능을 지원하기 위해, 장치 측면에서 네트워크 측면에 이르기까지 유연성과 안정성을 향상시키기 위해 여러 가지 인공지능 기반 기술이 제안되고 있다. 인공지능 기반 기술 중 강화학습은 네트워크 엔티티가 네트워크 환경의 불확실성을 고려한 상황에서 네트워크 성능을 최대화하기 위해 네트워크 상태를 고려하여 의사 결정 또는 행동을 포함한 최적의 정책을 얻을 수 있도록 사용된다. 이러한 강화학습은 시시각각 변하는 네트워크 서비스 요청 및 네트워크 상황을 지속적으로 학습함으로써, 자율적으로 네트워크를 관리하는 서비스를 가능하게 한다.

이 보고서에서는 자율형 네트워크를 위한 강화학습 기술의 활용 가능성에 초점을 맞추고 있다. 특히, 5G, 6G 및 SDN/NFV로 구성되는 차세대 네트워크 환경에서 강화학습을 기반으로 라우팅, 리소스 관리, 네트워크 보안 및 QoS/QoE 문제를 해결하는 기술들을 조사하여 정리하였다. 또한, 강화학습 기술 적용에 대한 중요한 과제, 미해결 문제 및 향후 연구 방향을 제안한다.

보 고 서 요 약

(1) 자율형 네트워크에서 강화학습의 필요성

- 에너지, 무선통신, 이동통신 및 사물인터넷 기술의 발달에 따라 컴퓨터 네트워크의 제어와 관리의 복잡도가 증가하고 있다. 이에 따라 이러한 네트워크의 제어와 관리의 복잡도를 낮추기 위해 제어와 관리를 자체적으로 수행하는 자율형 네트워크 기술이 등장하게 되었다. 이러한 지능화된 제어 및 관리를 위해 SDN/NFV 기반 플랫폼에 기계학습 기술을 접목하는 연구가 수행됐으며, 더욱 높은 유연성을 확보하는 방향으로 연구가 진행되면서 자율형 네트워크 프레임워크에 강화학습이 적용되고 있다.

(2) 강화학습 연구 동향

- 최근 네트워크 시스템 최적화 문제를 풀기 위해 인공지능 알고리즘 중 하나인 강화학습에 관한 많은 연구 및 개발이 이루어지고 있다. 강화학습은 네트워크 관리 시스템상의 강화학습 에이전트가 네트워크 환경에서 파생되는 정보를 이용하여 보상함수를 구성하고 반복적인 개선을 통해 최적의 목표를 달성하는 시스템 제어 방법이다. 이를 위해 강화학습 에이전트는 환경에 대한 상태 변화, 에이전트의 행동 제어, 가치함수 설계, 보상함수 설계, 정책 개선 및 최적화 모델 도출이라는 유기적인 프로세스를 진행한다. 하지만 사전에 정의한 상태와 행동을 통해 출력되는 기댓값을 예측하는 방법으로 의사 결정을 위한 가치함수를 학습하기 위해서는 많은 시간을 학습에 투자해야 하며, 제공되는 과도한 환경 상태 정보에 따라 학습이 잘 이루어지지 않거나, 잘못된 목표로 학습이 진행되는 경우가 발생할 수 있다. 이러한 문제점을 극복하기 위해 시스템 보상함수를 인공지능 신경망으로 구성함으로써 학습효율 및 예측 정확도 성능을 높이는 강화학습 모델도 연구되었다. 또한 이산적이며 한정된 개수의 행동을 제어하는 모델뿐만 아니라, 제어해야 하는 행동의 자유도가 매우 높은 경우에 대해서도 효과적인 학습을 수행할 수 있는 강화학습 모델의 연구도 진행되고 있다.

(3) 네트워크 기술에 강화학습 활용 동향

- 네트워크 구조가 복잡해짐으로써 라우팅, 리소스 관리, 보안, QoS/QoE 분야에서 다양한 이슈가 발생하고 있고, 이를 해결하기 위해 다양한 환경에 대한 적응 최적화 메커니즘을 포함하는 강화학습 적용 기법들이 연구되고 있다. 라우팅 분야에서는 네트워크 트래픽이 기하급수적 증가에 따라 강화학습을 통해 라우팅 프로세스를 최적화하는 연구가 진행되고 있다. 리소스 관리 분야에서는 스마트 시티 또는 에지 클라우드와 같은 급변하는 네트워크 환경에서 효율적인 리소스 관리 및 스케줄링을 위해 강화학습을 접목시키는 연구가 진행중이다. 이를 통해 네트워크 혼잡을 제어하거나 오버헤드를 줄여 효율적인 네트워크 관리를 가능하게 한다. 네트워크 보안 분야에서는 네트워크에서 발생하는 네트워크 과부하와 같은 이상 징후감지 및 대응 기법에 강화학습을 적용하고 있다. QoS/QoE 분야에서는 강화학습을 통해 동적으로 변하는 네트워크 특성을 고려하여 전반적인 QoS/QoE를 향상시키는 연구가 진행되고 있다.

※ 시사점

SDN/NFV 기술의 발달, 클라우드 시스템 및 데이터센터 활성화, MEC 기반의 서비스 확대에 따라, 네트워크 관리의 복잡도가 증가할 뿐만 아니라 보다 효율적 관리를 지원하는 기술 또한 발달하고 있다. 특히 사람의 개입을 최소화 하면서도 복잡한 네트워크 관리를 지속적으로 할 수 있는 자율형 네트워크 연구는 앞으로 꼭 필요한 상황이라 할 수 있다. 또한 스마트시티, 스마트팩토리, 에너지네트워크 등 다양한 산업 도메인에서도 센서 데이터 관리, 데이터 및 자원 전송 등과 관련되어 강화학습 기반의 자율형 네트워크 기술의 활용이 필요한 실정이다.

자율형 네트워크의 관리 및 제어를 위한 다양한 솔루션들이 연구되고 있으며, 그중에서도 인공지능 및 강화학습 기반의 기술들이 지속적으로 연구되고 있다. 특히 라우팅, 자원관리, 보안, QoS 등 다양한 핵심 기술들에 있어서 지능적인 네트워크 상황 인지 및 자율적 대응을 기반으로 저지연·고신뢰 지원 및 종단간 유연한 연결성 및 품질을 보장할 수 있는 행동 모델을 학습하는 강화학습 활용 방안에 대한 연구는 자율형 네트워크 기술을 선도하기 위해 꼭 필요하다.

이러한 강화학습 기반의 자율형 네트워크 기술 연구를 위해 선결되어야 하는 점은 지능적 네트워크 상황인지를 위한 네트워크 상태 정보 확보이다. 즉, 지속적이며 안정적으로 네트워크 상태정보를 수집할 수 있는 네트워크 텔레메트리(Telemetry) API가 제공되거나 관련 플랫폼이 구축되어야 한다. 네트워크 상태정보 수집 및 관리 플랫폼의 구축은 향후 강화학습 뿐만 아니라 여러 가지 인공지능 기반 네트워크 기술 개발에도 큰 도움이 될 것으로 기대한다.

주 요 내 용

(1) 자율형 네트워크에서 강화학습의 필요성

최근 SDN (Software Defined Networking)[2,3] 기술의 발전에 따라 기존의 하드웨어 중심의 네트워크에서 소프트웨어를 기반으로 하는 네트워크로 변화함으로써 컴퓨팅 자원을 엣지로 분산시키고 동시에 제어기능을 중앙 집중화함으로써 IoT 디바이스들로부터 들어오는 대용량의 데이터를 관리하고 제어할 수 있게 되었다. 또한, 데이터 센터를 운영하는데 네트워크와 스토리지, 서버에 대한 오케스트레이션을 수행하기 위해 SDN을 사용하여 네트워크 상황 및 복잡한 사용자 요청에 유연하게 대응하는 방식으로 운영할 수 있게 되었다. 한편 4차 산업혁명 이후 인공지능 융복합 기술은 일상생활에서 쉽게 접할 수 있는 인공지능 비서부터 효율적인 생산을 위한 스마트 팩토리까지 다양한 산업분야에서 빠르게 접목되어 상당한 부가가치를 창출하고 있다. 이에 따라 네트워크 산업에서도 인공지능 기술을 기반으로 자율형 네트워크 운영관리를 위한 제어 자동화 및 관리 자율화 기술 연구가 활성화되고 있다.

그림1은 인공지능 기반 자율형 네트워크의 개괄적인 구조를 보여준다[1]. 이 자율형 네트워크 구조는 기본적으로 정보 수집, 데이터 전처리 및 제어 명령 전송을 실행한다. 네트워크의 제어 계층에 지능을 도입하기 위해 자가 학습(Self-learning)을 하는 인공지능 모듈을 채택한다. 이러한 자가 학습을 위한 기능을 위해 네트워크의 인공지능 계층에는 자체 인식(self-aware), 자체 적응(self-adaptive) 및 자체 관리(self-managed)와 같은 모듈이 포함되어있다.

자율형 네트워크의 인공지능 모듈은 네트워크 컨트롤러에서 네트워크 상태를 수집하고 이 상태 정보를 사전 처리하여 분석 및 결정한다. 네트워크의 데이터 플레인은 사우스 바운드 인터페이스(SBI)의 원격 측정 기술(Telemetry technique)을 사용하여 트래픽 데이터, 네트워크 토폴로지, 링크 상태 및 리소스 상태를 포함하여 네트워크 상태를 설명하는 원시 데이터를 수집한다. 네트워크의 컨트롤 플레인은 데이터를 수신하고 데이터 전처리를 위한 인식 기능에서 데이터 처리 모듈에 업로드한다.

데이터 집계 모듈은 서로 다른 공급 업체 및 ISP의 이기종 데이터를 데이터 정렬로 처리하여 균일한 데이터를 얻는다.

