



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

2024년 2월
석사학위논문

모바일 IoT 서비스를 위한
인공지능 기반 LoRaWAN 자원 할당

조선대학교 대학원

정보통신공학과

윤정선

모바일 IoT 서비스를 위한
인공지능 기반 LoRaWAN 자원 할당

AI-based LoRaWAN Resource Allocation
for Mobile IoT Services

2024년 2월 23일

조선대학교 대학원

정보통신공학과

윤정선

모바일 IoT 서비스를 위한
인공지능 기반 LoRaWAN 자원 할당

지도교수

변재영

이 논문을 공학 석사학위신청 논문으로 제출함

2023년 10월

조선대학교 대학원

정보통신공학과

윤정선

윤정선의 석사학위 논문을 인준함

위원장 권구락 (인)

위 원 변재영 (인)

위 원 최동유 (인)

2023년 12월

조선대학교 대학원

목 차

표 목 차	iii
그림목차	iv
ABSTRACT	vi
I. 서 론	1
A. 연구 배경 및 목적	1
B. 연구 목표	3
II. LoRa 무선 통신 기술 및 관련 연구	5
A. LoRa 무선 통신	5
B. LoRaWAN	6
C. Adaptive Data Rate	9
D. 모바일 어플리케이션을 위한 Blind ADR	11
E. Hybrid ADR 기법	12
III. 제안하는 모바일 상태 식별 ADR	13
A. 제안하는 시스템 LoRaWAN 구성	13
B. 딥러닝 모델 학습	14
1. 시계열 딥러닝 모델 학습을 위한 데이터셋 생성	14
2. 모바일 단말과 정지 단말 간의 신호 분석	16
3. 딥러닝 모델 학습을 위한 시퀀스 데이터 생성	19
4. 시계열 딥러닝 모델 학습	20
C. 제안 시스템의 알고리즘 동작 및 구성	25
IV. 성능 비교 및 평가	27
A. 실험 환경 및 시나리오	27
B. 실험 결과 및 성능 평가	33

V. 결론 및 향후 연구과제	38
A. 결론	38
B. 향후 연구과제	39
참고문헌	40

표 목 차

표 1. DR 별 복조하기 위해 게이트웨이로부터 요구되는 SNR 값	9
표 2. DR 별 수집된 업링크 패킷 수	21
표 3. 딥러닝 학습 모델을 위한 하이퍼 파라미터	22
표 4. 모델과 시퀀스 길이에 따른 모델 성능 비교	23
표 5. LoRaWAN 네트워크 서버 구성	27
표 6. 실험 환경을 위한 LoRaWAN 파라미터	30

그림 목 차

그림 1. LoRaWAN 아키텍처	6
그림 2. LoRaWAN Class	8
그림 3. Blind ADR	11
그림 4. 제안하는 인공지능 기반 LoRaWAN 시스템	13
그림 5. 딥러닝 학습을 위한 데이터 수집 시나리오	15
그림 6. SDR로 관찰한 데이터 수집용 단말 동작	16
그림 7. 단말의 정지 및 모바일 상태에 대한 RSSI와 SNR 값 비교1	17
그림 8. 단말의 정지 및 모바일 상태에 대한 RSSI와 SNR 값 비교2	18
그림 9. LSTM 신경망 구조	20
그림 10. GRU 신경망 구조	21
그림 11. 제안하는 시스템의 딥러닝 모델 구조	22
그림 12. 모델 10의 mobility 분류 성능	24
그림 13. 네트워크 서버 측 MI-ADR	25
그림 14. 단말 측 MI-ADR	26
그림 15. LoRaWAN 게이트웨이 구성	28
그림 16. 게이트웨이의 실외 안테나	28
그림 17. 실험을 위한 실험 보드(ADR, BADR, MI-ADR)	29
그림 18. 단말의 모바일과 정지 상태가 포함된 실험 시나리오	31
그림 19. 실험 시나리오 출발 지점	32
그림 20. 실험 시나리오 반환 지점	32
그림 21. 첫 번째 실험의 PSR 성능	35
그림 22. 첫 번째 실험의 DR 분포	35
그림 23. 첫 번째 실험의 mobility 식별 성능	35
그림 24. 두 번째 실험의 PSR 성능	36
그림 25. 두 번째 실험의 DR 분포	36
그림 26. 두 번째 실험의 mobility 식별 성능	36
그림 27. 전체 실험의 PSR 성능	37

국문 초록

모바일 IoT 서비스를 위한 인공지능 기반 LoRaWAN 자원 할당

본 논문에서는 LoRaWAN (Long Range Wide Area Network)에서 정적 및 모바일 사물 인터넷 (Internet of Things, IoT) 응용 서비스를 위한 ADR (Adaptive Data Rate) 기술을 집중적으로 다루며, 네트워크 환경에서 자원 할당의 복잡성을 해결하고자 한다. 또한, 데이터 속도 (Data Rate, DR) 관리의 효율성을 극대화함으로써 단말의 배터리 수명 연장 및 네트워크 효율성 향상에 기여할 수 있는 새로운 접근법을 제안한다.

LoRaWAN은 IoT 애플리케이션 요구 사항을 충족하기 위해 ADR과 BADR (Blind ADR)을 각각 사용하여 정지 상태 및 모바일 상태 따라 단말에 자원을 할당한다. 그러나 이러한 ADR은 하나의 특정 어플리케이션을 위해 설계되었다. 따라서 다양한 IoT 어플리케이션을 수용하기 위해 정적 및 모바일 단말에 자원을 할당하는 모바일 상태 식별 ADR (Mobility Identification ADR, MI-ADR)이라는 새로운 접근 방식을 제안한다. 제안된 MI-ADR은 딥러닝 기법을 활용하여 단말의 모바일 상태를 식별하고 채널 상황을 고려하여 적절한 DR을 할당한다. 이는 기존 ADR 방식에 비해 향상된 성능을 제공하며, 특히 실시간 모바일 환경에서의 성능을 크게 향상시킨다. 본 논문에서 사전 학습된 모델은 별도의 위치 기반 센서에 의존하지 않고 LoRa 신호를 활용하여 움직임을 감지한다. 실험 결과에 따르면 MI-ADR은 정적 및 모바일 단말에 동시에 최적으로 자원을 할당하여 ADR 및 BADR에 비해 패킷 성공률을 각각 53% 및 33%까지 향상시켰다.

ABSTRACT

AI-based LoRaWAN Resource Allocation for Mobile IoT Services

Yoon Jeong-Sun

Advisor : Prof. Pyun Jae-Young, Ph.D.

Department of Information and Communication Engineering,
Graduate School of Chosun University

This thesis focuses on ADR (Adaptive Data Rate) technology for static and mobile IoT (Internet of Things) applications in LoRaWAN (Long Range Wide Area Network) and attempts to address the complexity of resource allocation in a network environment. In addition, we propose new approaches that can contribute to extending the battery life of devices and improving network efficiency by maximizing the efficiency of DR (Data Rate) management.

LoRaWAN assigns resources to static and mobile devices using ADR and blind ADR (BADR) to meet the IoT application requirements, respectively. However, these ADRs are designed for one specific application (e.g., either ADR for static or BADR for mobile devices). Therefore, to accommodate diverse IoT applications, we propose a novel approach called “MI-ADR (Mobility-Identification ADR)” for allocating resources to static and mobile devices.

The proposed MI-ADR utilizes deep learning techniques to identify the mobile state of the device and allocates an appropriate DR considering the channel condition. It provides improved performance compared to the existing ADR method and significantly improves performance, especially in real-time mobile environments. In this thesis, the pre-trained model uses LoRa signals to detect movement without relying on a separate location-based sensor. Test results

show that MI-ADR optimally allocates resources to static and mobile devices simultaneously, which improves the packet success ratio up to 53% and 33% compared to ADR and BADR, respectively.

I. 서론

A. 연구 배경 및 목적

최근 사물인터넷 (Internet of Things, IoT) 분야에서는 저전력 광역 네트워크 (Low-Power Wide-Area Network, LPWAN) 기술의 급속한 발전이 이뤄지고 있다 [1]. 이 기술은 장거리 통신과 저전력 사용이라는 두 가지 주요 요구사항을 충족시키며, Bluetooth나 Wi-Fi와 같은 근거리 통신 기술로는 해결할 수 없던 IoT 어플리케이션-스마트 그리드, 자산 추적 시스템 등에 이상적인 솔루션을 제공한다 [2]. SIGFOX, NB-IoT (Narrow Band Internet of Things), LoRaWAN (Long Range Wide Area Network) 등 LPWAN 기술 중에서도 LoRaWAN은 사용자에게 개방적인 표준 프로토콜을 제공하며, 사용자가 자신의 요구사항에 맞게 네트워크를 구성할 수 있도록 하는 유연성으로 인해 주목받고 있다 [3]-[4].

LoRaWAN 네트워크는 단말, 게이트웨이, 그리고 네트워크 서버로 구성되는 구조로, LoRa (Long Range) 무선 통신 기술을 기반으로 한 프로토콜을 사용한다. 해당 프로토콜은 물리 계층에서 칩 확산 스펙트럼 (Chirp Spread Spectrum, CSS) 변조기술을 채택하여 잡음이 많은 환경에서도 신뢰할 수 있는 통신을 가능하게 하며, 링크 계층에서는 pure ALOHA 방식을 이용하여 단말이 최소한의 전력을 사용하면서도 효율적으로 데이터를 송수신할 수 있도록 설계되었다 [5]. 그러나 LoRa 통신의 본질적인 특성으로 인해 발생하는 패킷의 Time on Air (ToA) 증가는 패킷 충돌의 가능성을 높이고, 단말의 전력 소비를 증가시키는 문제를 초래할 수 있다. 이뿐만 아니라, 간섭이 심한 무선 채널 환경이나 게이트웨이와 단말의 위치에 따라 패킷 손실이 발생할 가능성도 존재한다. 이와 같은 문제들을 최소화하기 위해서는 확산 인자 (Spreading factor, SF)와 송신 전력 (Transmit power, TP) 값을 적절하게 설정해야 하며, 이를 위해 LoRaWAN은 ADR (Adaptive Data Rate) 기술을 제공하고 있다 [6].

본 연구는 LoRaWAN이 주로 적용되는 정적이고 대규모의 IoT 연결을 요구하는 어플리케이션 뿐만 아니라, 동적인 환경에서도 효과적인 네트워크 자원 할당이 가능한가에 대한 질문에서 출발한다. 예를 들어, 현재 한국전력공사는 배전 설비의 모니터링 및

제어를 위해 LoRaWAN을 기반으로 한 시스템을 개발하고 있다 [7]. 모바일 어플리케이션 분야의 경우, SK텔레콤은 ‘스마트 지킴이’ 라는 플랫폼을 통해 치매 노인 및 발달 장애인과 같은 사회적 취약 계층의 실종자를 신속히 찾을 수 있도록 지원하는 LoRaWAN과 GPS (Global Positioning System) 기반의 웨어러블 트래커를 제공하고 있다 [8]. 이처럼 다양한 분야에서 활용되고 있는 웨어러블 트래커는 배터리 소모가 적어야 하며, 사용자가 이동 중일 때도 통신이 중단되지 않아야 한다는 중요한 전제조건을 가지고 있다.

본 연구는 딥러닝 기반의 학습 모델을 활용하여 ADR의 문제를 해결하고 네트워크 자원을 보다 효율적으로 할당할 수 있는 방법론을 제시하고자 한다. 제2장에서는 LoRa 무선 통신과 LoRaWAN의 ADR 기술, 그리고 이와 관련된 선행 연구들을 분석한다. 제3장에서는 시스템 모델을 상세히 설명하고, 본 연구에서 제안하는 새로운 기법에 관해 탐색한다. 이어지는 제4장에서는 실제 환경에서의 실험을 통해 제안된 기법의 성능을 분석하고, 제5장에서는 본 연구의 결론을 도출하며, LoRaWAN의 ADR 기술이 미래의 IoT 네트워크에서 어떻게 활용될 수 있는지에 대한 전망을 제시하고자 한다.

본 연구의 목적은 기존의 LoRaWAN ADR 기술의 한계를 극복하고, 특히 모바일 환경에서의 정확한 네트워크 자원 할당을 최적화함으로써 IoT 분야에서 LoRaWAN 기술의 적용 범위를 확장하는 데 있다. 이는 딥러닝 알고리즘을 이용하여, 네트워크의 다이내믹한 변화에 효율적으로 대응할 수 있다. 이를 통해 IoT 네트워크 설계 및 운영에 있어서 새로운 기준을 제시할 것으로 기대한다.

B. 연구 목표

IoT 기술이 날로 발전하고 있는 가운데, 저전력 네트워크 기술의 대표주자인 LoRaWAN은 그 효율적인 통신 지원 능력으로 저전력 센서 및 단말과의 상호작용에 있어서 중요한 역할을 담당하고 있다. 이러한 배경 하에, 본 연구는 딥러닝 기반의 접근 방식과 LoRaWAN의 자원 할당 기술인 ADR에 중점을 두고, 네트워크의 효율성과 에너지 소비 최소화에 기여할 수 있는 방법을 제시한다.

본 논문에서는 다음 내용과 같이 LoRaWAN 특성 소개 및 분석을 포함한다.

1. LoRaWAN ADR 알고리즘 분석: 본 연구는 기존 ADR 알고리즘을 분석하여, 통신 효율성을 극대화하고 에너지 소비를 최소화할 방법을 모색한다. 이를 위해 기계학습, 특히 딥러닝 기법을 활용하여 기존 ADR 알고리즘의 한계를 극복하고, 더욱 진보된 알고리즘을 개발한다.