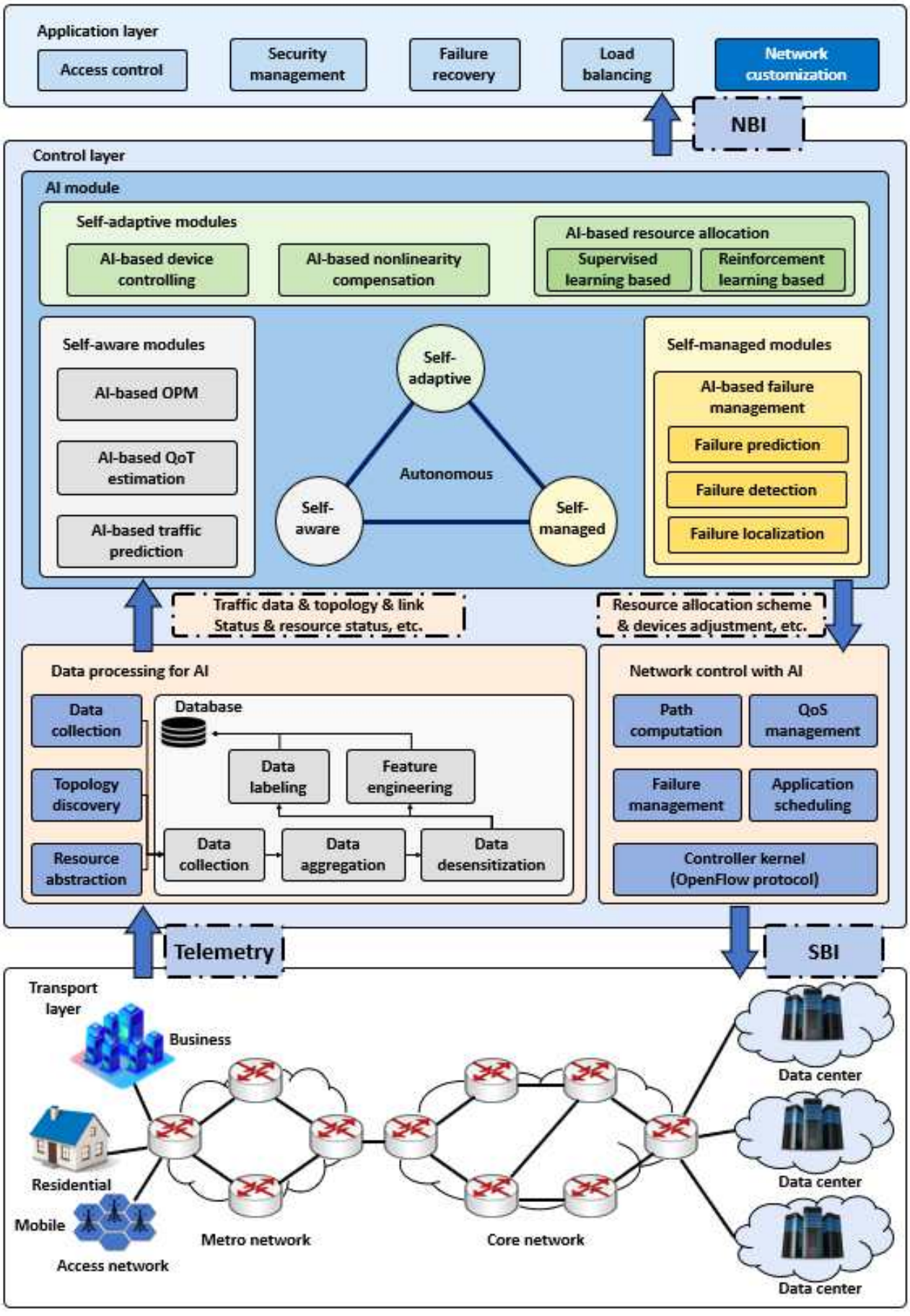


그림 1. 인공지능 기반 자율형 네트워크 구조

데이터 감도 해제 모듈(Data desensitization module)에서는 데이터를 감도 해제하여 개인 정보 보호를 위해 개인 정보를 제외하거나 마스킹한다. 민감하지 않은 데이터(Desensitized data)는 특정 규칙에 따라 데이터 레이블링으로 레이블이 지정되고 특징은 특징 엔지니어링 모듈로 추출된다. 기능 및 해당 레이블은 인공지능 알고리즘 학습을 위한 구조 데이터로 사용된다. 또한 학습을 위해 네트워크 및 인스턴스의 원시 데이터를 저장하기 위해 데이터베이스를 사용해야한다.

네트워크 구조상의 자체 인식 기능은 네트워크 매개 변수를 예측하고 추정하기 위해 인공지능 기반 모듈을 통해 구현된다. 네트워크 상태에 대한 자가 인식에는 인공지능 기반 네트워크 트래픽 예측, 인공지능 기반 성능 모니터링 및 인공지능 기반 전송 품질 추정이 포함된다. 자체 적응형 네트워크 제어 기능은 네트워크 운영이 인공지능을 통해 네트워크 상태의 변화에 적응할 수 있도록 한다. 인공지능 기반 자체 적응에는 네트워크 신호의 인공지능 기반 비선형 보상, 인공지능 기반 제어 및 인공지능 기반 네트워크 리소스 할당이 포함된다. 자체 관리 네트워크 관리 기능은 주로 인공지능 기반 오류 예측, 오류 감지 및 오류 위치를 포함한 자동 오류 관리에 중점을 둔다. 특히 장애 관리 작업에는 장애 예측, 장애 감지 및 장애 현지화가 포함된다. 인공지능 기반 장애 예측 모듈은 인식 기능에서 네트워크 신호, 링크 및 장치에 대한 정보를 수신하고 인공지능 모델을 사용하여 네트워크에 장애가 있는지 여부를 추가로 예측한다. 인공지능 기반 장애 감지에서 분류 모델은 이미 발생한 장애 등급을 식별하는데 사용된다. 인공지능 기반 장애 위치 파악은 현재 네트워크에 많은 수의 경보가 나타나고 실제 장애가 명시적이지 않은 문제에 직면해 있다. 인공지능 기법은 경보 정보로 실제 실패를 지역화하는데 사용된다. 예측된 오류, 오류 유형 및 실제 오류 지역화가 컨트롤러 기능의 오류 관리 모듈에 출력된다.

이러한 자율형 네트워크 기술은 기존의 네트워크 운영자가 직접 수행하던 네트워크 망의 구성, 복구, 최적화, 보안 등의 기능을 네트워크 인프라 자체적으로 자가 관리를 수행하는 기술로 네트워크의 망의 관리 주체를 지능화하여 네트워크 제어 및 관리를 자동화시킴으로써 문제 발생 감소, 비용 절감, 네트워크 탄력성 향상 등과 같은 문제를 해결할 수 있다. 초기 자율형 네트워킹 기술은 네트워크 운영에 관하여 지식, 규칙, 정책 등을 정형화함으로써 제어 및 관리를 위한 자가 관리 기능 구현에 초점을 맞춘다. 하지만, 이러한 기술들은 동적으로 변경되는 리소스 크기 및 배치와 같은 요소들을 고려하지 않아, 네트워크 환경이 변경될 때마다 네트워크 정책을 재구성해야만 했다. 이

러한 다양하고 동적인 네트워크 환경을 고려하기 위해, 적응 최적화 메커니즘인 강화학습을 사용하여 네트워크 환경 안에서 정의된 현재 상태를 인식하여 선택 가능한 행동 중 보상을 최대화 시켜 최적의 네트워크 정책을 구성에 대한 필요성이 두드러지고 있다.

(2) 강화학습 연구동향

가. 개요

인공지능은 몇 차례의 과도기를 지난 최근에서야 주목을 받기 시작했으며 GPU의 병렬처리와 같은 하드웨어의 발달로 빠른 속도로 성장하고 있다. 인공지능은 머신러닝을 기반으로 성장하고 있으며 머신러닝은 실제 사례를 통한 데이터로부터 배운다는 개념을 바탕을 두고 있다. 머신러닝은 그림 2와 같이 학습 방식에 따라 지도학습, 비지도학습, 강화학습으로 분류할 수 있다. 최근 시스템 최적화 문제를 풀기 위해 인공지능 알고리즘 중 하나인 강화학습에 대한 많은 연구 및 개발이 이루어지고 있다.

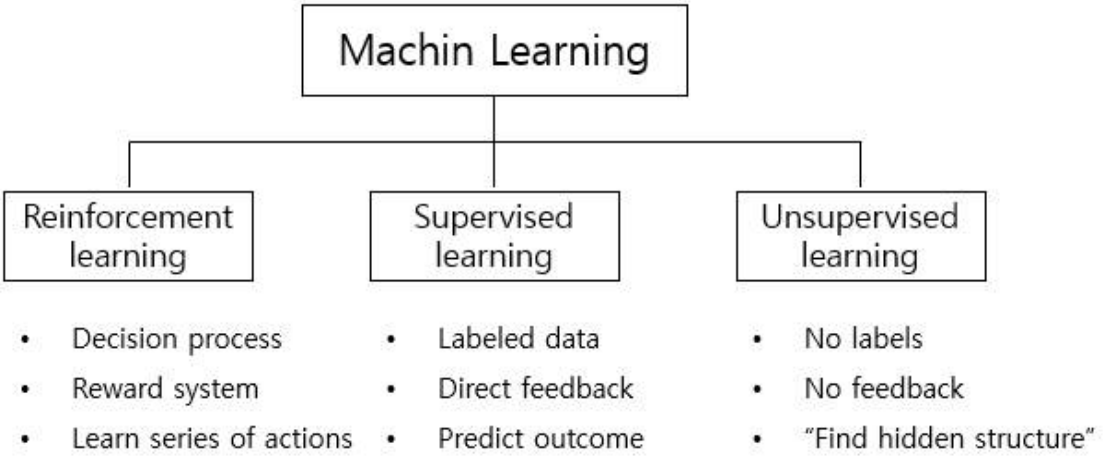


그림 2. 머신러닝 종류 분류

강화학습은 행동심리학에서 영감을 받아 어떤 환경에서 정의된 에이전트가 현재의 상태를 인식하여 선택 가능한 행동들 중 보상을 최대화하는 행동 또는 행동 순서를 선택하는 기법이다. 그림 3은 강화학습의 간단한 구조를 보여준다. 에이전트(Agent)는 환경(Environment)의 한 상태(State)에 대해서 행동(Action)을 취하고 이를 통해 얻은 보상(Reward)를 통해 목표를 달성하는 기법이다. 에이전트는 환경에서 얻은 보상에 따라 행동을 취하는데 이때 에이전트가 행동을 선택하는 방법을 정책(Policy)라고 하며 최종적으로 높은 보상을 받기 위해 반복적인 학습으로 진행된다. 따라서 지도학습 및 비지도 학습과는 다르게 유동적인 상황에 따른 의사 결정을 통해 경험해보지 못

한 상황에 대한 최적의 의사결정을 하는데 유리한 장점이 있다. 강화학습을 위한 학습 방식으로 고전적인 방식부터 최근 딥러닝 기법을 접목한 방식까지 다양하게 있으며, 이 중 가장 기본적이고 유망한 알고리즘 3가지 (Q-Learning, Deep Q-Network, Deep Deterministic Policy Gradient) 에 대해서 설명한다.

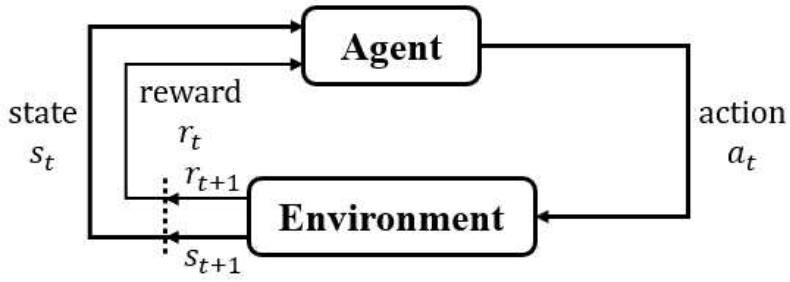


그림 3. 강화학습 아키텍처

나. Q-Learning

Q-learning[5]은 MDP(Markov Decision Process)[4]에서 최적의 정책을 찾기 위한 방법 중 하나이자 강화학습의 가장 기본적인 알고리즘이다. Q-learning 아키텍처는 기본적으로 강화학습 아키텍처를 따른다. Q-learning은 MDP를 기반으로 하고 있으며 벨만 방정식(Bellman Equation)을 이용한다. 벨만 방정식은 재귀적 함수를 통해 다음 상태에 대한 가치 함수(Value function)을 구하는 과정을 수식(1)으로 표현한다. 즉, 현재 가치와 미래 가치의 합이 최대가 되는 행동을 선택하여 가치(value) 값을 구하고 이를 최대로 하는 행동을 선택하는 것이다.

$$v(s) = \max_a(R(s,a) + \gamma v(s')) = \max_a(R(s,a) + \gamma \sum_{s'} P(s,a,s')v(s')) \quad (1)$$

수식(1)에서 구한 가치 함수에서 최댓값을 구하고자 하는 R(s,a) + γv(s') 를 Q(s,a)라고 치환하며 이를 Q 값이라고 한다. 따라서 가치 함수는 수식(2)과 같이 표현할 수 있으며 Q 값은 수식(3)과 같이 재귀적으로 할당된다.

$$v(s) = \max_a(\hat{Q}(s,a)) \quad (2)$$

$$\hat{Q}(s,a) \leftarrow R(s,a) + \gamma \max_a \hat{Q}(s',a) \quad (3)$$

Q-learning의 목표는 MDP와 마찬가지로 Q value를 최대로 하는 행동을 찾는 것이다. 즉, 각 상태에 따라 최적의 행동을 취하는 Q 행렬(Q matrix)을 만들어야 한다. Q 행렬은 수식(4)의 과정 반복을 통해 반복적으로 학습된다.

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha(R(s,a) + \gamma \max_a Q(s',a)) \quad (4)$$

여기서 α는 학습율(learning rate)로 한 번의 step에 얼마나 크게 반영을 할 것인가를 정하는 것이다.

이를 바탕으로 Q-learning을 통한 학습과정은 알고리즘 1과 같이 기술되며, 크게는 초기화 단계와 로직 단계로 진행된다.

Algorithm 1. Q-learning

Initialize $Q(s,a)$, $\forall_s \in S$, $a \in A(s)$, arbitrarily, and $Q(terminal-state, \cdot) = 0$
Repeat (for each episode):
  Initialize S
  Repeat (for each step of episode):
    Choose A from S using policy derived from Q (e.g., ϵ-greedy)
    Take action A, observe R, S'
    $Q(S,A) \leftarrow Q(S,A) + \alpha[R + \gamma max_a Q(S',a) - Q(S,A)]$
    $S \leftarrow S'$
  until S is terminal

초기화 단계 : 0~1의 ϒ과 α를 설정하고 보상 행렬(Reward matrix)을 환경변수로 입력한다. 이 보상 행렬은 각 상태에 대해 행동을 취했을 때의 보상을 행렬로 정리한 것이며 간선(Edge)이 없는 곳은 Null 값을 넣는다. ϒ값은 0에 가까울수록 멀리 있는 상태를 보지 않게 되고 1에 가까울수록 원시안적인 행동을 하게 된다. 마지막으로 학습시킬 Q 행렬을 0으로 초기화한다.

로직 단계 : 이 단계는 반복적인 학습을 통해 Q 행렬을 학습시키는 과정이며 에피소드(episode) 블록에 해당된다. 임의의 상태를 선택한 후 가능한 행동 중 하나를 선택하는데 보편적으로 사용하는 전략은 ε-greedy이다. ε-greedy는 모든 상태를 모두 한 번씩 경험해 본 후 가장 좋은 행동을 선택하는 것을 뜻한다. ε-greedy는 적절한 확률 ε를 정하고 이 확률만큼은 가장 좋았던 행동을 선택하고 나머지 확률만큼은 임의의 행동을 선택한다. 이 행동을 통해 다음 상태를 구할 수 있으며 Q 행렬을 학습시킨다.

다. Deep Q-Learning (DQL)

한동안 강화학습에서 Q-learning은 좋은 성과를 냈다. 하지만 기술이 발전하고 요구가 늘어나면서 상태의 수가 늘어나거나 차원이 높아졌으며 이 데이터를 기존의 방식으로 직접 다루는 것이 불가능에 가까워졌다. 현실 세계의 문제를 풀기 위해 고차원 데이터를 다루는 것이 반드시 필요했고 이를 해결하기 위해 2015년에 DeepMind에서 DQN[6]을 소개하였다. 그림 4는 DQN의 아키텍처를 보여준다.

DQN을 간단히 설명하면, 다루기 힘들었던 고차원의 상태를 학습에 이용하기 전에

CNN(Convolution Neural Network)을 이용하여 차원을 낮춘 후 이를 이용하여 학습을 진행하는 것이다. DQN은 기존의 강화학습과는 다르게 입력인 상태가 고려해야하는 픽셀의 크기가 상당히 크기 때문에 Q 함수를 실행시키기 전에 CNN을 이용하여 상태의 차원을 낮춘 후 Q 함수를 통해 강화학습을 진행한다.

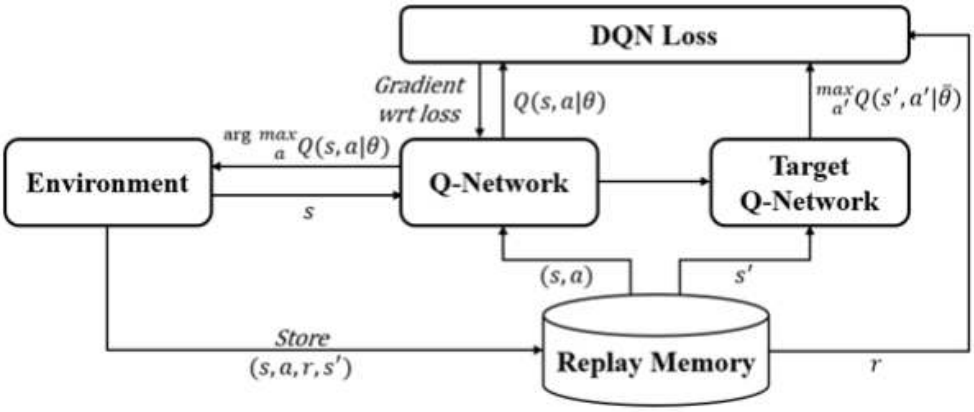


그림 4. Deep Q-learning 아키텍처

강화학습에 딥러닝 기술을 접목시키기 위해서 필요한 고려해야할 사항들이 있다. 첫 번째로, 기존의 딥러닝 기술들은 사전에 라벨링된 많은 양의 데이터를 갖고 있었다. 하지만 강화학습은 노이즈를 기반으로 보상을 매기고 이를 통해 학습이 진행된다. 이는 학습을 진행하는데 상당히 오랜 시간이 걸린다는 것을 의미한다. 그리고 두 번째로, 기존의 딥러닝은 데이터 샘플들을 독립 동일 분포(Independent and identically distribution)로 가정하여 학습시켰다. 하지만 강화학습의 경우 현재의 상태가 다음의 상태에 영향을 미치기 때문에 독립적이지 않다. 즉, 데이터간의 상관관계(correlation)가 학습에 영향을 줄 수 있다.

이러한 문제를 해결하기 위해 DQN은 경험 재현(Experience replay)[7] 방법을 도입하였다. 기존 알고리즘들은 환경과 상호작용하며 얻은 샘플들, 즉$s_t,\ a_t,\ r_t,\ s_{t+1},\ a_{t+1}$을 통해 파라미터들을 업데이트 하였다. 하지만 이와 같은 방법은 샘플들에 대한 의존성이 커서 정책이 수렴하지 못하는 문제가 있다. 이를 위해 경험 재현은 각 스텝별로 얻은 샘플들을 튜플 형태로 데이터 셋에 저장해두고 mini-batch 방식으로 업데이트한다. 따라서 샘플 튜플과 데이터 셋은 $e_t = (s_t,\ a_t,\ r_t,\ s_{t+1}),\ D = e_1,\ e_2,\ \cdots,e_N$ 와 같다. 이러한 mini-batch 방식의 업데이트를 이용하면 경험 기반의 각 스텝은 잠재적으로 많은 가중치 업데이트가 진행되므로 기존의 가중치 업데이트를 한 번만 사용하는 방식보다 데이터를 더 효율적으로 사용할 수 있다. 또한, 연속된 샘플로 학습하는 것은 샘플간의 상관관계 때문에 비효율적이지만 임의 샘플링을 적용함으로써 상관관계가 적고

고르게 분포된 데이터로 학습할 수 있으므로 목표에 수렴이 잘 된다.

학습 진행 과정은 정책에 따라 행동을 선택하고 다음 학습 샘플은 이 행동에 따라 결정된다. 예를 들어 정책이 '왼쪽으로 이동'이라면 다음 학습 샘플은 왼쪽에 있는 상태에서의 샘플들이 주로 나올 것이라고 예측할 수 있다. 이는 불필요한 피드백 과정을 줄일 수 있다.

Q-learning에 딥러닝과 경험 재현 기법을 접목시킨 DQN의 알고리즘은 에피소드 블록에 있는 상태를 CNN을 이용하여 크기를 줄인 과정이 되며 스텝 블록에서 경험 재현과 Q learning을 기반으로 알고리즘이 작동한다.

라. Deep Deterministic Policy Gradient (DDPG)

Q learning과 DQN은 가치 기반 강화학습이다. 가치 기반 강화학습은 Q 함수에 초점을 맞추어 Q 값을 구하고 그것을 토대로 정책을 선택하는 방식이다. 가치를 기반으로 정책을 선택하기 때문에 계산이 쉽지만 하나의 정책만 선택이 가능하고 가치가 조금만 달라져도 정책의 변화가 크다. 이와 다른 방식의 강화학습으로 정책 기반 강화학습이 있다. 이는 정책 자체를 간략화하여 함수 근사기(Function approximator)에서 나오는 것이 가치 함수가 아닌 정책 자체이다. 즉, 가치 기반 강화학습이 각 행동에 대해 값을 얻어 이를 기반으로 정책을 선택했다면 정책 기반 강화학습은 각 행동에 따른 정책 분포를 구하여 이를 기반으로 선택할 수 있다. 때문에 행동의 제약을 덜 받는다.

DDPG는 DPG(Deterministic Policy Gradient)에 심층신경망을 접목시킨 알고리즘이다. DPG는 2014년 DeepMind에서 제안한 기법으로 앞에서 설명한 가치 기반 강화학습과 정책 기반 강화학습이 함께 사용되는 행위자 비판(Actor-Critic) 구조로 이루어져 있다. 즉, 환경 사이에 정책을 결정하는 행위자(Actor)와 행위자가 선택한 정책을 평가하는 비판(Critic)이 존재한다. 행위자는 정책 기반으로 정책을 결정하며 비판은 결정된 정책으로부터 얻은 보상을 이용하여 가치 기반으로 정책을 평가한다. 이러한 구조를 행위자 비판 구조라 하며 DPG와 DDPG의 기초 구성이다.