2. 네트워크 트래픽 및 성능 분석: ADR이 적용된 네트워크의 트래픽 패턴과 성능을 심층적으로 분석하고, 이를 통해 네트워크의 통신 효율성을 최적화할 수 있는 매개변수를 도출한다. 이 분석은 네트워크 상태의 다양한 시나리오를 포괄하여, 실제 운영 환경에서의 예측 불가능한 변화에도 효과적으로 대응할 수 있다.

3. 에너지 효율성 평가: 에너지 소비는 IoT 네트워크 운영의 핵심 요소 중 하나이다. 이 연구는 ADR을 적용한 시나리오와 그렇지 않은 시나리오에서의 에너지 소비를 비교 분석하여, ADR 알고리즘의 에너지 효율성을 평가한다. 이 평가를 통해 ADR이 에너지 소비에 미치는 영향을 정량적으로 측정하고, 이를 기반으로 에너지 효율성을 극대화하는 새로운 ADR을 제안한다.

4. 실제 환경에서의 실험 및 적용: 개발된 ADR 알고리즘을 실제 환경에서 실험하여, 현장에서의 적용 가능성을 평가한다. 실험은 다양한 환경과 조건에서 진행되어야 하며, 결과는 IoT 시스템 설계자와 운영자에게 실질적인 가이드라인을 제공할 수 있다. 이러한 실험적 접근은 이론과 실제의 간극을 좁히고, 현실 세계에서 LoRaWAN 기반 시스템의 구현 가능성을 검증하는 데 기여할 수 있다.

본 연구는 딥러닝 기법을 통합하여 LoRaWAN의 ADR 기술을 한 단계 더 발전시킬 수 있는 방안을 제시함으로써, IoT 네트워크의 효율성과 지속 가능성을 개선하는 것을 목표로 한다.

II. LoRa 무선 통신 기술 및 관련 연구

A. LoRa 무선 통신

LoRa는 저전력 장거리 무선 통신 기술로서, 특히 저전력 IoT 단말들 간의 통신에 적합한 기술이다. 이 기술이 장거리 통신을 가능하게 하는 핵심은 CSS 변조기술에 있다. CSS는 시간에 따라 주파수가 변화하는 정현파 신호를 사용하여, 신호가 잡음이나 간섭에도 강한 특성을 가진다 [5]. LoRa의 주요 특징은 심볼 지속 기간 (Symbol Duration, T_s)이 SF 및 대역폭 (Bandwidth, BW)에 따라 달라지는 점이다. T_s 를 결정하는 수식은 다음과 같다.

$$T_s = \frac{2^{SF}}{BW} \quad (1)$$

(1)에 따르면, 대역폭이 고정되어 있을 때, SF 의 값이 증가함에 따라 T_s 는 기하급수적으로 증가하게 된다. 이는 ToA, 즉 메시지 전송 시간을 증가시켜 장거리 통신을 실현시킨다. 또한, SF 간의 직교성은 동일한 채널에서 서로 다른 SF 값을 사용하는 신호들이 게이트웨이에 의해 동시에 수신될 수 있도록 한다. 이러한 직교성은 LoRa 네트워크의 효율성을 높이는 중요한 요소이다.

하지만 이와 같은 T_s 의 증가는 채널 점유 시간을 늘리며, 이는 통신 채널에서의 패킷 충돌 가능성을 높이는 결과를 낳는다. 이는 LoRa 통신에서의 DR 감소라는 문제점으로 이어진다. 즉, SF 가 높아질수록 DR 은 반감하게 되며, 이는 LoRa 기술이 장거리 통신에는 유리하지만, DR 이 중요한 어플리케이션에는 제약이 될 수 있다는 것을 의미한다. 이러한 특성들은 통신 범위와 DR 사이의 균형을 어떻게 설정할 것인가에 관한 연구를 요구한다. 이를 위해, 본 연구는 단말의 상태에 따라 최적의 DR 을 찾아내는 것을 목표로 하며, 이를 통해 LoRa 기술의 장점을 극대화하고 단점은 최소화할 수 있는 방법을 제안한다.

B. LoRaWAN

LoRaWAN은 LoRa 무선 통신 기술을 기반으로 한 저전력 광역 네트워크 통신 프로토콜이다. 이 프로토콜은 그림 1의 단말과 게이트웨이 간의 효율적인 통신을 관리하고, 네트워크 서버에서 강력한 보안성 및 뛰어난 통합성과 같은 여러 중요한 장점을 제공한다. LoRaWAN의 대표적 기술적 구성은 LoRaWAN 아키텍처, 단말의 네트워크 가입 절차, 클래스, 그리고 DR을 조절하는 ADR 등 네 가지 핵심 요소로 요약할 수 있다.

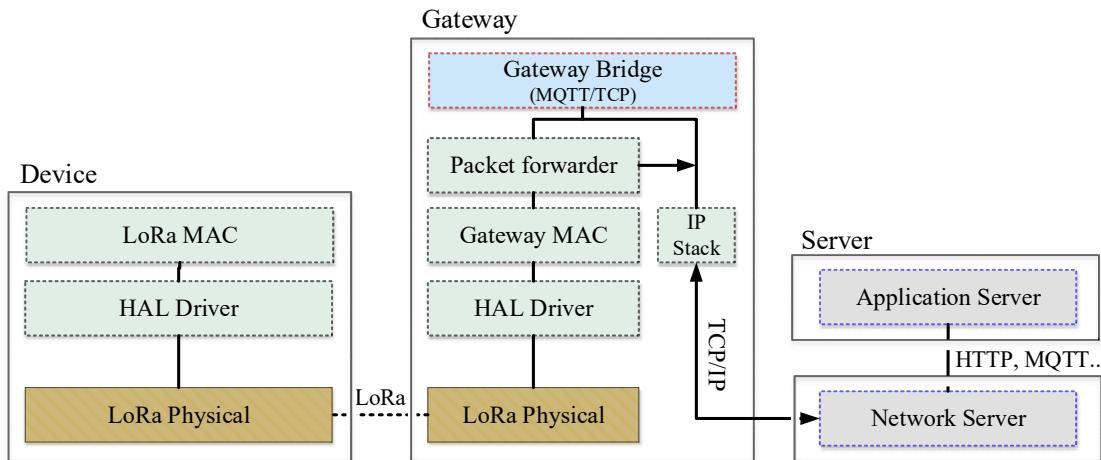


그림 1. LoRaWAN 아키텍처

첫째, LoRaWAN은 그림 1과 같이 단말, 게이트웨이, 네트워크 서버, 그리고 어플리케이션 서버로 구성된다. 단말은 두 가지 주요 구성 요소로 이루어진다 [11]. 단말의 구성 중 하나인 LoRa MAC (Mac Access Control) 소프트웨어는 LoRaWAN의 MAC 계층 프로토콜을 구현한다. 이 소프트웨어는 물리 (Physical) 계층과의 인터페이스 역할을 하는 HAL (Hardware Abstraction Layer) 드라이버를 통해 LoRa 물리 계층을 제어하고, 이를 통해 단말이 LoRaWAN 네트워크 내에서 통신할 수 있도록 한다. 물리 계층은 LoRa 변조 및 복조를 담당하며, 네트워크의 다른 단말 및 게이트웨이와의 무선 통신을 가능하게 한다. 게이트웨이는 Gateway MAC 소프트웨어를 통해 동작하며, 이 또한 HAL 드라이버를 통해 물리 계층을 제어한다. 게이트웨이는 단말로부터 수신한 패킷을

패킷 전달자 (Packet forwarder) 소프트웨어를 통해 IP (Internet Protocol) 스택에 전달하고, 이는 네트워크 서버로의 패킷 전달을 담당한다. 패킷 전달 방식은 MQTT (Message Queuing Telemetry Transport) 프로토콜을 사용한 TCP (Transmission Control Protocol) 전송을 활용하여, 신뢰성 높은 메시지 전송을 보장한다. 네트워크 서버는 다수의 게이트웨이로부터 수신된 업링크 패킷을 처리하고, 이를 통해 IoT 네트워크의 효율적인 운영을 지원한다. 이 서버는 인터넷 연결을 통해 게이트웨이에서 전송된 중복된 업링크 패킷을 식별하여 제거함으로써, 네트워크의 대역폭 사용을 최적화한다. 또한, 수신된 프레임에 대한 정보를 분석하여 어플리케이션 서버에 필요한 결과를 제공하며, 채널이 한정된 상황에서 많은 단말들을 효과적으로 관리하기 위해, MAC Command를 발행한다 [11]. 이는 ADR의 주요 기능 중 하나이다. 어플리케이션 서버는 스마트 그리드와 같이 어플리케이션에 따라 네트워크 서버와 통신하며, IoT 사용자에게 서비스를 제공하는 역할을 한다.

둘째, 서버에서 단말이 송신한 업링크 패킷을 복호화하기 위해 네트워크 가입 절차가 필요하다. 가입 절차는 크게 OTAA (Over-The-Air Activation)와 ABP (Activation By Personalization) 두 가지 방식으로 나뉜다. 각 가입 방식의 차이점은 인증 메커니즘에 있다. ABP 방식은 네트워크와 통신하기 위해 단말이 사전에 할당된 어플리케이션 키와 네트워크 키를 보유하고 있는 방식이다. 즉, 단말이 네트워크 서버로 패킷을 송신할 때, 이 두 키가 서버의 키와 일치하면 네트워크 서버는 해당 패킷을 성공적으로 복호할 수 있다. 반면, OTAA 방식은 완전한 암호키를 미리 가지고 있지 않기 때문에, 단말이 네트워크에 가입하기 위해 Join Request와 Join Accept 과정을 거치게 된다. 이 Join 과정은 기본 채널에서만 이루어져야 한다 [9]-[10]. OTAA 방식은 Join 절차 과정에서 무선 주파수 해킹을 통해 암호키가 노출될 수 있는 가능성이 있으므로, ABP 방식에 비해 보안이 더 취약하다.

셋째, LoRaWAN의 클래스는 그림 2와 같이 Class A, B, C로 나뉘며, 각각의 클래스는 다른 에너지 소비 패턴과 통신 행위를 가진다 [11]. Class A는 단말이 업링크 패킷을 송신 후, 게이트웨이가 송신한 다운링크 패킷을 수신할 준비를 하므로 단말이 다운링크에 대한 수신 성공률은 낮지만 전력 소모는 가장 적다. Class B는 단말이 게이트웨이로부터 시간 동기화 비콘 신호를 받고, 이 비콘 신호를 기반으로 정해진 시간에

데이터를 수신할 수 있는 추가적인 슬롯을 할당받는다. 이는 다운링크 패킷을 수신받을 수 있는 성공률은 증가시키지만, Class A 대비 높은 전력을 소모한다. Class C는 지속적인 전력 공급이 가능한 환경에서 사용되며, 단말이 게이트웨이로부터 신호를 수신할 준비가 항상 되어 있는 상태이므로 전력 소모가 가장 심하다.

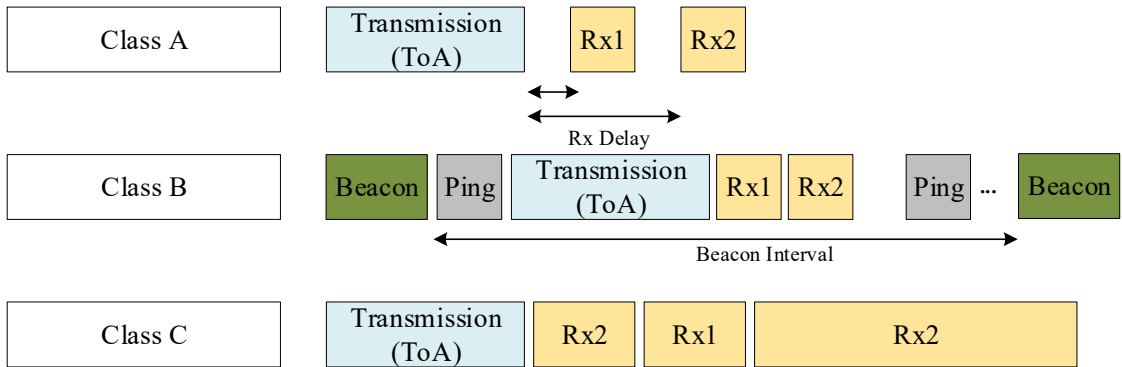


그림 2. LoRaWAN Class

넷째, LoRaWAN 사양에 따르면, 단말의 에너지 효율성을 최대화하고 네트워크 용량을 향상시키기 위해 ADR을 사용할 것을 권장한다. LoRa 기술은 CSS 변조 방식을 통해 장거리 통신을 가능케 하나, 채널 조건에 부합하지 않는 DR 설정은 전력 소모를 증가시키고 네트워크 내 단말의 수가 증가함에 따라 패킷 충돌의 가능성을 높인다. ADR은 이러한 문제를 해결하기 위해 네트워크 서버가 단말로부터 수신한 신호 대 잡음 비 (Signal to Noise Ratio, SNR)를 분석하여 적절한 DR과 TP를 선택하도록 설계되었다 [12]-[13].

C. Adaptive Data Rate

LoRaWAN은 ADR을 알고리즘 1의 네트워크 서버 측 ADR과 알고리즘 2의 단말 측 ADR 알고리즘으로 정의한다. 네트워크 서버 측 ADR은 단말이 ADR 활성화 비트를 설정하여 업링크 패킷을 송신하면 시작된다. 이 패킷이 게이트웨이에 의해 성공적으로 수신되면, 네트워크 서버는 해당 단말로부터 수신한 최근 업링크 패킷들 중 가장 높은 SNR (SNR_{max})을 활용한다. SNR_{max} 을 통해 게이트웨이가 정의한 표 1의 요구 SNR (SNR_{req}) 과 기본 10으로 설정된 $Device_{margin}$ 을 활용하여 SNR_{margin} 을 계산한다. SNR_{margin} 은 실제 SNR 과 SNR_{req} 의 차이이며, 안정적인 통신을 위해 요구되는 여유 값이다. SNR_{margin} 을 통해 단계별로 DR 과 TP 를 제어한다. 만약 DR 이 가장 최대치인 $DR5$ 에 도달하면, 단계에 따라 TP 를 2 dBm 씩 감소시켜 단말의 전력 소모를 최소화시킨다. 채널 환경에 따라 단말과 게이트웨이 간 최적의 DR 과 TP 를 찾아 안정화될 때까지 수렴 기간 (Convergence period)을 가진다 [12].