DPG의 비판에서는 Q 함수를 다루며 행위자에서 정책 기반 방식으로 정책을 결정하는 방법이 필요하다. 앞에서 설명했듯이 정책은 상태에 대한 분포를 기반으로 결정해야 하며 필요한 분포들은 다음과 같다.

$p_0(s)$: 상태에 대한 초기 분포

$p^{\mu}(s \rightarrow s', k)$: 상태 s에서 정책에 따라서 k 스텝이후 상태 s'에 도달하기까지의 방문

확률 밀도(Visitation probability density)

$p^{\mu}(s') = \int \sum_{sk=1}^{\infty} \gamma^{k-1} p^{\mu}_{0(s)p}(s \rightarrow s', k) ds$: 할인된 상태 분포(Discounted state distribution)

이를 이용하여 우리가 최적화해야 할 목표 함수(Objective function)는 다음과 같다.

$$J(\theta) = \int_{s} p^{\mu}(s) Q(s, \mu_{\theta}(s)) ds$$

최적의 목표 함수를 위해 DPG는 기울기(Gradient)를 이용한다. 우선 행동에 대해 Q의 기울기를 구하고, 정책 파라미터에 대해 결정적 정책 함수(Deterministic policy function)의 기울기를 구한다. 여기서 정책의 분포에 따라 확률론적으로 결정한다. 또한 DPG는 비 정책적 접근(Off-policy approach)으로 작동되는데 이는 학습 궤적(Training trajectory)가 확률론적 정책(Stochastic policy)에 의해 생성되고, 이로 인해 상태 분포가 할인된 상태 배포를 따르면 된다.

DDPG (Deep Deterministic Policy Gradient)는 2016년에 DeepMind에서 제안됐으며 DPG와 DQN을 결합시킨 모델이다[6]. 그림 5는 DDPG의 아키텍처를 보여준다. 기존의 DPG는 연속적인 행동의 고려가 가능했으며, 기존의 DQN은 연속적인 상태의 고려가 가능했다. DDPG는 이러한 장점을 결합한 모델이라 할 수 있다. 연속적인 행동이 가능한 DPG 기반의 행위자-비판 구조에서 비판 네트워크에 심층신경망을 접목시킴으로써 연속적인 상태 고려가 가능하도록 하였다.

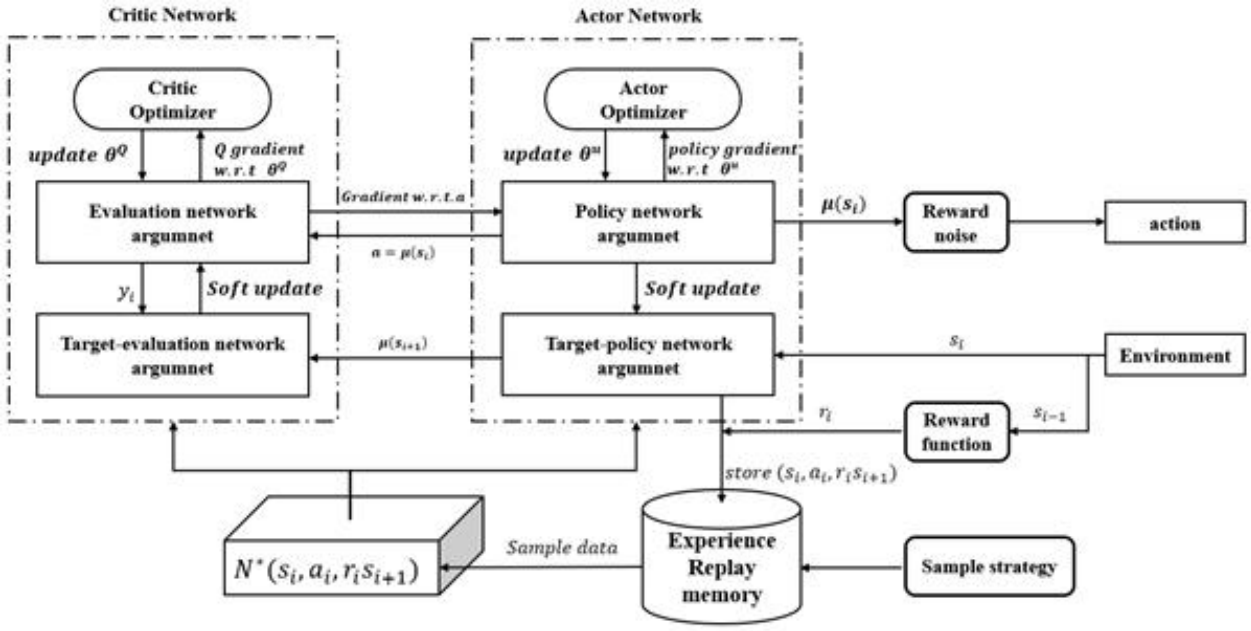


그림 5. Deep Deterministic Policy Gradient의 아키텍처

(3) 네트워크 기술에 강화학습 활용 동향

SDN/NFV 기술이 발달하면서 여러 가지 지능형 기법을 네트워크 기술에 적용하려는 시도가 계속되고 있다. 자율형 네트워크를 구현하기 위해 네트워크 관리에 다양한 방법으로 강화학습을 적용하기 위한 연구가 끊임없이 진행되고 있으며 Routing, Resource Management, Security, QoS/QoE 와 같은 여러 분야에서 활용되고 있다.

가. Routing

네트워크 안에서 통신 데이터를 보낼 최적의 경로를 선택하는 과정인 라우팅은 네트워크 관리를 위한 중요한 요소 중 하나이다. 특히 종단간의 데이터 전송을 위한 최적의 경로는 최단 거리가 될 수 있으며 가장 빠른 시간에 전달하는 것을 고려하여 계산될 수 있다. 이러한 라우팅 과정은 보통 다양한 네트워크 목적지에 대한 기록을 관리하는 라우팅 테이블을 기초로 수행되며 네트워크 상태, 서비스 요청 등을 지속적으로 고려하여 강화학습 방법을 활용하여 라우팅 테이블을 학습하는 형태의 연구가 주로 진행되고 있다.

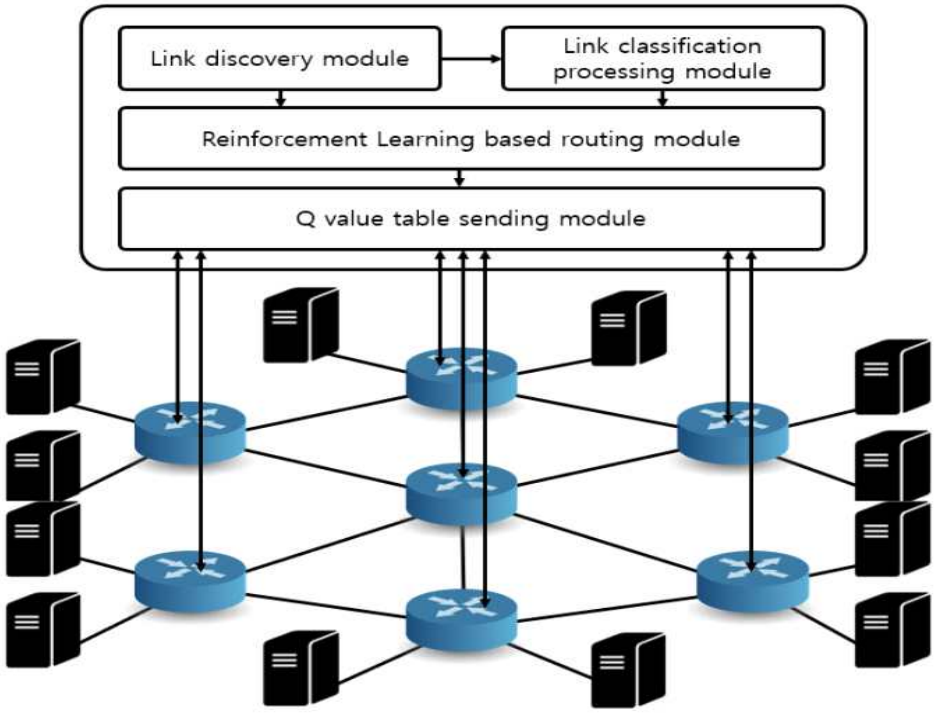


그림 6. SDN 환경에서 서비스 분류를 고려하는 Q-Learning 기반 라우팅 기법

그림6은 Q-Learning 알고리즘과 서비스 분류를 고려한 비즈니스 흐름 속성을 기반으로 최적의 경로를 설정하는 기법을 제안하였다[8]. 제안된 기법은 4개의 모듈

(링크 검색, 링크 분류, 집중 학습 및 Q-값 테이블 전송)을 구축하여 제어 계층에 배포된 여러 속성 데이터 스트림에 서로 다른 경로를 할당한다. 링크 검색 모듈과 링크 분류 모듈을 사용하여 Q러닝으로 학습하기 위해 필요한 링크 지연, 가능한 대역폭, 패킷 손실률 및 대역폭 사용량과 같은 4가지 네트워크 요소 데이터를 얻는다. 또한 각 요소의 부정적, 긍정적 상관관계를 고려하여 4가지 범주(인터랙티브 서비스, 온라인 미디어 서비스, 인터랙티브 서비스, 데이터 레이어 서비스)로 분류된 보상 및 패킷 속성을 정의한다. 패킷의 속성에 따라 각 요소의 가중치를 얻은 다음 요소 값과 가중치를 사용하여 보상을 얻을 수 있다. 이를 바탕으로 학습이 수행될 때마다 Q테이블의 값을 업데이트한다. 학습이 완료된 후 Q테이블은 OpenFlow 스위치에 설치되고 이를 통해 패킷의 서비스 속성에 따라 알맞게 선택된 라우팅 테이블을 통해 쉽게 라우팅 할 수 있다. 또한 로컬 혼잡을 피하기 위해 네트워크 자원 소비 비율을 합리적으로 할당하고 사용자의 QoS 요구를 최대한 충족할 수 있다. 다만, 이 방식은 정적 토폴로지를 기반으로 학습하는 반면, 동적 네트워크 상황을 고려하지 못하는 한계를 가진다.

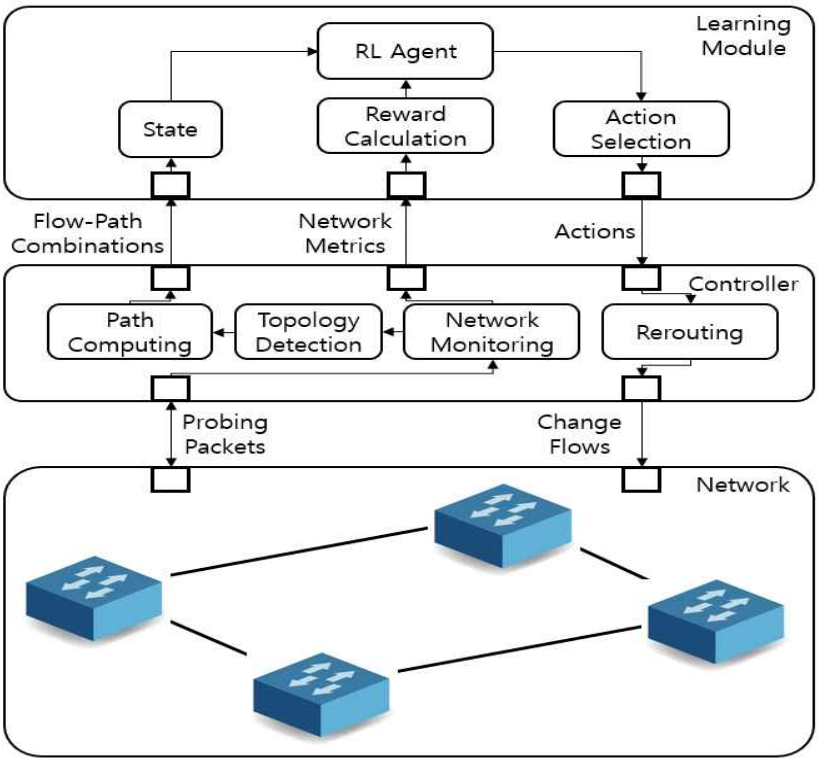


그림 7. QR-SDN 아키텍처

이와 유사한 방법으로 그림 7과 같이 Q-Learning을 사용하여 네트워크 플로우 기반으로 경로를 할당하는 QR-SDN기법도 연구되었다[9]. QR-SDN기법은 종단간 여러 개의 플로우 각각에 대해서 서로 다른 경로를 할당할 수 있는 라우팅 테이블을

Q-Learning을 사용하여 학습하도록 한다. QR-SDN은 초저지연 통신을 목표로 하며 플로우 각각의 지연을 보상으로 설계하여 Q-Learning을 수행하도록 한다. 플로우 단위로 Q-Learning을 수행할 경우 상태-행위 공간이 매우 커질 수 있으며 이러한 문제 해결을 위해서는 심층신경망을 활용하는 심층강화학습(DRL : Deep Reinforcement Learning)의 사용이 필요할 수 있다. 하지만 QR-SDN에서는 심층강화학습을 사용하는 대신, 분산 SDN 컨트롤러를 사용하여 상태-행위 공간을 여러 개의 컨트롤러에 분산시키는 방법을 통해 학습시간의 확장성 문제를 해결하고자 하였다.

이러한 Q-Learning기반의 방법에서의 상태-행위 공간 및 학습시간의 확장성 문제의 해결을 위해 심층강화학습(DRL : Deep Reinforcement Learning)을 사용하는 라우팅 기법도 연구되고 있으며, 대표적인 연구로 DRL-TE[18], TIDE[19], DROM[11] 등이 있다. DRL-TE는 사용자의 트래픽 이용형태를 심층강화학습으로 분석하는 방법으로 라우팅 테이블을 학습한다. TIDE는 네트워크 링크 사용정도를 포함하는 네트워크 상태의 시계열정보를 이용하는 심층강화학습 기법이며, DROM은 네트워크 플로우의 소스-목적지 매트릭스를 심층강화학습으로 학습하는 기법이다.

통신망의 급속한 성장으로 인한 네트워크 트래픽 흐름의 시공간적 분포의 강도, 다양성 및 복잡성이 크게 증가하는 상황에 대응하기 위해 네트워크 플로우 시계열 데이터 기반의 심층강화학습을 사용하여 QoS를 만족하는 라우팅 최적화 기법인 TIDE가 연구되었다[19]. TIDE는 수집, 결정, 조정 루프로 구성되어 있다. 수집 단계는 라우팅 학습을 위한 네트워크 상태 및 성능에 대한 정보 수집하고, 결정단계는 라우팅 학습 알고리즘을 포함하며, 조정단계는 학습된 라우팅 정책을 각 스위치의 흐름 테이블로 전달하는 작업을 포함한다.

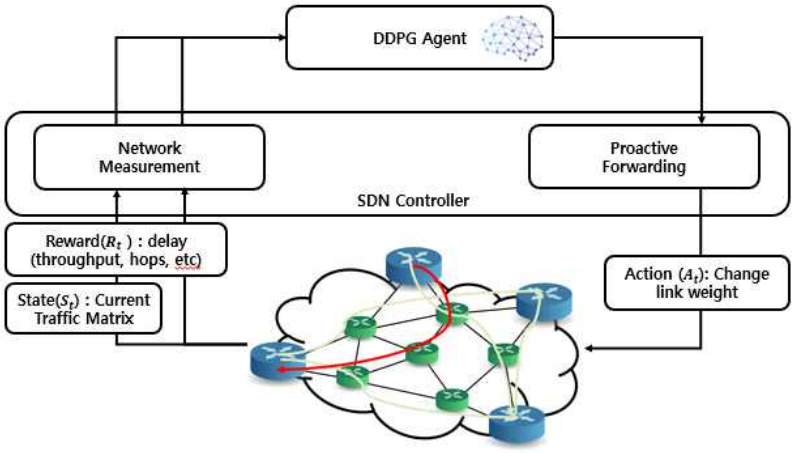


그림 8. DROM의 개략적 구조

DROM은 DDPG(Deep Deterministic Policy Gradient)를 사용하여 SDN에서 라우팅을 최적화하는 머신 러닝 기반 SDN 프레임워크이다[11]. 네트워크 트래픽이 기하급수적으로 증가함에 따라 서비스 품질을 유지하면서 SDN 라우팅 프로세스를 간소화해야 함에 따라, DROM에서는 그림 8과 같이 DQN과 DPG를 통합한 DDPG기반의 지능형 결정 모듈을 제어 플레인에 추가하였다. DDPG는 신경망을 사용하여 전략적 기능과 Q 기능을 통합하고 이산 제어 동작의 실용적이고 안정적인 모델을 제공한다. DROM 라우팅 최적화 방법은 개인화 된 네트워크 인텔리전스 제어 및 관리를 실시간으로 처리할 수 있도록 설계 되었다. DROM 에이전트는 네트워크의 링크 가중치를 변경하여 네트워크 로드 및 작업의 트래픽 매트릭스를 변경하여 데이터 플로우의 경로를 변경할 수 있다. 에이전트의 보상은 지연, 지연과 같은 사용자 지정 성능 매개 변수를 자동으로 최적화하는 이점이 있다. 전달 경로 길이, 네트워크 운영 및 유지 관리 전략과 관련된 처리량의 관리를 통해 실시간 네트워크 제어를 가능하게 한다.

나. Resource Management

자율형 네트워크 구현을 위해 필요한 기술 중, Routing 다음으로 중요한 기술은 Resource Management이다. 네트워크 자원은 다양하게 고려할 수 있다. 네트워크 링크 대역폭과 같이 직접적인 자원뿐만 아니라, SDN에서 동적 라우팅 테이블을 관리하는 스위치 메모리, 그리고 NFV 구현을 위한 컴퓨팅 자원과 같은 다양한 형태의 네트워크 자원을 고려할 수 있다. 특히 클라우드 시스템의 도입과 데이터센터 규모의 확대 및 MEC(Multi-access Edge Computing) 활성화에 따라 컴퓨팅 자원관리의 중요성이 점점 높아지고 있다.

네트워크 자원관리의 경우 NFV 구현을 위한 서비스 체인 분야에서 활발하게 연구되고 있으며 스마트 팩토리 및 스마트 빌딩과 같은 스마트 환경을 지원하는 네트워크 자원관리에서도 활발히 연구되고 있다. 그림 9에서는 스마트 팩토리를 위한 인공지능 기반 에지컴퓨팅 및 인공지능 기반 클라우드 구조를 나타낸다. 스마트 팩토리 관리를 위한 무선자원 및 에지컴퓨팅 자원 정보에 기반하여 인공지능 기반 클라우드에서 네트워크 자원을 관리하는 구조를 가진다. 또한, 그림 10에서는 스마트 시티에서 다양한 센서 및 에너지 발전 소자들의 관리를 위한 DRL기반의 에너지 클라우드 관리 구조를 나타낸다. 이 구조에서는 각 에지와 중앙 클라우드에서 각기 DRL 네트워크 자원관리를 하는 것을 알 수 있다.