표 1. DR 별 복조하기 위해 게이트웨이로부터 요구되는 SNR 값

DR	SNR [dBm]
5	-7.5
4	-10.0
3	-12.5
2	-15.0
1	-17.5
0	-20.0

단말 측 ADR은 네트워크 서버가 설정한 DR 과 TP 가 실제 통신 환경에서 적절하지 않을 경우, 단말의 자가 복구를 위해 그림 4와 같이 DR 과 TP 를 조정할 수 있게 설계되었다. ACK 는 다운링크 패킷 중 하나인 $LinkADRReq$ 에 대한 수신 여부이다. 단말은 업링크 패킷을 송신할 때마다 ADR_ACK_CNT 를 증가시킨다. 네트워크 서버로부터 단말이 ACK 를 수신하면 ADR_ACK_CNT 를 초기화시킨다. 하지만 ACK 를 수신하지 못하여 ADR_ACK_LIMIT 를 초과했을 때, 단말은 DR 을 강제로 감소시켜 통신 복구를 우선시한다.

Algorithm 1: Network server-side ADR.

Input : $TP = 2 \sim 14$ dBm, $DR = 5 \sim 0$, SNR_{req} , $Device_{margin}$

Output: DR and TP

```
1 a.  $SNR_{max} = \text{Max}(SNR \text{ of last 20 uplink packet})$ 
2 b.  $SNR_{req} = \text{demodulation floor}(\text{current } DR)$ 
3 c.  $Device_{margin} = 10$  (LoRaWAN default)
4 d.  $SNR_{margin} = (SNR_{max} - SNR_{req} - Device_{margin})$ 
5 e.  $steps = \text{int}(SNR_{margin}/3)$ 
6 At each uplink packet reception
7 while ( $steps > 0$  and  $DR < DR5$ ) do
8   |    $DR = DR + 1$ 
9   |    $steps = steps - 1$ 
10 end
11 while ( $steps > 0$  and  $TP > TP_{min}$ ) do
12   |    $TP = TP - 2$  dBm
13   |    $steps = steps - 1$ 
14 end
15 while ( $steps > 0$  and  $TP < TP_{max}$ ) do
16   |    $TP = TP + 2$  dBm
17   |    $steps = steps - 1$ 
18 end
19 Network server transmits LinkADRReq
```

알고리즘 1. 네트워크 서버 측 ADR 알고리즘

Algorithm 2: Device-side ADR.

Input : $DR = 5 \sim 0$, $ADR_ACK_LIMIT = 64$, $ADR_ACK_CNT = 0$,
 $TP = 14$ dBm

Output: DR and TP

```
1 At each uplink packet transmission
2  $ADR\_ACK\_CNT += 1$ 
3 if ( $ADR\_ACK\_CNT \geq ADR\_ACK\_LIMIT$ ) then
4   |    $DR = DR - 1$ 
5 end
6 if ( $ACK == TRUE$ ) then
7   |   reset  $ADR\_ACK\_CNT$ 
8 end
9 if ( $DR == DR5$ ) then
10  |    $TP = TP - 2$  dBm
11 end
```

알고리즘 2. 단말 측 ADR 알고리즘

D. 모바일 어플리케이션을 위한 Blind ADR

LoRaWAN 네트워크에서 ADR은 기존의 채널 상태를 반영하여 단말과 게이트웨이 간의 DR 을 최적화한다. 이 과정은 최근 업링크 패킷들의 SNR 을 기반으로 계산된 SNR_{margin} 을 통해 단계별로 수행된다. 일단 단말이 수렴 기간을 지나 안정화에 접어들면, 네트워크 서버는 이를 주기적으로 유지하며, 다운링크 패킷의 송신을 줄인다. 이는 고정된 위치에서 대규모 IoT 연결을 처리하는 스마트 그리드와 같은 정적 어플리케이션에 적합하지만, 모바일 어플리케이션의 경우, 채널 조건이 빠르게 변화함에 따라 기존 ADR이 적절히 대응하기 어렵다 [14]-[15].

LoRa를 개발한 Semtech은 모바일 어플리케이션, 예를 들어 실종자 추적 시스템 [16]과 같이 채널 상태가 급격히 변화하는 환경을 위해 BADR (Blind ADR)을 제안하였다. BADR은 단말과 게이트웨이 간 별도의 데이터 송수신 없이, 사전에 설정된 일련의 DR 스케줄을 기반으로 동작한다. 예를 들어, 그림 3과 같이 단말이 각각 $DR5$ 를 사용하여 3회, $DR2$ 를 사용하여 2회, $DR0$ 를 사용하여 1회, 1시간 동안 반복하는 패턴이다. 이러한 패턴은 단말이 게이트웨이와의 거리에 무관하게 일정한 통신 가능성을 확보하는 데 유리하다. 그러나 이 방식은 단말이 게이트웨이에 가까울 때 불필요하게 높은 DR 을 사용할 수 있어, 장기적으로 배터리 사용 측면에서 비효율적일 수 있다는 단점이 있다. 게다가, 단말이 실제 채널 상태보다 높은 DR 을 사용할 경우, 패킷 손실의 위험성이 증가하게 된다.

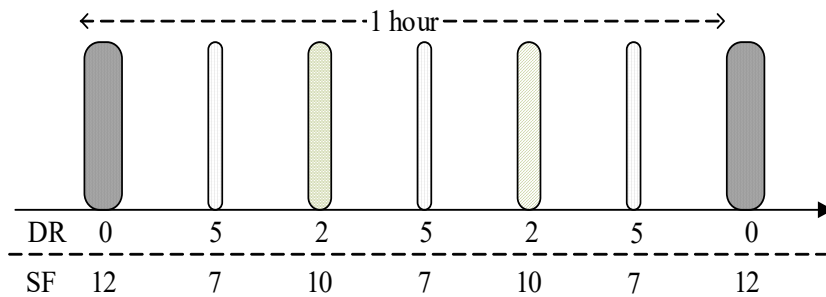


그림 3. Blind ADR

E. Hybrid ADR 기법

정적 및 모바일 IoT 어플리케이션에 대해 동시에 적절한 자원을 할당하는 것은 상당히 어렵다. ADR과 BADR은 각각 정적 및 모바일 상태의 단말에 적용되었으나, 이 두 자원 할당 방식은 서로 다른 상황에 특화되어 있어 한계가 명확하다.

HADR은 이러한 한계를 극복하고자, 모바일 상태인 단말에 기반한 자원 할당 방식을 채택한다 [17]. 이 기법은 단말의 이전 위치와 현재 위치 간 거리를 계산한 후, 그 거리가 한계치에 도달하면 단말의 모바일 상태로 판단하고, 그에 따라 적절한 ADR 설정을 선택하는 방법이다. 단말이 정지 상태일 경우, ADR을 적용하여 데이터 전송률을 최적화하고, 단말이 모바일 상태일 경우, BADR을 적용하여 데이터 전송의 신뢰성을 보장한다.

HADR은 정확한 거리 계산을 요구한다. 하지만 실제 환경에서 LoRa 신호를 활용하여 경로 손실 모델을 통해 거리를 계산하는 것은 불가능에 가깝다. 그러므로 정확한 거리를 구하기 위해선 위치 기반 서비스가 필수적이다. GPS와 같은 위치 센서의 활용은 단말의 정확한 위치 정보를 제공하지만, 이는 단말의 소모 전력 증가와 업링크 패킷 크기의 증가라는 부담을 초래한다. 특히, 배터리 수명이 중요한 IoT 어플리케이션에서는 이러한 추가적인 전력 소모가 전체 시스템의 효율성 저하와 비용을 증가시킨다 [18]. 본 연구는 이러한 HADR 접근법이 가진 잠재력을 인식하고, 실제 환경에서의 복잡한 환경에서도 단말의 모바일 상태 여부를 식별할 수 있는 방법을 제시한다.

III. 제안하는 모바일 상태 식별 ADR

A. 제안하는 시스템 LoRaWAN 구성

무선 신호 세기인 *RSSI* (Received Signal Strength Indicator)와 *SNR*를 활용한 측위 및 ADR 알고리즘 향상에 대한 다양한 연구가 진행되고 있다. 그러나 동적 실외 조건에서는 건물, 자동차 같은 주파수 간섭으로 위치 추정 성능이 저하된다. 따라서 실제 환경에 대한 채널 조건을 학습하고 단말의 모바일 상태를 식별하는 방법이 필요하다.

본 연구에서는 이 식별 능력을 향상시키기 위해 인공지능 기반 LoRaWAN 시스템을 제안한다. 본 연구에서 제안하는 시스템은 LoRa Alliance의 사양을 준수하며, 각 구성 요소는 특정 기능을 수행하고, 이들 간의 상호작용은 효율적이고 안정적인 운영을 보장한다.

그림 4는 제안된 LoRaWAN에서 인공지능 기반 모바일 상태 식별을 위한 흐름도를 제시한다. 먼저 단말은 게이트웨이에게 업링크 패킷을 송신한다. 네트워크 서버는 업링크 패킷에 대한 *RSSI*와 *SNR*는 AI 모델 추론 단계의 입력으로 제공된다. AI 추론 단계에서 결정된 단말의 모바일 상태 여부는 네트워크 측 MI-ADR에 입력되어 결정된 값을 단말에게 다운링크를 통해 송신한다. 마지막으로 단말이 다운링크를 수신하면 단말 측 MI-ADR이 단말의 최종 *DR*을 결정한다.

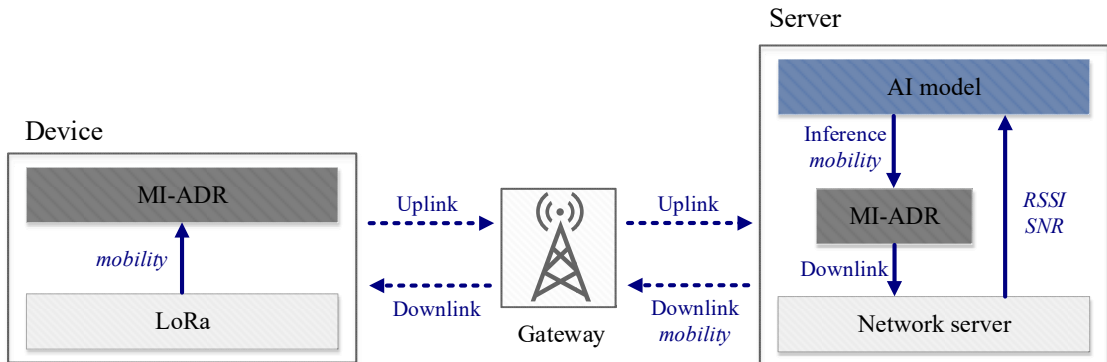


그림 4. 제안하는 인공지능 기반 LoRaWAN 시스템

B. 딥러닝 모델 학습

LoRaWAN은 ADR을 활용하여 네트워크의 자원인 *DR*과 *TP*를 할당하여 대용량 데이터를 처리하는 정적 어플리케이션에서는 효과적이었으나, 채널 환경이 급변하는 모바일 상태인 단말에 대해서는 최적화된 결과를 제공하지 못한다. 이러한 문제를 해결하기 위해, 채널 환경을 고려하지 않고 BADR은 단말에 사전에 설정된 *DR*을 사용하여 업링크 패킷을 송신한다. HADR은 두 ADR의 문제를 해결하기 위해 단말의 모바일 상태를 판단하여 상황에 맞게 ADR을 활용했다.

하지만, 실제 환경에서 경로 손실 모델을 통해 거리를 계산하는 것은 굉장히 어려우며, 다양한 조건으로 인해 채널 환경을 예측하기 어렵다. 본 연구에서는 단말의 상태를 보다 효율적으로 식별하기 위해 딥러닝 모델 접근법을 채택하였다. 이 모델은 네트워크 서버에서 단말의 업링크 패킷에 대한 *RSSI*와 *SNR* 값을 수집하고 분석한 후, 다양한 채널 상태와 단말의 모바일 패턴을 학습한다.

딥러닝 모델의 학습 과정은 수많은 변수와 가능한 시나리오를 고려한다. 이는 단순히 과거 데이터에 기반한 추론을 넘어서, 실시간 데이터와 시계열 분석을 통합하여, 단말이 경험할 수 있는 다양한 조건에 대응할 수 있도록 한다. 연구 과정에서 모델은 지속적으로 평가되며, 네트워크의 실제 운영 환경에서 발생할 수 있는 예상치 못한 변화에 대응할 수 있는 유연성을 갖추게 된다.

1. 시계열 딥러닝 모델 학습을 위한 데이터셋 생성

딥러닝 모델의 효과적인 학습을 위해 다양한 환경에서 수집된 대량의 시계열 데이터가 필수적이다. 이러한 데이터는 모델이 시간에 따른 패턴과 상관관계를 인식하고, 이를 기반으로 실시간 예측을 수행할 수 있는 기반을 마련한다. 본 연구에서는 그림 5에 나타난 바와 같이 설계된 경로를 따라 단말을 사용하여 데이터를 수집하였다. 시나리오의 범위는 각 675.6 m, 427.5 m이다. 각 건물 구간별로 이동하며 학습 데이터를 수집함으로써, 단말이 겪게 될 다양한 환경적 조건들을 포괄적으로 반영하였으며, 그림 5와 같이 단말이 송신한 패킷을 게이트웨이가 전혀 수신하지 못하는 수신 불가 지역도 포함된다. 실험은 미리 정의된 경로를 따라 이동하며, 무작위로 정지하였다. 또한, 각 단말에는 서로 다른 채널과 *DR*이 할당되었으며, 이는 LoRa 채널과 *DR*이 수신 신호에 미치는 영향을 독립적으로 분석할 수 있게 한다.

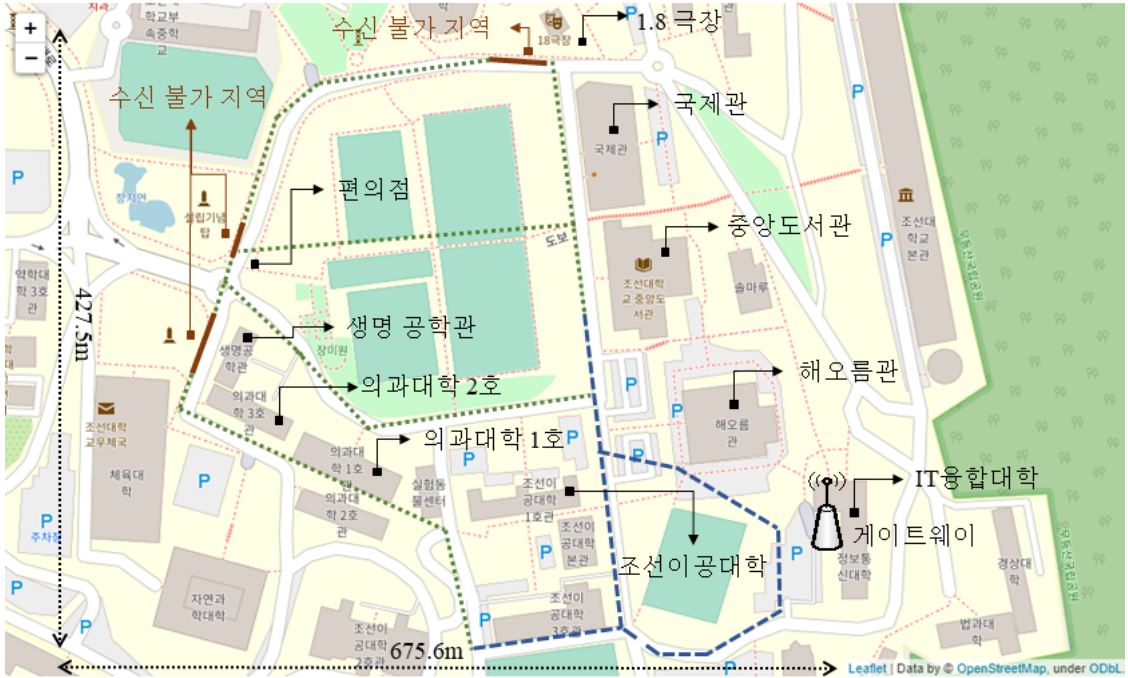


그림 5. 딥러닝 학습을 위한 데이터 수집 시나리오

그림 6은 6개의 단말에 서로 다른 채널에 DR 을 할당하여, SDR (Software Defined Radio) 장비를 사용하여 관측한 결과를 나타낸다. 이러한 관측은 실제 환경에서 SF 의 증가가 ToA 에 미치는 영향을 실증적으로 보여주며, 데이터 수집 방법을 보여준다.

각 채널에 단말을 고정적으로 할당하여 DR 을 단계적으로 낮추면서 데이터를 수집하였다. 922.3MHz 대역에 할당된 단말은 $DR5$ 에서 시작하여 $DR0$ 까지 순차적으로 낮추는 과정을 반복하고, 923.3MHz 대역에 할당된 단말은 $DR0$ 에서 시작하여 $DR5$ 로 초기화하여 반복하게끔 설정됐다. 수집된 데이터가 동일한 환경에서 DR 의 전체 범위를 포괄시켜, 모델에게 다양한 채널 환경을 포함하는 데이터셋을 제공한다.

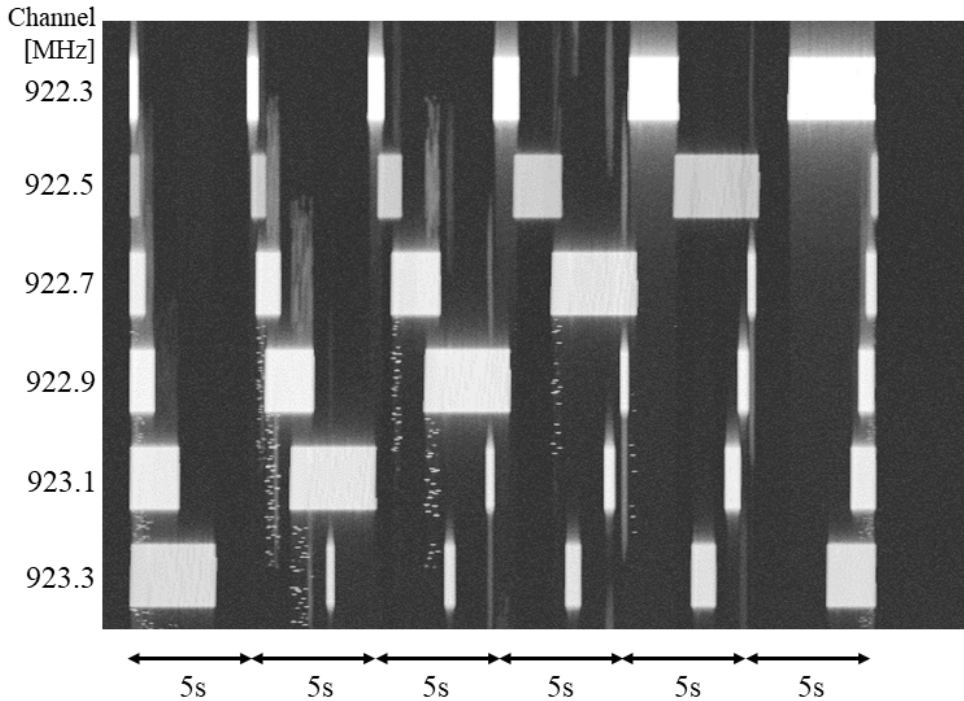


그림 6. SDR로 관찰한 데이터 수집용 단말 동작

2. 모바일 단말과 정지 단말 간의 신호 분석

본 연구에서 데이터 분석은 단말의 위치 및 상태 변화에 따른 *RSSI* 및 *SNR* 값의 변동성을 조사한다. 그림 7은 비슷한 위치에서 단말의 정지 상태와 모바일 상태에 대한 *RSSI*와 *SNR* 값을 각각 보여준다. 단말이 모바일 상태일 때, 신호 데이터의 변동성이 증가하는 것을 분산 값 2.0365 대비 3.7173으로 확인할 수 있었다. 이는 모바일 상태의 단말이 수신 신호에 더 큰 변동성을 경험함을 나타낸다.

그림 7의 데이터에서 주목할 만한 부분은 모바일 단말이 보여지는 70~100초 사이에서 더 안정적인 *RSSI* 값을 나타낸다는 것이다. 이는 고정된 위치의 단말이 항상 더 안정적인 신호 강도를 보이는 것은 아님을 시사한다.

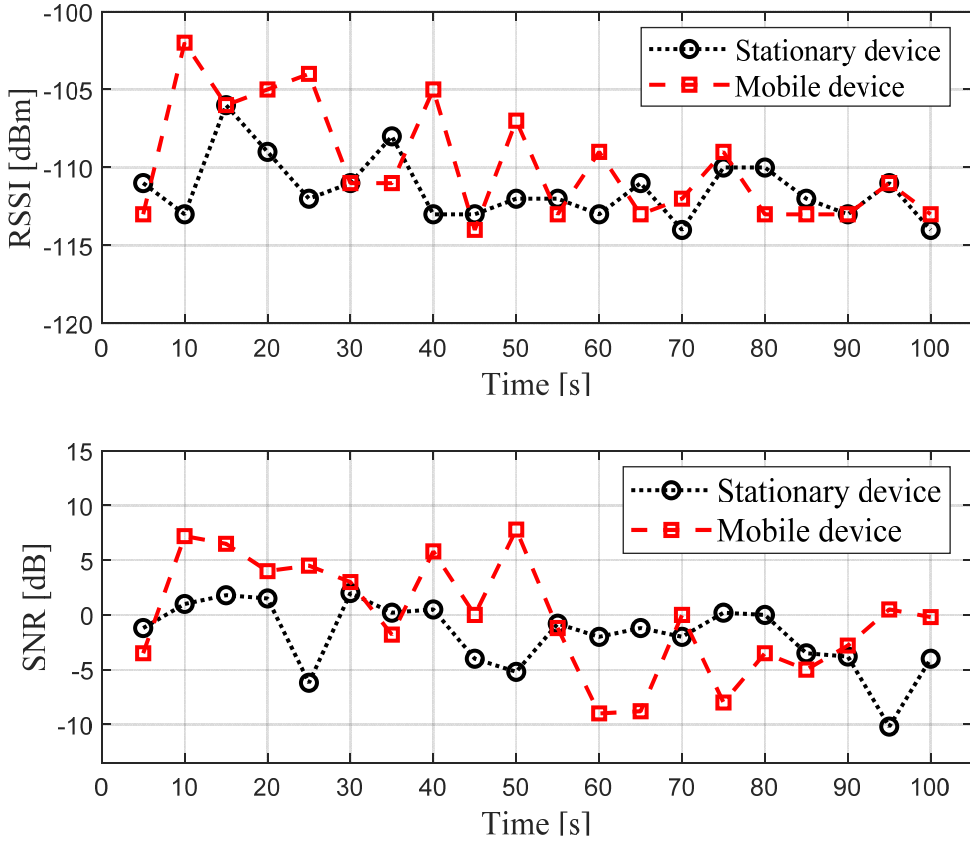


그림 7. 단말의 정지 및 모바일 상태에 대한 RSSI와 SNR 값 비교1

그림 8의 시나리오에서 평균 RSSI 값이 정지 상태와 모바일 상태에서 -112.5 dBm으로 동일하게 나타나며, 실제로 모바일 상태에서의 SNR 분산치가 더 낮은 결과를 보였다. 이러한 결과는 이는 장거리 무선 채널 환경의 복잡성을 반영한다.

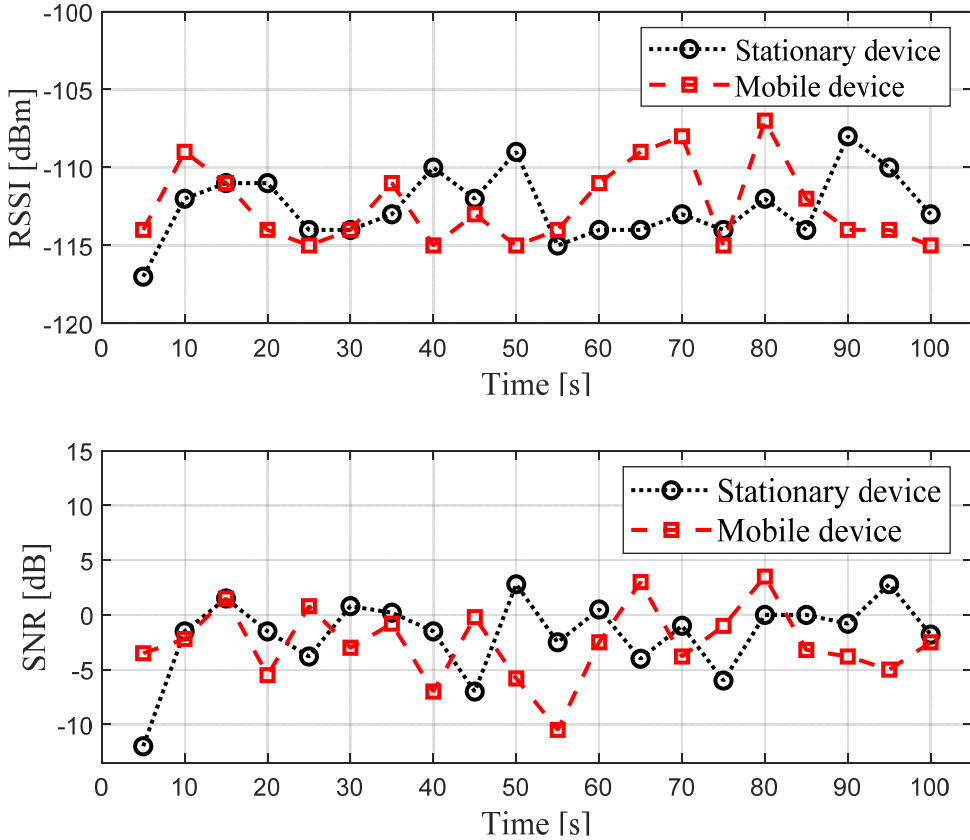


그림 8. 단말의 정지 및 모바일 상태에 대한 RSSI와 SNR 값 비교2

무선 통신 환경은 건물이나 사람에 의한 비가시 조건, 차량과 같은 이동체에 의한 신호 변동성 등과 같은 다양한 요소에 의해 영향을 받는다. 이러한 환경적 요인들은 신호 강도와 품질에 예측하기 어려운 변화를 일으키며, 이는 전통적인 통계 알고리즘으로는 단말의 정지 상태와 모바일 상태를 정확하게 분류하는 것이 어렵다는 것을 의미한다 [24]. 이에 따라, 본 연구는 딥러닝 모델을 활용하여 단말의 모바일 상태와 정지 상태에 따른 신호 데이터의 패턴을 학습하고, 이를 바탕으로 단말의 상태를 보다 정확하게 예측할 수 있는 모델을 개발한다.