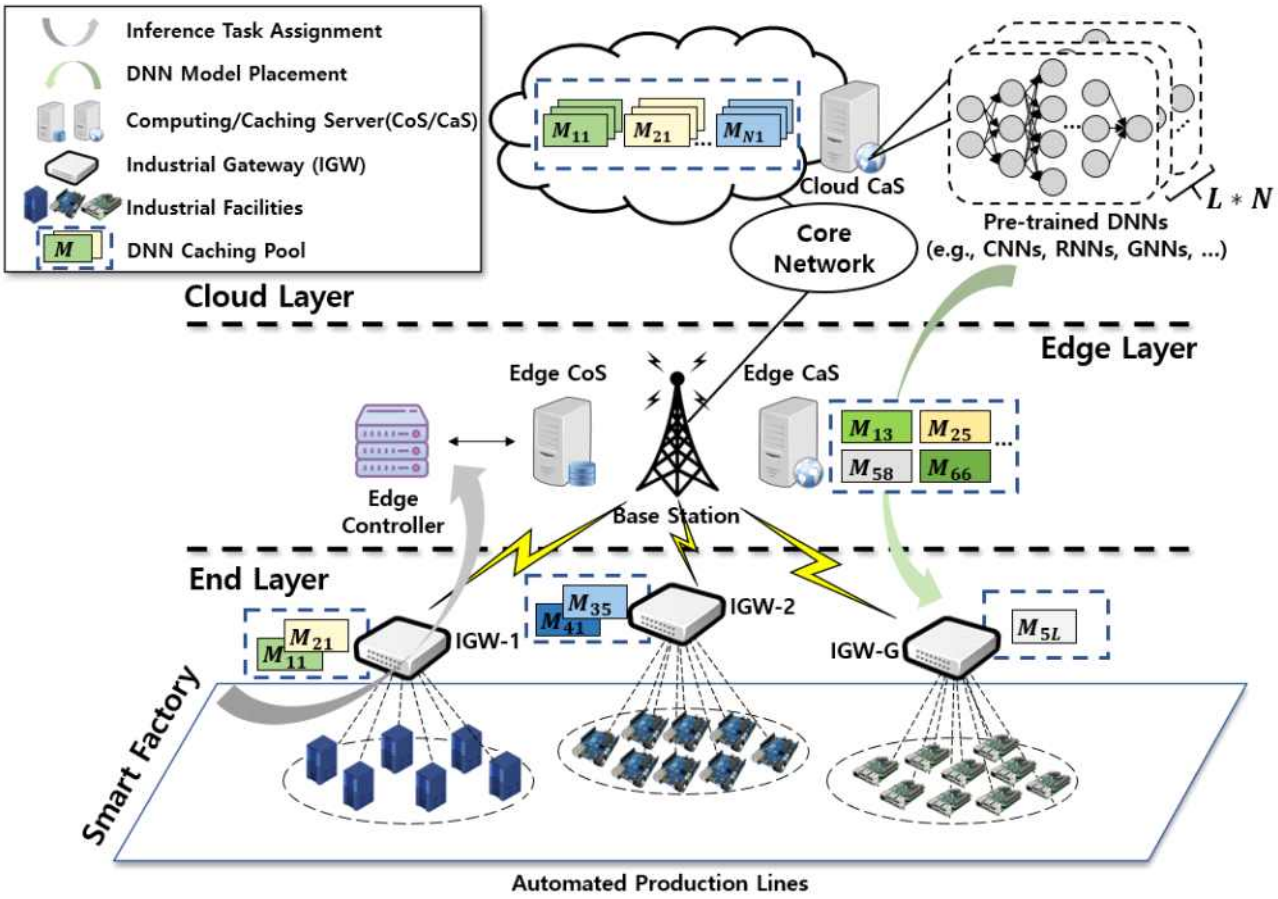


그림 9. 스마트 팩토리를 위한 인공지능 기반 에지 클라우드 구조

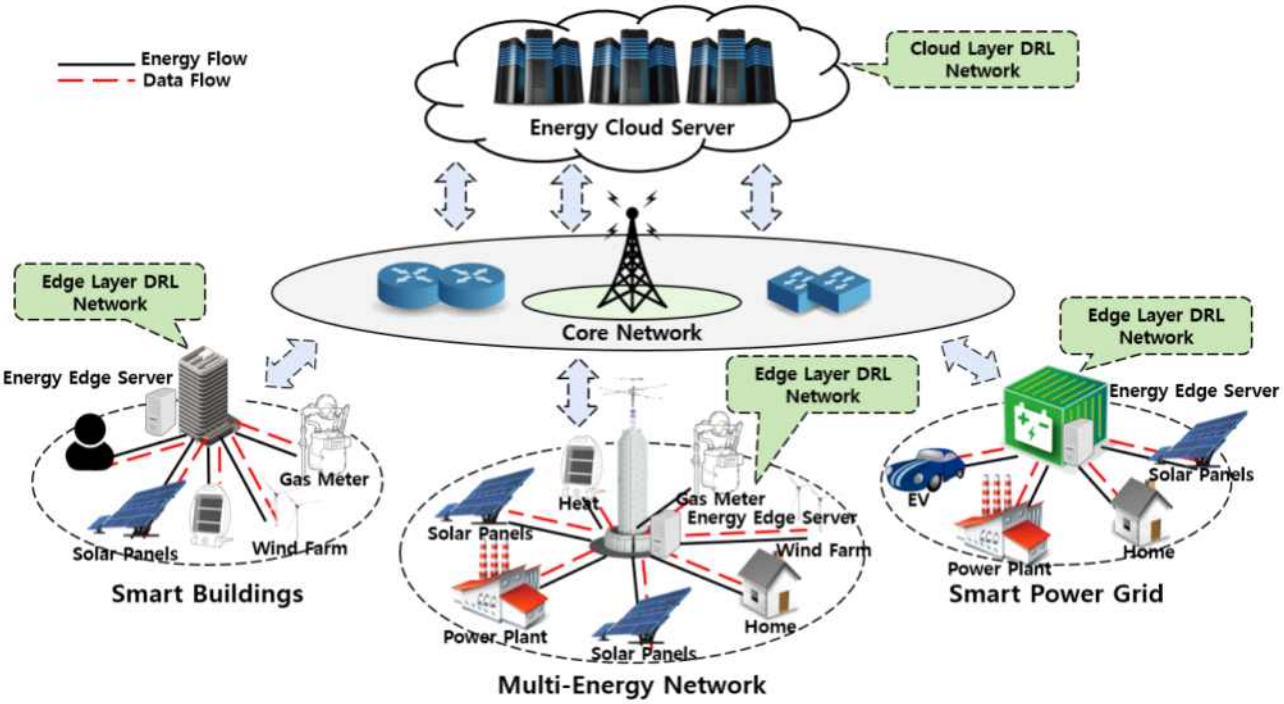


그림 10. 스마트 시티를 위한 강화학습기반 에너지 네트워크 관리 아키텍처

이러한 강화학습 기반 클라우드 자원관리 기술의 연구와 관련되어, SDN 데이터 센터에 강화학습을 접목시켜 네트워크의 혼잡제어를 위한 기술이 제안되었다[12]. 클라우드 컴퓨팅과 빅데이터의 발전으로 인해 데이터센터의 트래픽이 심각하게 증가하고 있으며, 데이터의 센터의 대역폭은 요구된 대역폭에 충족되는 것이 어려워지고 네트워크의 복잡성 위험에 직면하고 있다. 이를 위해 데이터센터에서의 플로우 할당 비율을 강화학습으로 최적화함으로써 문제를 해결한다. 학습을 위해 5개의 원소(Flow, Link State, Action, Reward, Q-matrix)로 구성된 튜플이 사용된다. Flow는 스위치들 사이에 연결된 link들의 flow로 구성된 벡터이며 Link State는 각 link들에 점유된 대역폭 크기로 구성된 벡터이다. 또한 Action은 각 link에 대한 변경 가능한 대역폭이며 Action을 취한 후의 점유된 대역폭과 네트워크 혼잡의 threshold의 차를 이용하여 Reward를 구할 수 있다. 이를 통해 학습이 진행되고 학습이 끝나면 이를 기반으로 Q-matrix를 만들어 사용한다. Q-matrix는 각 State간의 Q-value들로 구성된 행렬이며 이를 바탕으로 각 flow의 할당 비율을 결정한다. 제안되는 기술에서는 Q-Learning 및 DQN을 활용하는 학습 알고리즘을 사용한다.

또한 클라우드 네트워킹 오버헤드를 줄이기 위해, 강화학습을 통한 네트워크 자원관리 최적화 기술도 연구되었다[13]. SDN 컨트롤러는 OpenFlow를 이용하여 flow 테이블에 있는 포워딩 규칙이 수정될 때마다 스위치에게 컨트롤 메시지를 보내게 되는데, 이 정보 교환 프로세스가 대규모 서버 풀을 가지는 클라우드 데이터센터에서는 네트워킹 리소스를 소비하고 패킷 전달 프로세스를 지연하는 오버 헤드로 간주된다. 제안된 기술은 수명이 긴 Elephant 패킷과 수명이 짧은 Mice 패킷으로 플로우의 패킷을 구분한다. Elephant 패킷은 전체 패킷의 10% 미만의 적은 빈도수를 나타내지만 이 플로우에 운반되는 페이로드는 전체 트래픽 볼륨의 80%를 차지하며 Mice 패킷은 트래픽 볼륨 비중은 낮지만 많은 빈도수를 갖고 있기 때문에 두 패킷 사이의 공존이 중요하다. 이러한 데이터센터 네트워크 플로우의 특성을 고려하여 강화학습의 상태는 플로우의 빈도수와 플로우의 스위치 메모리 내에서의 체류시간으로 구성하고, 행동은 플로우 빈도수의 변동으로 설정한다. 이렇게 설정된 상태와 행동 그리고 보상(오버헤드 최적화)을 이용하여 네트워크 플로우 할당을 하기 위한 강화학습을 진행한다. 학습된 네트워크 플로우 할당 모듈은 네트워크 링크 자원뿐만 아니라 컴퓨팅 메모리 자원(스위치 라우팅 메모리 및 스위치 데이터 메모리)의 오버헤드를 최적화 할 수 있도록 돕는다.

다. Security

네트워크의 복잡성이 증가함에 따라 보안 위협 또한 함께 증가하게 된다. SDN/NFV 환경의 발달에 따라 네트워크의 복잡한 상태 데이터 관리를 자동화 하는 기술이 발달하고 있으며, 이러한 자동화 기술을 바탕으로 네트워크상의 이상행동을 식별하는 기술 또한 발달하고 있다. 이상행동 탐지 기술은 네트워크 침입탐지 및 악성코드 탐지 등으로 확대가 가능하며 머신러닝 및 인공지능 기술의 도입을 통해 점진적으로 그 기술이 고도화 되고 있다. 이처럼 네트워크 보안에서도 인공지능 기술의 도입은 적극적으로 이루어지고 있으며 자율형 네트워크와 같이 인간의 개입이 최소화되는 상황에서도 지속적으로 네트워크 보안 이슈를 해결하기 위한 강화학습 기반의 보안 기술이 연구되고 있다. 강화학습이 활용되는 네트워크 보안 분야로는 호스트 기반 IDS(Intrusion Detection System : 침입탐지시스템), 네트워크 기반 IDS, 이상 징후 기반 탐지 기술, 시그니처 기반 탐지 기술, 스푸핑 탐지 기술, 악성코드 탐지 기술 등이 있다.

침임 탐지 시스템의 구현을 위한 가장 기초적인 기술은 이상 징후 탐지 기술이다. 최근, 다양한 공격 시나리오에서 최적의 완화 정책을 학습하고 실시간으로 DDoS (Distributed Denial-of-Service) flooding 공격을 완화할 수 있는 심층 강화 학습 기반의 프레임워크가 제안되었다[14]. 네트워크상 공격 트래픽이 양성 트래픽과 혼합될 수 있다는 이유로 이러한 공격을 실시간으로 스마트하고 효과적으로 완화하는 것은 어려운 작업이다. 정보 수집 모듈은 표준 OpenFlow 프로토콜을 기반으로 전체 네트워크의 트래픽 정보 수집을 목표로 설계되었다. DDoS 공격 완화 모듈은 DDoS 공격을 효과적으로 완화하고 네트워크 자원 서버를 사용자가 이용할 수 있도록 하기 위해 설계되었다. DDoS 공격 완화 모듈은 Mitigation 서버와 Mitigation 에이전트의 두 부분으로 구성된다. 서버는 SDN 컨트롤러에서 애플리케이션으로 작동하며, 에이전트는 심층 강화 학습 알고리즘(DDPG)을 사용하여 학습을 수행한다. Actor는 현재 정책을 정의하고 State를 결정적으로 특정 Action에 매핑한다. 이 State는 각 OpenFlow 스위치 및 각 포트의 플로우의 통계 값으로 정의하고, Action은 특정 호스트의 최대 대역폭을 나타내는 벡터 값, Reward은 최적화로 설정한다. Critic 함수 Q(s, a)는 벨만 방정식에 따라 추정한다. Actor는 매개 변수와 관련하여 트래픽 전송이 시작되는 지점으로부터 예상되는 네트워크 트래픽에 대한 Chain Rule을 사용하여 학습한다. 이를 통해 DDoS 공격 완화 모듈은 DDoS 공격이 발생한 경우의 타겟 서버(Victim Server)로의 트래픽의 전송률 또는 대역폭을 제한함으로써 네트워크 공격을 완화한다.