3. 딥러닝 모델 학습을 위한 시퀀스 데이터 생성

모델의 학습 과정에 앞서 수행되는 데이터 전처리 단계는 모델의 예측 정확도와 성능에 결정적인 영향을 미친다. 본 연구에서는 LoRaWAN 시스템의 핵심 구성 요소인 단말, 게이트웨이, 네트워크 서버 및 어플리케이션 서버를 통해 수집된 LoRa 신호 데이터를 대상으로 전처리를 진행하였다. 그림 5에 설명된 환경에서 약 5일 동안 6,000개의 실시간으로 수집된 데이터는 복잡한 실제 환경에서의 단말의 모바일 상태를 식별하는 데 사용된다. 수집된 데이터는 순차적으로 기록되며, 이를 효과적으로 학습 가능한 시퀀스 데이터 (\mathbf{X})로 변환 과정을 거친다. (2), (3)과 같이 변환 과정에서 시퀀스의 길이 (*Sequence length*)와 단말의 상태를 각각 n , \mathbf{Y} 로 나타낸다.

$$\mathbf{X} = \begin{bmatrix} RSSI_1 & RSSI_2 & \dots & RSSI_n \\ SNR_1 & SNR_2 & \dots & SNR_n \end{bmatrix}, \quad (2)$$

$$\mathbf{m} = [m_1 \ m_2 \ \dots \ m_n]. \quad (3)$$

실제 환경에서 수집한 데이터를 시퀀스 데이터로 변환하는 과정은 몇 가지 문제점을 내포한다. 첫 번째 문제는 패킷 데이터 누락이다. 단말이 송신한 업링크 패킷이 게이트웨이에서 수신되지 않았을 경우 데이터가 끊기고 누락될 수 있다. 이는 단말의 모바일 상태 또는 무선 채널 환경의 급변으로 발생할 수 있다. 이러한 데이터 누락을 처리하기 위해, 본 연구에서는 누락된 데이터를 '0'으로 표기하여 시퀀스의 연속성을 유지하였다. 이 접근법은 채널 환경이 좋지 않음을 모델에게 표시하고, 더 나은 분류 성능을 달성하는 데 도움을 준다. 또한, 시퀀스 데이터가 끊기지 않으므로 실시간 딥러닝 모델 추론을 가능케 한다.

두 번째 문제는 하나의 시퀀스 데이터 내에 단말의 다양한 상태가 혼재하는 경우이다. 단말의 상태는 (4)에 제시된 시퀀스 데이터에 기반하여 결정된다. 시퀀스 데이터의 *mobility*는 정지 상태를 '0', 모바일 상태를 '1'으로 할당하며, 벡터 \mathbf{m} 의 요소의 합이 '0'이면 정지 상태로, '1'이 아닐 경우 모바일 상태로 분류한다. 이는 단말 대부분의 상태가 정지이더라도 순간적인 *mobility* 감지가 채널 상황의 급격한 변화를 야기할 수 있기 때문에, 패킷 손실을 줄이기 위함이다.

$$mobility = \begin{cases} 0 & \text{if } \sum_{i=1}^n m_i = 0 \\ 1 & \text{otherwise} \end{cases} \quad (4)$$

4. 시계열 딥러닝 모델 학습

본 논문에서는 시계열 데이터 학습에 적합한 LSTM (Long Short-Term Memory) 와 GRU (Gated Recurrent Unit)를 채택하였다. 그림 9의 LSTM의 핵심 구성 요소는 Cell State와 Hidden State 이다. Cell State는 시간에 걸쳐 정보를 유지하고, 필요에 따라서는 오랜 시간 동안 정보를 보존하거나 버릴 수 있는 능력을 가진다 [19]. Hidden State는 현재 타임 스텝의 출력을 결정하며, 다음 타임 스텝에 전달될 정보를 담고 있다. LSTM의 세 가지 주요 게이트인 Forget Gate, Input Gate, 그리고 Output Gate는 네트워크 내에서 정보의 흐름을 세밀하게 조절한다. Forget Gate는 Cell State의 정보 중 불필요한 부분을 제거하고, Input Gate는 새로운 정보를 Cell State에 추가하는 역할을 한다. Output Gate는 Hidden State를 통해 어떤 정보를 출력할지를 결정한다.

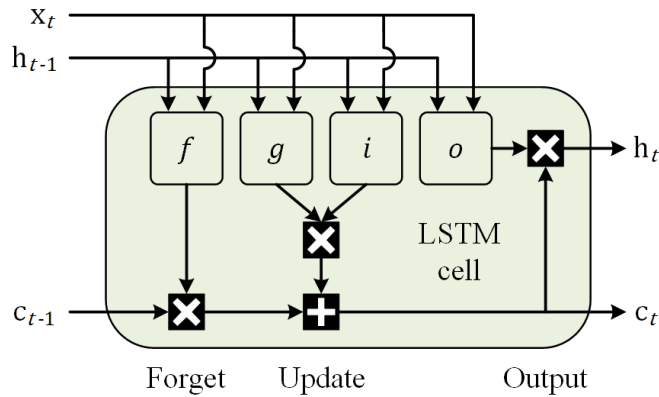


그림 9. LSTM 신경망 구조

그림 10의 GRU는 LSTM의 복잡한 구조를 단순화하면서도 장기 의존성을 학습할 수 있다 [20]. GRU는 LSTM의 Forget Gate와 Input Gate를 하나의 Update Gate로 통합하고, Cell State와 Hidden State를 하나의 Hidden State로 결합함으로써 모델의 복잡성을 감소 시킨 모델이다.

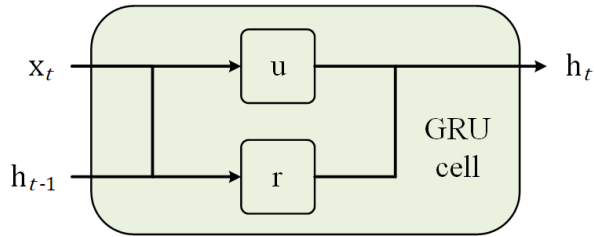


그림 10. GRU 신경망 구조

수집된 데이터의 *DR* 별 업링크 패킷 개수는 표 2와 같다. 학습 데이터, 검증 데이터, 테스트 데이터는 각각 80%, 10%, 10%로 나뉘며, 전처리가 완료된 학습 데이터는 다양한 시퀀스 길이(5, 10, 15, 20)에 대해 그림 12에 정의된 모델과 표 3의 하이퍼 파라미터를 통해 학습되었다. 모델의 입력 데이터는 시퀀스 길이와 입력 차원이 2인 *RSSI* 및 *SNR* 값으로 구성된다. 이러한 입력 데이터는 LSTM 및 GRU 레이어를 통과하여 전처리 과정을 거친 시계열 데이터를 학습시키고, 25%의 **dropout** 레이어를 통해 오버피팅을 방지하는 동시에, 완전 연결 레이어 (Fully Connected Layer)를 통해 모든 뉴런이 이전 레이어의 모든 뉴런과 연결되어 복잡한 패턴을 인식할 수 있도록 한다 [21]. 마지막으로 **softmax** 레이어는 다중 분류 문제에서의 확률 분포를 생성하여 최종적인 *mobility* 분류 결정을 내린다.

표 2. DR 별 수집된 업링크 패킷 수

DR	Uplink packet count
5	821
4	910
3	982
2	1017
1	1068
0	1124

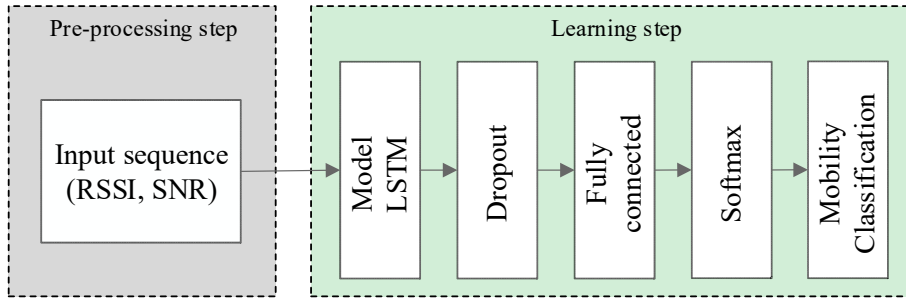


그림 11. 제안하는 시스템의 딥러닝 모델 구조

표 3. 딥러닝 학습 모델을 위한 하이퍼 파라미터

Parameter	Value
Input features	2
Epoch	200
Dropout	25%
Learning Rate	0.001
Hidden Unit	200
Optimizer	Adam
Loss Function	Cross-entropy

표 4는 시퀀스 길이에 따른 DNN, LSTM, GRU 모델의 성능을 정확도, 정밀도, 재현도 및 F1 Score를 기준으로 비교한다. DNN은 시퀀스 데이터를 학습시킬 수 없으므로, 평활화를 통해 입력한다 [18]. 그림 11의 시계열 Model을 Fully Connected Layer로 대체했으며, Hidden Unit의 개수는 200으로 설정했다. 데이터는 시퀀스 길이가 20일 때 모델이 가장 높은 분류 성능을 보이며, 모델에 관계없이 시퀀스 길이가 길어질수록 분류 성능이 높아지는 것을 확인하였다. 그러나 LoRaWAN 어플리케이션의 경우, 처음 단말이 동작을 하고 나서 시계열 데이터 구성을 위해 데이터를 수집하는 시간이 길어지는 문제가 있기 때문에 적절한 시퀀스 길이의 결정은 효율성과 효과성의 균형을 맞추는데 중요하다.

표 4의 모델10은 그림 12의 혼동 행렬은 테스트 데이터에 대한 단말의 모바일 및 정지 상태를 식별하는 능력을 보여준다. 그림 12에서 행은 예측 레이블 (Output label)에 해당하고 열은 실제 레이블 (Target label)에 해당한다. 주대각선 상에 있는 셀은 올바르게 분류된 관측값에 해당하며, 주대각선 바깥의 셀은 오분류된 관측값에 해당한다.

각 셀에는 관측값의 개수와 관측값의 총 개수의 비율을 모두 표시한다. 오른쪽 끝에 있는 열에는 각각의 레이블에 속한다고 예측된 표본 중 바르게 분류된 예와 오분류된 예의 백분율을 표시한다. 이 메트릭은 정밀도 (Precision)이다. 하단에 있는 행에는 올바르게 분류된 레이블과 잘못 분류된 레이블에 속하는 모든 표본의 백분율을 표시한다. 이러한 메트릭은 재현율 (Recall)이다. 오른쪽 하단에 있는 셀에는 전반적인 정확도 (Accuracy)를 표시한다.

표 4. 모델과 시퀀스 길이에 따른 모델 성능 비교

Model	Network	Sequence length (<i>n</i>)	Accuracy	Precision	Recall	F1 Score
1	DNN	5	43.2%	43.6%	43.3%	43.3%
2		10	61.5%	61.2%	61.2%	61.3%
3		15	70.3%	70.3%	70.4%	70.3%
4		20	79.1%	79.2%	79.1%	79.2%
5	GRU	5	68.2%	68.2%	68.2%	68.2%
6		10	90.2%	90.3%	89.9%	90.1%
7		15	95.3%	95.5%	95.2%	95.3%
8		20	99.0%	98.7%	99.2%	98.9%
9	LSTM	5	68.8%	68.9%	68.8%	68.8%
10		10	92.3%	92.5%	92.0%	92.2%
11		15	98.1%	98.1%	98.1%	98.1%
12		20	99.0%	98.8%	99.2%	99.0%

그림 12에서 1행 2열의 5.2% 오류는 딥러닝 모델이 정지 상태의 단말을 모바일 상태로 잘못 분류한 경우이며, 이는 업링크 패킷에 대한 PSR (Packet Success Ratio) 관점에서 성능 향상을 유도한다. 이는 제안된 시스템이 모바일 상태인 “1”로 예측할 경우, 채널의 상황이 불확실할 때의 시나리오를 고려하여 통신의 신뢰성을 향상시키고, LoRaWAN 네트워크에서 연결성을 보장하는 전략을 사용함을 의미한다.

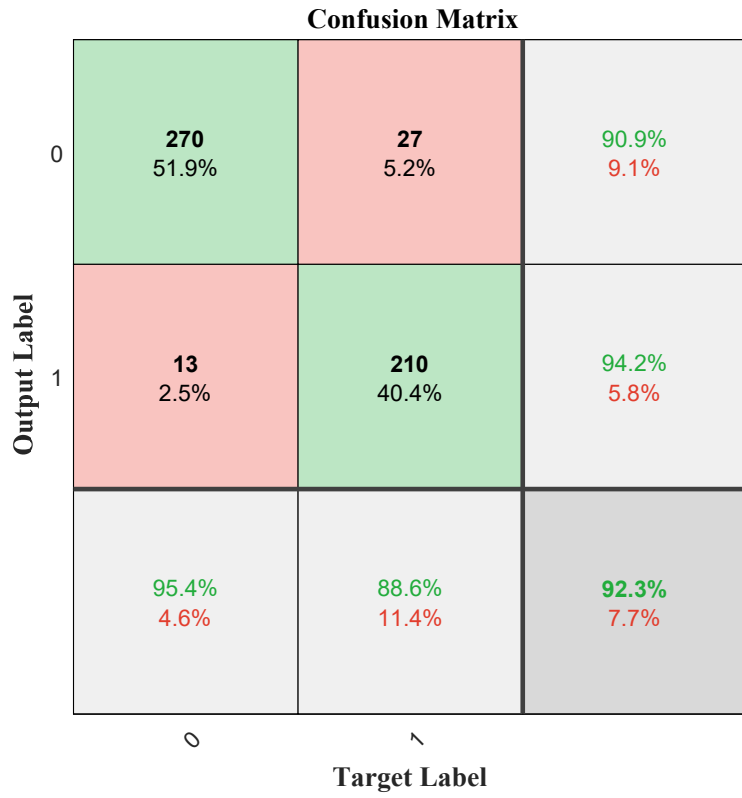


그림 12. 모델 10의 mobility 분류 성능

C. 제안 시스템의 알고리즘 동작 및 구성

본 연구는 사전 학습된 모델을 활용해 MI-ADR이 설계되었다. 해당 알고리즘은 동적 네트워크 환경에서 단말의 통신 효율성을 극대화하는 것을 목표로 한다. MI-ADR은 네트워크 서버와 단말에 각각 적용된다. 그림 13에서 보여지는 바와 같이, 서버 측 MI-ADR은 Preprocessing 과정에서 단말로부터 수신한 업링크 패킷들을 시계열 데이터로 구성하여 딥러닝 모델에 입력한다. 패킷이 n 개 축적되면, 이 데이터는 AI Inference를 통해 식별된 단말의 상태를 예측하고 단말의 Mobility에 따라 다운링크를 송신한다.

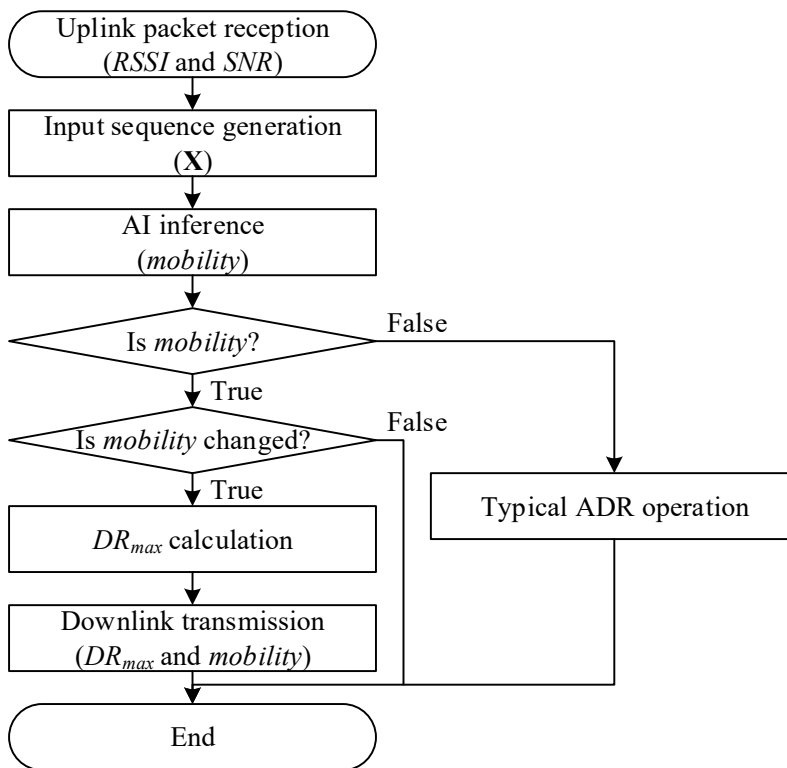


그림 13. 네트워크 서버 측 MI-ADR

서버 측 MI-ADR은 예측된 상태가 이전 상태와 다를 때에만 단말에 다운링크 패킷을 송신한다. 이는 네트워크 트래픽을 최소화하고, 불필요한 패킷 소모를 방지하는 효율적인 통신 전략이다. 만약 예측된 상태가 이전과 동일하다면, 네트워크 서버는 다운링크를 송신하지 않는다. 만약, 단말의 ADR이 활성화됐다면, 단말은 ADR로 동작할 수

있게 SNR_{margin} 을 계산하여 단계별로 DR 을 조정한다.

$$DR_{schedule} = [DR_0 DR_1 \dots DR_{max}]. \quad (5)$$

그림 14의 단말 측 MI-ADR은 서버로부터 받은 예측된 모바일 상태 여부에 따라 다르게 동작한다. 정지 상태로 예측되면, 단말은 ADR을 활성화한다. 이는 정적인 상황에서 기존 ADR을 그대로 활용한다. 그러나 모바일 상태로 예측되면, 네트워크 서버로부터 수신받은 DR_{max} 이 갱신된다. DR_{max} 은 네트워크 서버에서 SNR_{margin} 을 통해 단계별로 DR 을 최대로 조정한 값으로, 이는 단말의 현 상황에서 통신할 수 있는 최대 DR 임을 의미한다. 단말이 지속적으로 모바일 상태로 식별되면, (5)와 같이 DR_0 으로 시작해서 DR_{max} 까지 점진적으로 상향시키고, DR_{max} 에 도달했다면 다시 DR_0 으로 복귀하여 이를 반복하며 업링크 패킷을 송신한다. 이를 시퀀스 길이만큼 반복했다면, 네트워크 서버로부터 수신한 DR_{max} 을 다시 설정한 후, 반복한다. 한 번 설정된 DR 스케줄을 변경하지 않는 BADR과 달리 MI-ADR은 단말의 모바일 상태 여부에 따라 주기적으로 DR 스케줄을 변경한다. 모바일 상태인 단말이 DR_{max} 을 사용함으로써 채널 상태를 예측할 수 없는 환경에서 전력 소모를 최소화하고, DR_{max} 보다 낮은 DR 들을 사용함으로써 더 안정적인 통신을 유지할 수 있도록 하는 조치이다.

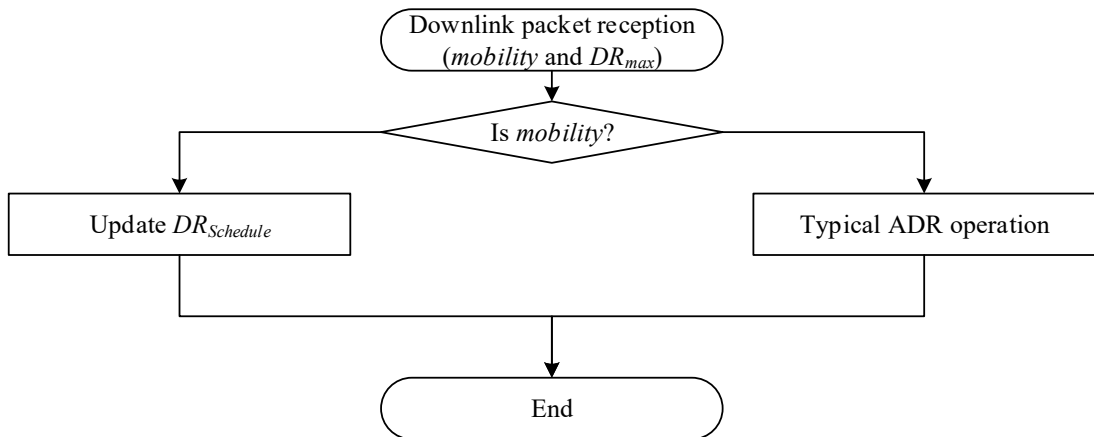


그림 14. 단말 측 MI-ADR

IV. 성능 비교 및 평가

A. 실험 환경 및 시나리오

실험 환경을 위한 LoRaWAN 구성은 서버, 게이트웨이, 단말로 구성되었다. 본 논문에서 제안하는 LoRaWAN 서버는 ChirpStack, 오픈소스로 개발된 설치형 서버를 기반으로 한다. ChirpStack은 LoRaWAN 사양 1.2에 호환되는 네트워크 서버를 제공한다. 시스템 사양은 표 5에 자세히 제시한다 [22].

표 5. LoRaWAN 네트워크 서버 구성

Category	Configuration
CPU	Intel i7 3770
RAM	DDR3 8GB
SSD	128GB
Operating System	Ubuntu 20.04 LTS
Open Source	ChirpStack v3

게이트웨이는 그림 15와 같이 RAK wireless의 RAK2245 모듈을 선택하여 구현하였다 [23]-[24]. RAK2245 모듈은 Raspberry Pi 3B+와 같은 싱글보드 컴퓨터를 통해 게이트웨이로 구현된다. Raspberry Pi 3B+는 그림 15에서 확인할 수 있는 것처럼, RAK2245 모듈과 함께 작동하여 게이트웨이의 호스트 컴퓨터 역할을 담당한다. 이 호스트 컴퓨터는 LoRa 패킷을 처리하고, 네트워크 서버로의 패킷 중계를 위한 인터페이스 역할을 수행한다. 게이트웨이가 네트워크의 중심 역할을 수행할 수 있도록 한다. 그림 16은 그림 15의 게이트웨이에 연결된 실외 안테나 사진이며, 본 연구의 학습 데이터 수집 및 추론을 위해 활용된 실험 환경 구성 중 하나이다.

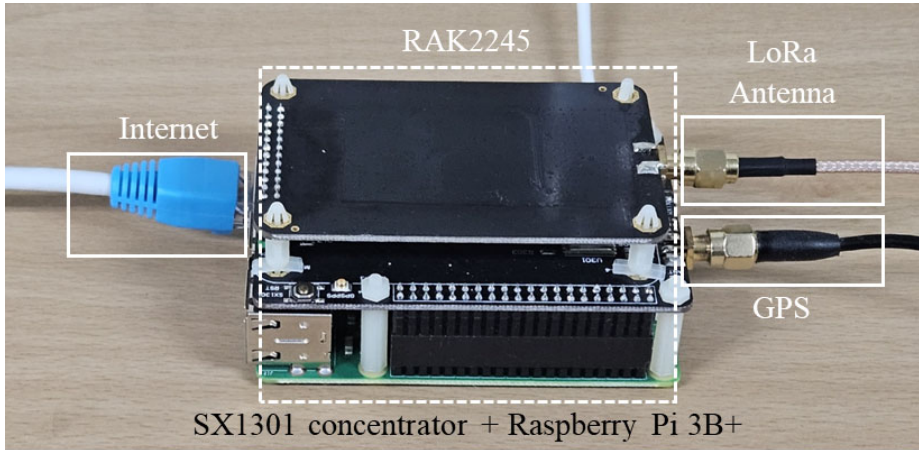


그림 15. LoRaWAN 게이트웨이 구성



그림 16. 게이트웨이의 실외 안테나

단말의 소프트웨어 구조는 다양한 기술 스택을 포함하고 있다. 여기에는 LoRaMAC-Node, Beelan LoRaWAN, 그리고 Mbed OS가 포함되며, 본 연구는 특히 ARM Mbed OS 6을 채택하였다. Mbed OS는 실시간 운영 시스템에 기반을 두고 있어, IoT 통신 어플리케이션의 구현에 매우 유리하다. 이 OS는 저전력 작업 스케줄링과 다양한 커넥티비티 스택을 제공하는데, 이는 특히 IoT 어플리케이션 구현 측면에서 중요한 이점을 가져온다. 단말의 하드웨어 구성은 국내 주파수 밴드를 지원하는 Semtech의 SX1272MB2DAS 모듈과 STM Nucleo 평가 보드를 사용하여 구현된다 [25]-[26]. 그림 17에서 왼쪽부터 ADR, BADR, MI-ADR이 탑재된 단말들이며, 실험 중 각 단말에서의 채널 충돌 방지를 위해 기본 채널이 아닌 922.7MHz, 922.9MHz, 923.1MHz의 주파수를 각각 할당하였다. 이는 LoRaWAN 네트워크의 가입 절차에 사용되는 기본 채널과의 간섭을 방지하고, 모든 단말이 동일한 실험 조건에서 작동하도록 하기 위한 조치이다. 동일한 조건들로 실험하기 위해 동시에 단말이 업링크 패킷을 게이트웨이로 송신하며, 실험을 하면서 업링크 패킷을 확인하기 위해 패킷 카운트 확인용 새그먼트를 장착했다.

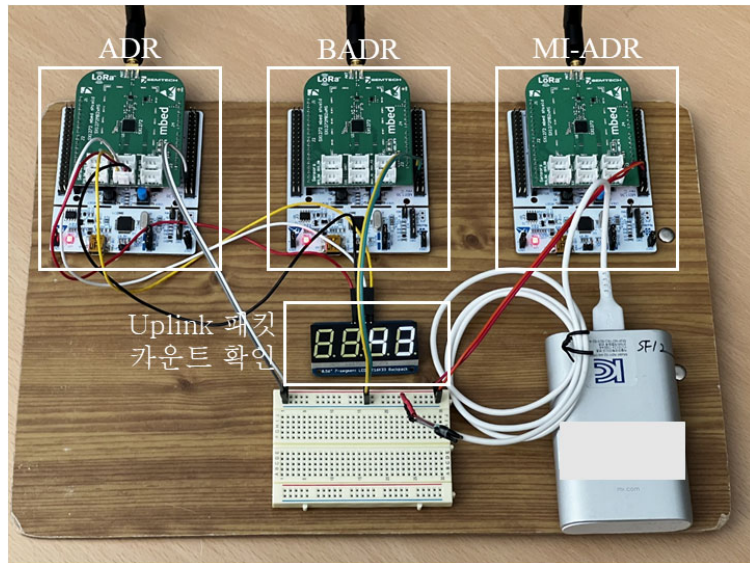


그림 17. 실험을 위한 실험 보드(ADR, BADR, MI-ADR)

실험 시나리오는 조선대학교 캠퍼스 내에서 실제 환경에서 이루어졌다. 세 개의 LoRa 단말과 하나의 게이트웨이로 구성된 LoRaWAN 네트워크를 통해 실험이 진행되었다. 모든 단말은 Class A에 속하며, 페이로드 크기는 2 바이트로 설정하였다. 업링크 송신 주기는 5초로 설정하여 각 단말에서 일정한 간격으로 패킷이 생성되도록 하였다. 이는 네트워크의 부하를 균일하게 분산시키고, 각 기법의 성능을 정확하게 비교하는데 필수적인 조건이다. 표 6에 제시된 파라미터는 이러한 실험 환경을 구축하기 위한 기초 데이터로 사용되었다 [27]. *TP*는 통제된 실험을 위해 6 dBm으로 설정되었다 [29]. ADR 기법이 *DR*과 *TP*를 조절하는 동작을 가지고 있기 때문이다. 이는 ADR 알고리즘의 성능을 *DR*의 변화에만 초점을 맞추어 평가하는 데 도움을 준다.