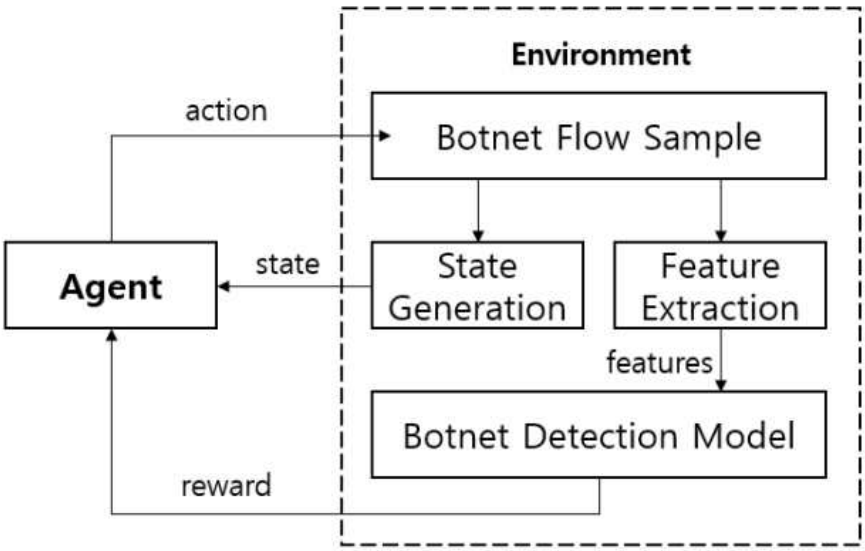


그림 11. DQN 기반 봇넷 트래픽 탐지 프레임워크

NFV 기술과 강화학습을 사용하여 SDN상의 봇넷 트래픽을 검출 및 완화하는 기술도 연구되었다[15]. 제안된 기술은 봇넷 트래픽뿐만 아니라 다양한 이상 징후(공격 및 네트워크 과부하)에 대한 탐지기술로 확장이 가능하다. 제안된 기술이 적용된 이상 징후 탐지 시스템은 NFV 아키텍처의 orchestrator에 상주하는 에이전트에 의존하며, 네트워크 매트릭스로부터 이상 징후를 수집하여 네트워크 프로파일을 구축한다. 네트워크 프로파일은 다양한 공격의 처리를 계층화하여 에이전트가 먼저 우선 순위화한 이상 현상을 제거하는 데 집중하도록 한다. 특정한 현상을 다루는 각 프로파일은 State 표 및 Action표로 구성된다. 각 학습 주기에서 에이전트는 프로필을 선택하고, 사용 가능한 Action을 실행하며, 프로필의 Q-테이블을 업데이트하는 Reward을 받는다. 에이전트는 각 주기마다 서로 영향을 미치는 여러 작업을 피하기 위해 프로필의 한 가지 작업만 실행한다. 그림 11은 이러한 여러 프로파일 중 봇넷 트래픽 탐지를 위한 강화학습 모델의 구성을 나타낸다. 네트워크 프로파일(로드밸런싱, 서버 복제, DoS공격)은 다양한 위협의 처리를 계층화하여 에이전트가 가장 긴급한 이상을 제거하는 데 집중한다. 각 프로파일에는 자체 우선순위가 있을 수 있으므로 세 가지 중 둘 이상의 프로파일에 의해 이상 징후가 감지될 경우 에이전트는 Action을 선택할 때 이들 중 하나를 우선순위로 뽑는다. 이러한 계층화는 위협을 제거하기 위한 조치로 인해 다른 프로파일의 조치로 인한 영향이 취소되는 것을 피할 수 있게 한다.

최근 네트워크 보안에 있어서 가장 도전적인 분야중 하나는 악성코드 탐지이며 모바일 기기에서의 악성코드 탐지, 특히 제로데이 공격은 가장 도전적인 문제라고 할

수 있다. 제로데이 공격은 공격자들은 이미 알고 있지만, 사용자들에게 공개적으로 알려지지 않은 취약점을 내포하는 프로그램들을 사용자들이 사용하도록 유도하여 해당 프로그램의 배포가 활발해 지거나 원하는 목표에 해당 프로그램이 도달하게 되면 공격이 시작된다. 이러한 제로데이 공격을 막는 방법 중 하나는 해당 프로그램이 수행되는 상황 및 로그를 지속적으로 수집하여 보안 위협을 분석해 내는 것이다. 만약 네트워크 프로그램의 보안 위협이 예상된다면 해당 프로그램을 분석 할 수 있는 상태로 오프로딩(offloading)하여 분석 작업을 주기적으로 수행해야 하며, 이는 시스템 운영에 있어서 큰 오버헤드로 작용하게 된다. 또한 모바일 에지컴퓨팅의 활성화에 따라, 종단 사용자의 의심 가는 프로그램을 오프로딩하여 정밀 분석하는 작업도 필요해 지는데, 이러한 오프로딩 작업은 네트워크 관리에 있어서 오버헤드로 생각될 수 있다.

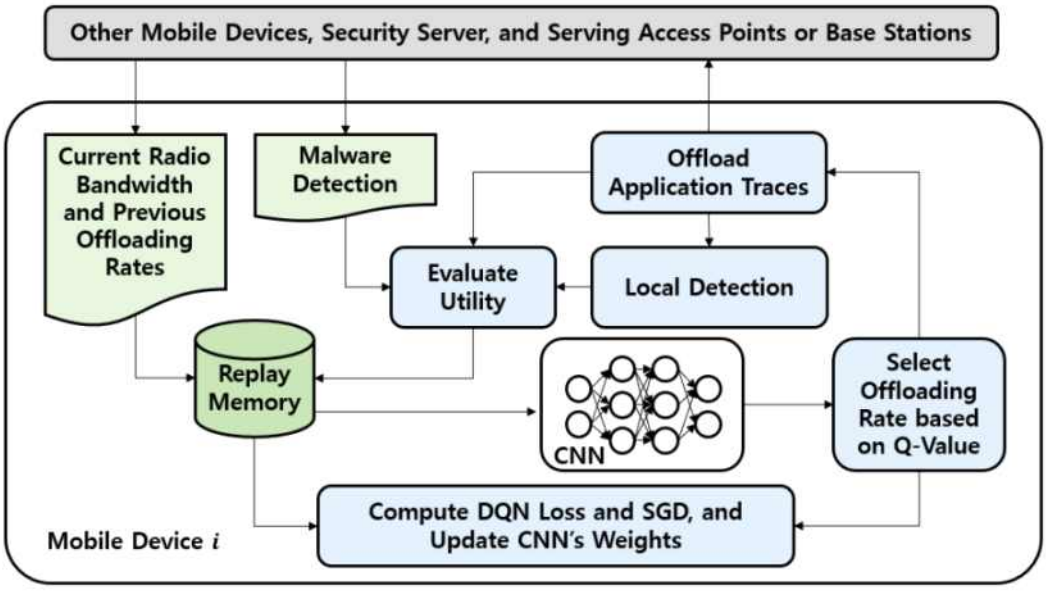


그림 12. DQN활용 클라우드 기반 악성코드 탐지 아키텍처

최근 그림 12와 같이 모바일 기기의 악성코드 탐지를 위한 오프로딩 작업의 효율을 높이기 위해 강화학습을 사용하는 기술이 연구되었다[20]. 제안된 기술에서는 다양한 악성코드 정보를 가지고 있는 클라우드에서 오프로딩된 프로그램(어플리케이션)을 정밀 분석하여 악성코드를 탐지한다. 이때 오프로딩하여 클라우드에 분석요청을 내리는 결정을 클라우드 악성코드 탐지결과 및 평가 유용도, 현재의 네트워크 대역폭 및 오프로딩 빈도 등을 사용하여 DQN 및 CNN을 활용하는 강화학습을 수행하는 모델을 사용한다. 학습된 모델을 이용해서 정밀 분석을 위한 프로그램 오프로딩의 빈도를 결정함으로써 네트워크 사용 효율을 최적화 할 뿐만 아니라 악성코드 탐지 정확도를 동시에 올릴 수 있다.

라. QoS/QoE

네트워크의 통신서비스 품질 향상을 위해 QoS 보장은 중요한 요소 중 하나이다. QoS는 네트워크의 품질을 사전에 합의된 통신 서비스 수준으로 표현한 값으로 네트워크 성능을 표현하는 지표인 QoS 및 QoE 목표 달성을 위해, 강화학습을 활용하는 연구가 진행되고 있다.