표 6. 실험 환경을 위한 LoRaWAN 파라미터

Parameter	Value
Mac protocol version	1.0.2
Class	A
Frequency region	KR-920
Join type	ABP
Transmit power	6 [dBm]
Transmit interval	5 [s]
Channels	922.7, 922.9, 923.1 [MHz]

실험은 그림 18에 나타난 경로를 따라 진행되었으며, 주요 랜드마크인 버스 정류장에서 출발하여 국제관에 이르렀다가 다시 복귀하는 경로로 이루어졌다. 게이트웨이로부터 출발지와 반환지점의 거리는 각각 228.3m, 469.7m이며, 출발지와 반환지점의 거리는 317.5m이다.

본 실험은 정류장에서 정지 상태로 대기 후, 이동을 시작한다. 해오름관을 지나 중앙도서관과 국제관 사이에 정지 후, 경로를 따라 이동한다. 반환지점인 1.8극장에 도착하면 다시 정지 상태로 대기한다. 반환지점에서 똑같은 정지 지점에서 이동과 정지를 반복하면서 도착 지점인 정류장에 복귀하는 시나리오이다. 이와 동일한 방식으로 총 2번의 실험을 진행하였다.

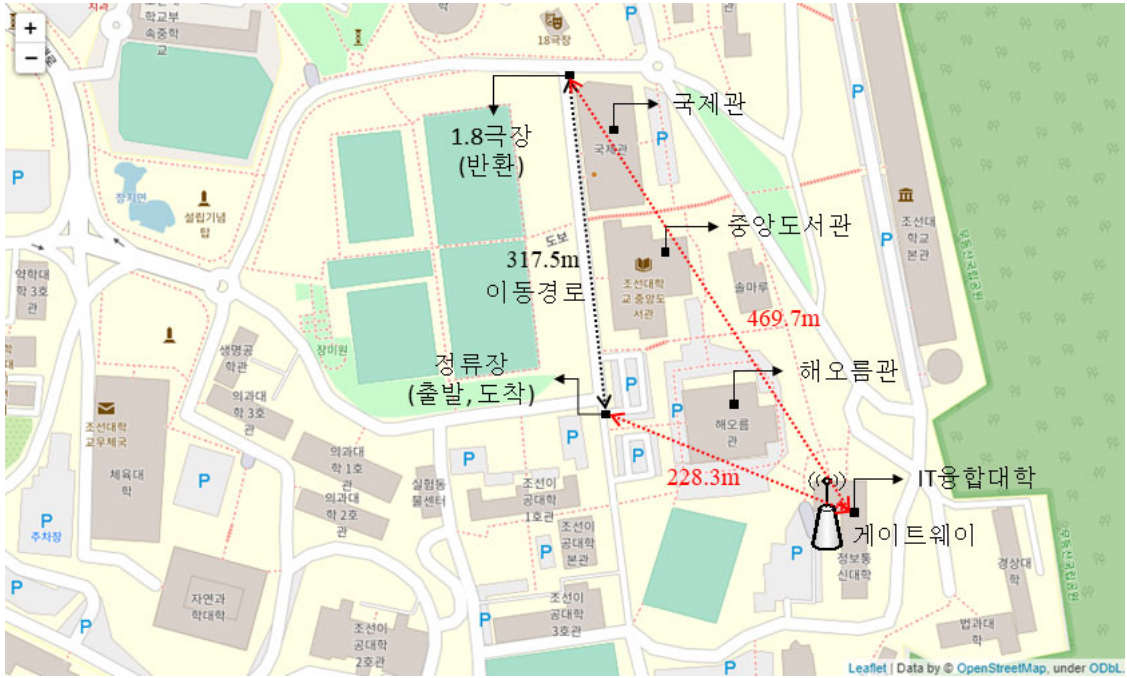


그림 18. 단말의 모바일과 정지 상태가 포함된 실험 시나리오

그림 19는 시작 지점인 버스 정류장과 단말이 목적지로 향하는 방향, 그리고 게이트웨이의 위치를 촬영한 실제 환경을 나타낸다. 그림 20은 도착 지점인 국제관에서 단말의 시점을 보여주며, 건물에 의해 게이트웨이가 시야에서 가려진 상태를 보여준다. 이는 실제 실험 환경이 LoRaWAN 통신의 가시와 비가시성 조건을 모두 포함하고 있음을 시사한다 [28]-[29].

본 실험의 결과는 기존의 ADR 및 BADR과 본 논문에서 제안하는 MI-ADR의 성능 평가를 통해 MI-ADR이 실제 네트워크 환경에서 LoRaWAN의 통신 효율성과 신뢰성을 어떻게 향상시키는지에 대한 실증적인 증거를 제시한다.



그림 19. 실험 시나리오 출발 지점



그림 20. 실험 시나리오 반환 지점

B. 실험 결과 및 성능 평가

본 논문에서 제안된 MI-ADR 알고리즘의 성능을 기존 ADR과 BADR 기법과 비교하고 평가하기 위해, 조선대학교 캠퍼스에서 수행된 실험 결과를 분석하였다. 실험은 각 기법의 PSR 성능, DR 분포, 모델 성능을 조사하여 동일한 통신 환경에서의 전송 효율성을 비교하는 방식을 택했다.

총 173개의 업링크 패킷 송신하는 시나리오에서 그림 21, 24의 PSR 성능에서 MI-ADR, BADR, ADR 순으로 각 성능을 보여준다. MI-ADR은 항상 94% 이상인 높은 PSR을 달성했으며, 이 데이터는 MI-ADR이 기존 ADR들에 비해 현저히 높은 PSR을 달성함을 명확하게 보여주고 있다.

그림 22, 25의 DR 분포에서는 시간에 따른 단말의 DR 변화를 보여준다. ADR의 경우, 두 실험 동일하게 대부분의 DR이 DR5로 나타났다. 이는 ADR이 최근 20개의 패킷으로 계산된 SNR_{margin} 을 통해 단계별로 천천히 DR을 결정하기 때문에 급변하는 채널 환경에 적응하지 못하고, 결과적으로 DR을 낮추지 못한 사례가 관찰되었다. 400~600초 사이 구간에서 DR5로 유지하기 때문에 대부분의 패킷이 손실된 것을 확인할 수 있다. 이 구간은 그림 18에서 국제관과 1.8극장 사이 구간이다. 이는 ADR이 모바일 상태에서 성능이 저하됨을 확인할 수 있다. BADR의 경우, ADR과 다르게 400~600초 사이 구간에 DR0과 DR2로 설정된 패킷들을 확인할 수 있다. 모바일 상태의 경우, 채널 상황을 예측할 수 없지만, 사전에 설정한 낮은 DR들로 인해 ADR과 다르게 완전히 통신이 끊기는 문제를 방지할 수 있었다. 하지만 0~100초 구간에 불필요하게 DR0과 DR2로 설정된 패킷들을 수신된 것을 확인할 수 있다. 이 구간은 해오름관과 버스 정류장 사이이며, DR5로 충분히 통신할 수 있는 거리이다. 이는 게이트웨이와 근거리에 위치하지만, 사전에 설정된 낮은 DR로 인해 단말의 전력 소모가 불필요하게 낭비되는 문제를 확인할 수 있다. MI-ADR의 경우, 0~100초 구간에서는 DR5로 유지하여 전력 효율성을 높이고, 게이트웨이와 거리가 멀고 채널 환경이 좋지 못한 구간인 400~600초 구간은 DR을 유기적으로 제어하여 통신을 유지 시키는 것을 확인할 수 있다.

그림 23, 26의 mobility 식별 정확도는 평균 72.4%로 측정되었다. 이는 학습 단계에서 진행한 92.3%에 비해 비교적 낮은 수치이다. 하지만 450~500초 사이는 단말이 정지

상태를 모바일 상태로 식별한 것을 보여준다. 이는 PSR 관점에서 통신 성능을 향상시키는 역할을 한다. 채널 상황이 불안정함을 인지하고 DR을 사전에 설정하여 통신이 끊기는 것을 방지한다. 또한, 단말이 초기화되는 구간인 0~50초 구간은 AI 모델 추론을 위한 시퀀스 데이터의 길이를 충족시키지 못하였기 때문에, *mobility*에 관한 레이블 값은 존재하지 않는다.