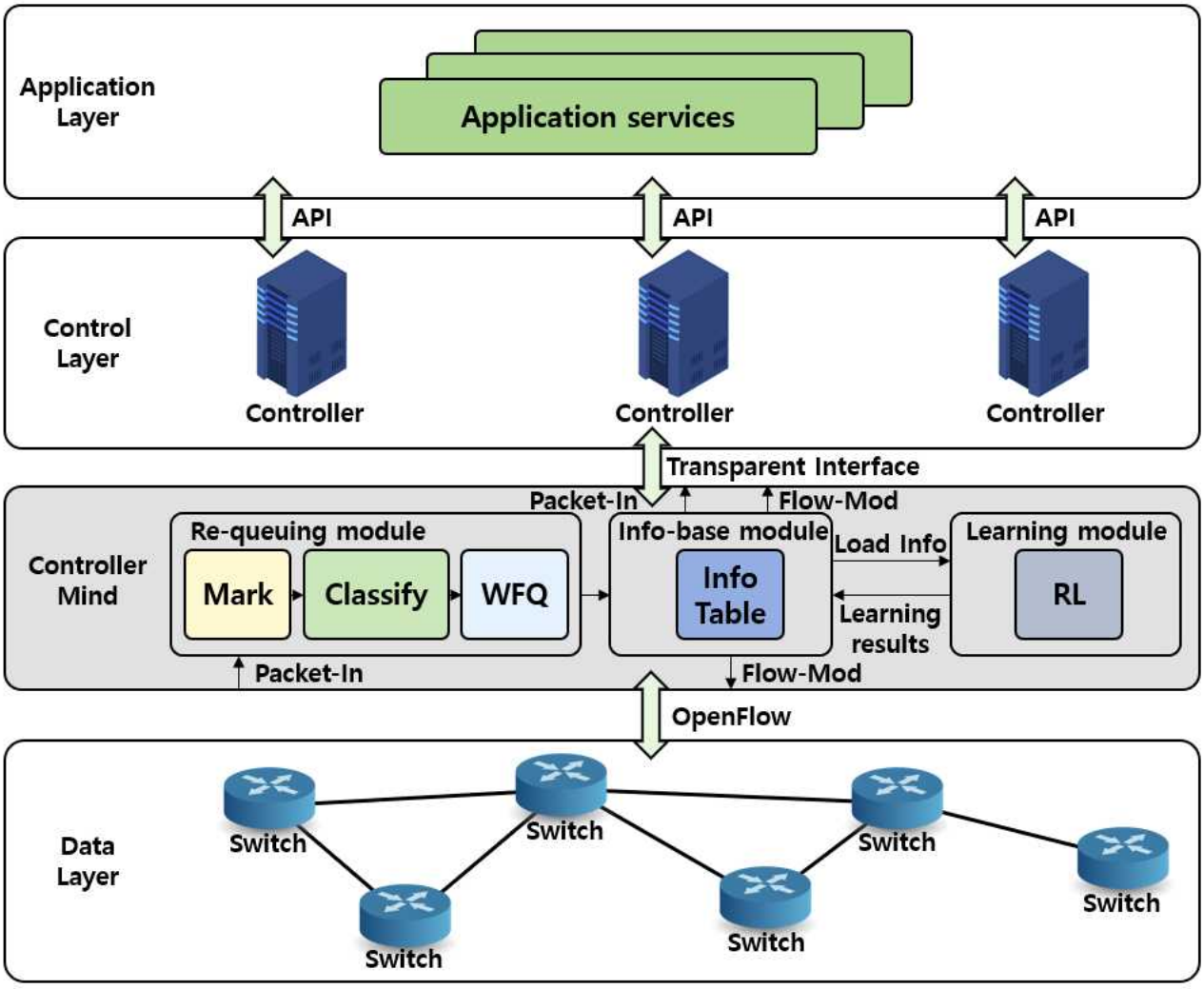


그림 13. 소프트웨어 정의 에너지 인터넷의 컨트롤러 마인드 (CM) 프레임워크

그림 13은 스마트 그리드 환경제공을 위해 구축된 데이터 네트워크를 위한 QoS 기반의 로드 스케줄링을 위한 Controller mind(CM) 프레임워크를 나타낸다[16]. CM 프레임워크는 컨트롤러 사이의 자동 관리를 구현하기 위해 사용되며 이는 강화학습을 통해 학습된다. 프레임워크는 Data Plane과 Control Plane 사이에 위치하며 Data Plane에서 받은 패킷을 받아 스케줄링을 한다. 패킷은 높은 우선순위를 갖는 QoS 큐와 낮은 우선순위를 갖는 Best-effort 큐로 분류되어 있다고 가정하며 Weight Fair Queuing(WFQ) 알고리즘을 이용하여 스케줄링을 진행한다. 이를 기반으로 Info table

을 만들고 학습 모듈을 이용하여 로드 정보를 구한 후 Control Plane에 넘겨준다. 때문에 컨트롤러 사이의 로드 변화와 QoS flow의 대기 시간을 줄이기 위해 강화학습을 통해 패킷을 컨트롤러에 할당되는 최적의 로드를 결정한다. 학습 모델의 State는 {QoS 레벨, 모든 컨트롤러의 로드 상태와 QoS flow의 수}로 이루어져 있다. 또한 Action은 패킷이 할당되어 있는 컨트롤러의 조합이며, Reward는 더 좋은 로드 밸런싱과 최소의 QoS flow 대기시간을 위한 Reward가 된다. 이를 통해 최선의 로드 스케줄링을 하여 효율성을 높인다.

QoS 뿐만 아니라 QoE 향상을 위한 적응형 트래픽 제어 메커니즘도 연구되고 있다[17]. 전체 네트워크에 큰 비율을 차지하는 멀티미디어 트래픽은 QoS보다 QoE의 최적화가 중요시된다. 하지만 거대해진 네트워크와 동적인 특성 때문에 QoE 최적화를 위한 트래픽 제어는 상당히 어려운 문제였다. 제안된 적응형 트래픽 제어 메커니즘은 3가지 전략을 사용하여 트래픽 제어를 시도한다. 첫째, 트래픽 제어를 위한 주요 척도로 QoE를 사용하였다. 둘째, 강화학습을 이용하여 문제를 해결하기 시도하며 연속적인 제어 문제를 해결하기 위해 DDPG를 사용한다. 셋째, 환경에서 빠르고 정확한 Reward을 얻기 위해 각 flow마다 심층신경망을 사용하여 QoE를 매핑한다. 학습을 위한 State는 flow의 상태이며 여기에는 대역폭, 지연, 패킷 손실률 등이 포함된다. 최적의 트래픽 제어 정책을 결정을 위한 Action은 경로 선택과 대역폭 적용이다. 이를 통해 결정된 State와 Action을 통해 환경으로부터 QoE인 Reward를 받으며 Mean Opinion Score (MOS)를 통해 QoE를 평가하여 측정한다. 하지만 실시간 MOS 측정이 힘들기 때문에 이를 위해 심층신경망을 통해 네트워크와 어플리케이션 메트릭스에 MOS를 매핑한다.

(4) 결론 및 시사점

네트워크 복잡성이 증가하고 서비스 요구사항이 더욱 엄격해지며 다양해짐에 따라 네트워크 자동화 기술은 여러 기업으로부터 큰 관심을 받고 있다. 또한 SDN/NFV 기술의 발달은 오버레이상에서 구현되던 네트워크 자동화 기술을 네트워크 계층에서 구현하는 것을 가능하게 하였다. 네트워크 자동화는 네트워크로부터 관찰되는 상태 데이터를 수집하고, 의사 결정에 사용되는 지식을 추출하며, 주어진 네트워크 자원으로 요구되는 네트워크 서비스를 관리하기 위한 제어하는 기능의 정의가 필요하다. 이러한 기능을 지원하기 위해 장치 측면에서 네트워크 측면에 이르기까지 유연성과

안정성을 향상시키기 위해 여러 가지 인공지능 기반 기술이 제안되고 있다. 특히 인공지능 기반 기술 중 강화학습은 네트워크 엔티티가 네트워크 환경의 불확실성을 고려한 상황에서 네트워크 성능을 최대화하기 위해 네트워크 상태를 고려하여 의사 결정 또는 행동을 포함한 최적의 정책을 얻을 수 있도록 사용된다. 이러한 강화학습은 시시각각 변하는 네트워크 서비스 요청 및 네트워크 상황을 지속적으로 학습함으로써, 자율적으로 네트워크를 관리하는 서비스를 가능하게 한다.

이러한 자율형 네트워크의 수요에 발맞춰 강화학습 기술 및 활용 연구도 활발히 진행중이다. 주요 강화학습 기법으로 Q-Learning, Deep Q-Learning (DQN), DDPG 등이 있다. 강화학습 기법을 적절히 활용하기 위해서는 에이전트의 환경에 대한 상태 변화, 행동 제어, 가치함수 설계, 보상함수 설계, 정책 개선 및 최적화 모델 도출을 적용하는 도메인에 알맞도록 설계해야 한다. 또한 고려해야하는 문제 공간의 크기에 따라서, 문제를 분할하거나, 심층신경망을 사용하여 학습효율 및 예측 정확도를 높일 수 있는 심층강화학습을 활용해야 한다. 또한 제어해야 하는 행동의 자유도가 매우 높은 경우 DDPG와 같은 모델의 활용이 필요할 수 있다.

자율형 네트워크의 실제적 구현을 위해 네트워크 관리를 위한 다양한 분야에서 강화학습을 활용하는 연구가 진행되고 있다. 특히 이 보고서에서는 5G, 6G 및 SDN/NFV로 구성되는 차세대 네트워크 환경에서 강화학습을 기반으로 라우팅, 리소스 관리, 네트워크 보안 및 QoS/QoE 문제를 해결하는 기술들을 분석하고 정리하였다. 라우팅 분야에서는 네트워크 트래픽이 기하급수적 증가에 따라 강화학습을 통해 라우팅 프로세스를 최적화하는 연구가 진행되고 있다. 리소스 관리 분야에서는 스마트 시티 또는 에지 클라우드와 같은 급변하는 네트워크 환경에서 효율적인 리소스 관리 및 스케줄링을 위해 강화학습을 접목시키는 연구가 진행중이다. 이를 통해 네트워크 혼잡을 제어하거나 오버헤드를 줄여 효율적인 네트워크 관리를 가능하게 한다. 네트워크 보안 분야에서는 네트워크에서 발생하는 네트워크 과부하와 같은 이상 징후감지 및 대응 기법에 강화학습을 적용하고 있다. QoS/QoE 분야에서는 강화학습을 통해 동적으로 변하는 네트워크 특성을 고려하여 전반적인 QoS/QoE를 향상시키는 연구가 진행되고 있다.

참 고 문 헌

[1] Ji, Yuefeng, et al. "Artificial intelligence-driven autonomous optical networks: 3S architecture and key technologies." Science China Information Sciences 63 (2020): 1-24.

[2] Kirkpatrick, Keith. "Software-defined networking." Communications of the ACM 56.9 (2013): 16-19.

[3] Mestres, Albert, et al. "Knowledge-defined networking." ACM SIGCOMM Computer Communication Review 47.3 (2017): 2-10.

[4] Even-Dar, Eyal, Sham M. Kakade, and Yishay Mansour. "Experts in a Markov decision process." Advances in neural information processing systems 17 (2005): 401-408.

[5] Watkins, Christopher JCH, and Peter Dayan. "Q-learning." Machine learning 8.3-4 (1992): 279-292.

[6] Mnih, Volodymyr, et al. "Playing atari with deep reinforcement learning." arXiv preprint arXiv:1312.5602 (2013).

[7] Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." nature 518.7540 (2015): 529-533.

[8] Zijin Jin, Weifei Zang, Yiming Jiang and Julong Lan. "A QLearning Based Business Differentiating Routing Mechanism in SDN Architecture" Journal of Physics: Conference Series. vol. 1168. page. 022025. Feb. 2019

[9] Rischke, Justus, et al. "Qr-sdn: towards reinforcement learning states, actions, and rewards for direct flow routing in software-defined networks." IEEE Access 8 (2020): 174773-174791.

[10] Ding, Ruijin, et al. "Packet routing against network congestion: A deep multi-agent reinforcement learning approach." 2020 International Conference on Computing, Networking and Communications (ICNC). IEEE, 2020.

[11] C. Yu, J. Lan, Z. Guo and Y. Hu, "DROM: Optimizing the Routing in Software-Defined Networks With Deep Reinforcement Learning," in IEEE Access, vol. 6, pp. 64533-64539, 2018.

[12] Zhang, Weiting, et al. "Deep Reinforcement Learning Based Resource Management

for DNN Inference in Industrial IoT." IEEE Transactions on Vehicular Technology (2021).

[13] Liu, Yi, et al. "Intelligent edge computing for IoT-based energy management in smart cities." IEEE network 33.2 (2019): 111-117.

[14] Nguyen, Thanh Thi, and Vijay Janapa Reddi. "Deep reinforcement learning for cyber security." arXiv preprint arXiv:1906.05799 (2019).

[15] Wu, Di, et al. "Evading machine learning botnet detection models via deep reinforcement learning." Proc. ICC, 2019.

[16] Qiu, Chao, et al. "A novel QoS-enabled load scheduling algorithm based on reinforcement learning in software-defined energy internet." Future Generation Computer Systems 92 (2019): 43-51.

[17] Huang, Xiaohong, et al. "Deep reinforcement learning for multimedia traffic control in software defined networking." IEEE Network 32.6 (2018): 35-41.

[18] Z. Xu et al. "Experience-driven networking: A deep reinforcement learning based approach." Proc. IEEE INFOCOM, pp. 1871-1879, Apr. 2018.

[19] P. Sun, et al. "TIDE: Time-relevant deep reinforcement learning for routing optimization." Future Gener. Compu. Syst., Vol. 99, pp. 401-409, Oct. 2019.

[20] X. Wan, et al. "Reinforcement learning based mobile offloading for cloud-based malware detection." Proc. IEEE GLOBECOM 2017, pp. 1-6