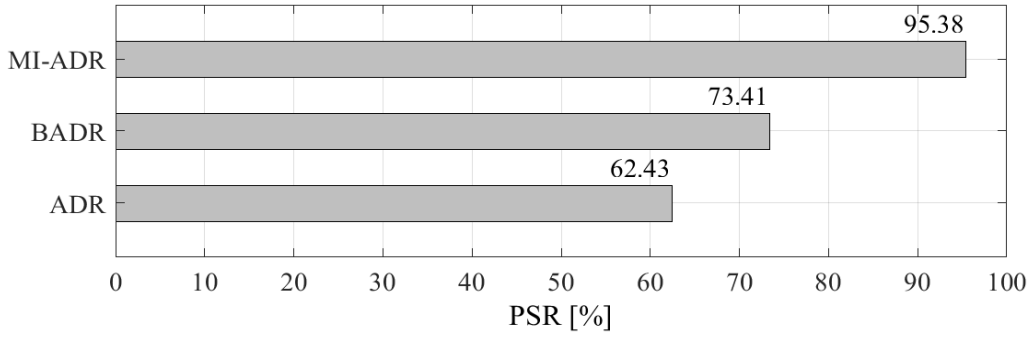


그림 21. 첫 번째 실험의 PSR 성능

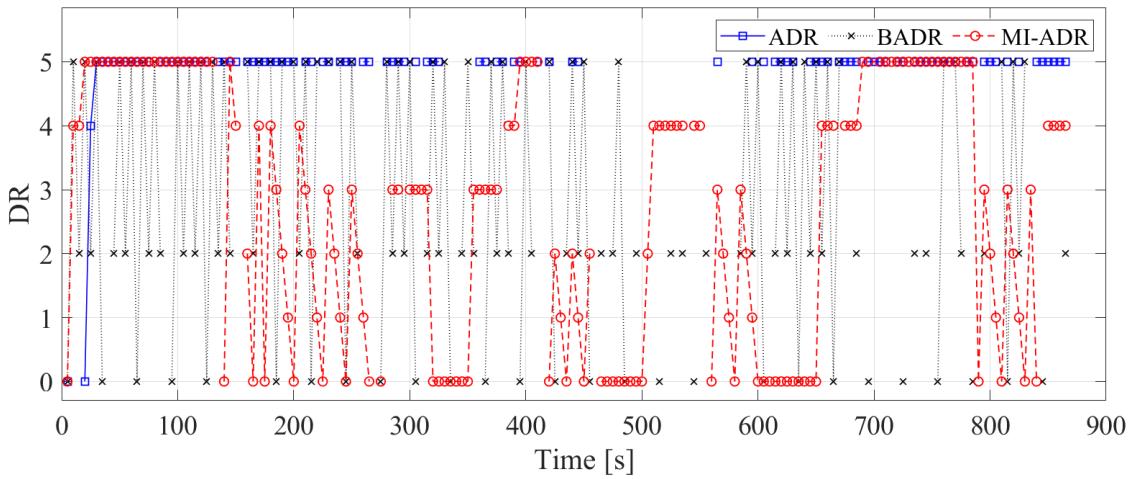


그림 22. 첫 번째 실험의 DR 분포

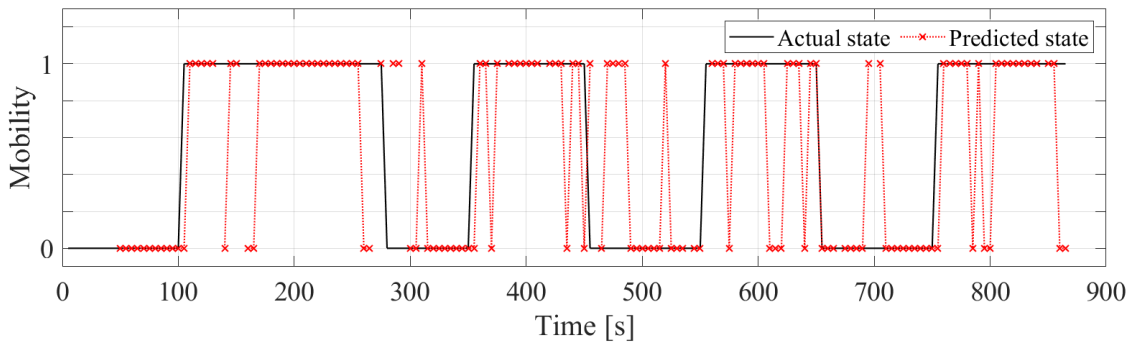


그림 23. 첫 번째 실험의 mobility 식별 성능

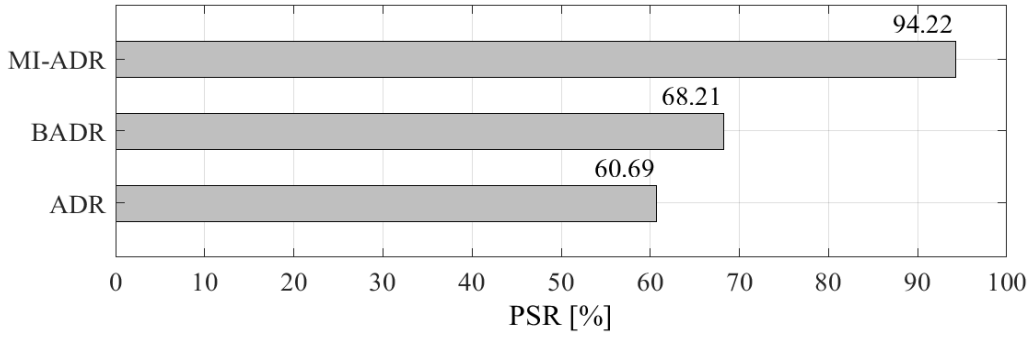


그림 24. 두 번째 실험의 PSR 성능

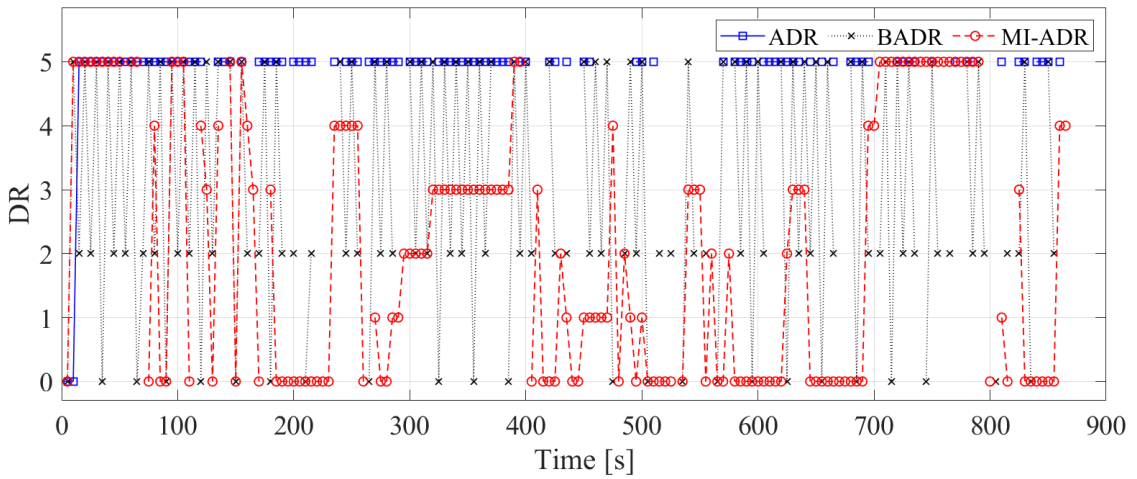


그림 25. 두 번째 실험의 DR 분포

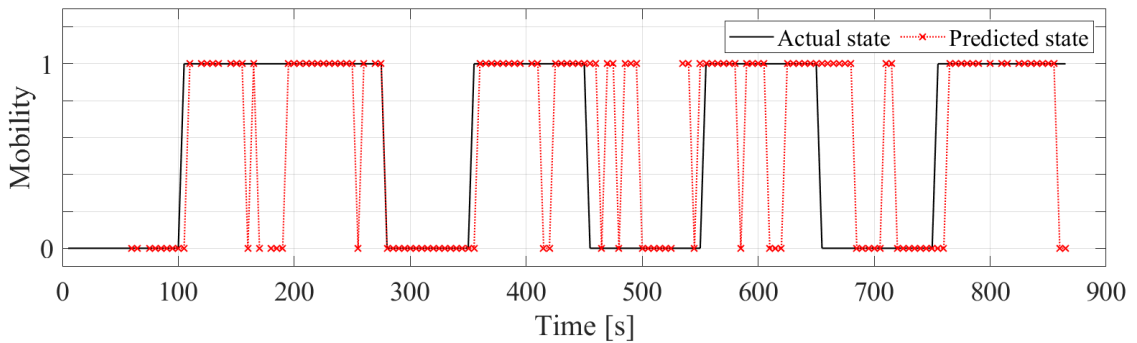


그림 26. 두 번째 실험의 mobility 식별 성능

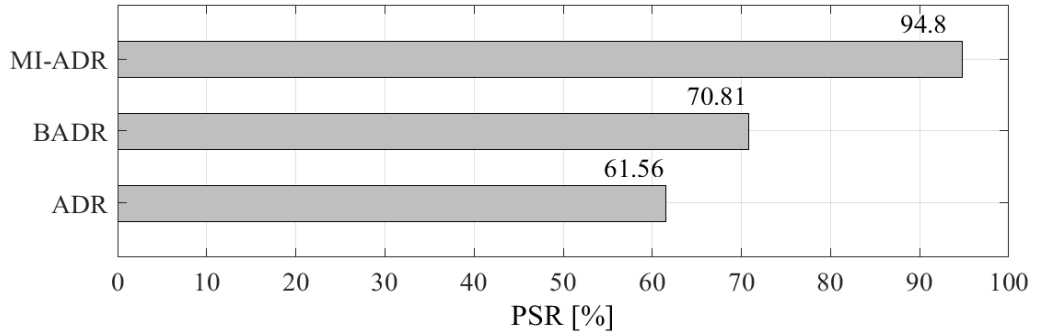


그림 27. 전체 실험의 PSR 성능

그림 27은 두 실험 평균에 대한 PSR 성능 지표이며, MI-ADR은 BADR 대비 33.8%, ADR 대비 53.9% 향상률을 보여준다. 이러한 결과는 MI-ADR 알고리즘이 실제 네트워크 환경에서의 동적 상황에 효과적으로 대응하며, 특히 모바일 상태의 단말에 대해 우수한 DR 결정 능력을 보유하고 있음을 보이는 결과이다. LoRaWAN 네트워크의 자원 최적화 및 통신 효율성에서 유의미한 개선을 제공함을 명확하게 보여준다. 이러한 분석은 MI-ADR 알고리즘이 LoRaWAN 기술을 활용하는 다양한 IoT 애플리케이션에서의 신뢰성 있는 데이터 전송 및 네트워크 자원 관리에 기여할 수 있음을 시사한다.

V. 결론 및 향후 연구과제

A. 결론

본 연구는 LoRaWAN 네트워크의 자원 할당 문제에 대해 딥러닝 모델을 활용한 모바일 상태 식별과 이를 활용한 MI-ADR 알고리즘을 통한 효과적인 해결 방안을 제시하였다. 이 알고리즘은 특히 위치 기반 센서 없이 기존 동적 환경에서 우수한 성능을 발휘하며, 단말의 모바일 상태를 정확하게 추론하는 능력을 입증하였다. 조선대학교 캠퍼스에서 구축된 실제 실험 보드를 사용하여 수행된 실험을 통해, MI-ADR 알고리즘은 기존의 ADR 및 BADR 대비 각각 53.9%, 33.8%라는 성능 향상을 보였다.

실시간 시계열 딥러닝 모델을 이용한 분석에서 MI-ADR은 단말의 정지 상태와 모바일 상태를 식별함에 있어 72.4%라는 정확도를 달성하였다. 이는 모바일 상태에서도 비교적 높은 PSR을 유지하는 데 기여하였으며, 네트워크의 자원을 보다 효율적으로 할당할 수 있도록 하였다. ADR 방식이 단말이 이동할 때마다 빈번한 환경 변화에 적응하지 못하는 반면, MI-ADR은 이러한 변화에 빠르게 대응하였다. BADR 방식은 단말이 정지 상태일 때 적합하지 않은 것으로 나타났으나, MI-ADR은 정적 및 모바일 환경 모두에서 유용하였다.

더욱이, 실험 결과는 MI-ADR 알고리즘이 높은 PSR을 달성하면서도 효율적인 DR을 제공하는 것을 보여주었다. 이는 IoT 어플리케이션의 다양한 요구 사항에 대응하는 데 있어 MI-ADR 알고리즘이 핵심적인 역할을 수행할 수 있음을 시사한다. 본 연구는 LoRaWAN 기술의 발전을 촉진하며, 네트워크의 효율성과 신뢰성을 극대화하는 데 중요한 기여를 한다. 나아가, 더욱 발전된 알고리즘 설계를 위한 토대를 마련하고, IoT 시스템의 효율성 및 신뢰성 향상에 필요한 핵심 기술로서의 잠재력을 보여준다.

B. 향후 연구과제

본 연구에서 도출된 MI-ADR 알고리즘의 성능은 LoRaWAN 네트워크에서의 자원 할당 방식에 대한 새로운 방향을 제시했다. 정적 및 모바일 상태를 구분함으로써 자원 할당의 효율성을 크게 개선하는 MI-ADR 알고리즘은 기존의 ADR 및 BADR 접근법에 비해 상당한 이점을 보였다. 이러한 이점은 특히 IoT 어플리케이션의 신뢰성과 데이터 전송률에 있어서 중요하며, 다양한 환경과 응용 프로그램에서의 유연성을 보장한다.

LoRaWAN 네트워크에서 자원 할당의 최적화는 사물인터넷 기술이 점차 복잡해지고 다양화됨에 따라 중대한 과제로 부각되고 있다. 본 연구에서 제안된 MI-ADR 알고리즘은 이러한 과제에 대응하기 위한 인공지능 접근법을 제시하며, 다음과 같은 몇 가지 중요한 제언을 통해 알고리즘의 효과성을 한층 더 발전시킬 수 있는 방안을 모색하고자 한다.

첫째, MI-ADR 알고리즘의 강인성을 다양한 환경에서 평가하고, 알고리즘의 적응성을 강화하기 위해 더 광범위한 조건과 잠재적 통신 장애 요소에 대한 실험이 요구된다. 둘째, 본 연구결과를 토대로, 대규모 네트워크 환경에서의 MI-ADR 알고리즘의 확장성을 검증하고, 다수의 단말과 복잡한 네트워크 토폴로지에서의 성능을 분석하여 알고리즘의 범용성을 증명해야 할 것이다. 셋째, 실시간 데이터 처리와 응답 시간을 최소화하기 위한 기법을 개발하여 긴급한 응답이 요구되는 어플리케이션의 요구를 충족시켜야 한다. 넷째, 에너지 소비를 최소화하고 네트워크 수명을 극대화하기 위해 알고리즘의 파라미터를 미세 조정하고, 저전력 운영 전략을 개발해야 할 것이다.

본 연구는 LoRaWAN 네트워크의 자원 할당 문제 해결에 기여하며, 동시에 향후 연구자들에게 중요한 방향을 제시하고 있다. 더욱 발전된 알고리즘 설계 및 네트워크 운영 전략을 개발함으로써, IoT 네트워크의 지속 가능한 성장과 발전을 도모할 수 있을 것이다.

참고 문헌

- [1] U. Raza, P. Kulkarni, and M. Sooriyabandara, “Low Power Wide Area Networks: An Overview,” *IEEE Commucation Surveys & Tutorials*, vol. 19, no. 2, pp. 855-873, Jan. 2017.
- [2] 최환석, 이우섭, “초연결 사회를 위한 oneM2M 표준화 기술 동향,” 한국통신학회지 (정보와통신), 제31권 제4호, pp. 37-43, Mar. 2014.
- [3] B. Citoni, F. Fioranelli, M. A. Imran, and Q. H. Abbasi, “Internet of things and loraw an-enabled future smart farming,” *IEEE Internet of Things Magazine*, vol. 2, no. 4, pp. 14-19, Dec. 2019.
- [4] A. Augustin, J. Yi, T. Clausen, W. M. Townsley, “A study of LoRa: Long range & low power networks for the Internet of Things,” *Sensors*, vol. 16, no. 9, Sep. 2016, Art. no. 1466.
- [5] T. T. Nguyen, H. H. Nguyen, R. Barton, and P. Grossetete, “Efficient design of chirp spread spectrum modulation for low-power wide-area networks,” *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9503-9515, Dec. 2019.
- [6] Orange, *Lora device developer guide (2016)*, Accessed on Nov. 2023. [Online]. Available: <https://developer.orange.com/wp-content/uploads/LoRa-Device-Developer-Guide-Orange.pdf>
- [7] 구성완, 류대우, “IoT환경에서 LoRa모듈의 장시간 사용을 위한 전류소모량 개선방안,” *Journal of Digital Contents Society (JDSC)*, vol. 23, No.7, pp. 1307-1311, Jul. 2022.
- [8] A. P. Murdan, and J. S. Ramphul, “LoRa-based Smart Patient Monitoring System,” in *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, vol. 15, no. 1, pp 15-21, Mar. 2023.
- [9] SK 텔레콤, SK Telecom 저전력 IoT LoRa 디바이스 기술요구사항 V1.92 (2016), Accessed on Nov. 2023. [Online]. Available: <http://gns.biztworld.co.kr/service/d5.jsp>
- [10] B. Reynders, W. Meert, and S. Pollin, “Power and spreading factor control in low power wide area networks,” in *IEEE International Conference on Communications (ICC)*, Paris, France, May 2017, pp. 1-6.

- [11] LoRa Alliance, *Lorawan Link Layer Specification v1.0.4. (2020)*, Accessed on: Nov. 2023. [Online]. Available: <https://lora-alliance.org/lorawan-for-developers>
- [12] Semtech, *Understanding the lora adaptive data rate (2019)*, Accessed on Nov. 2023. [Online]. Available: <https://lora-developers.semtech.com/library/techpapers-and-guides/understanding-adr>
- [13] R. Kufakunesu, G. P. Hancke, and A. M. Abu-Mahfouz, "A survey on adaptive data rate optimization in lorawan: Recent solutions and major challenges," *Sensors*, vol. 20, no. 18, Jul. 2020, Art. no. 5044.
- [14] K. Kousias, G. Caso, Ö. Alay, and F. Lemic, "Empirical analysis of lorawan adaptive data rate for mobile internet of things applications," in *Proc. Mobile Computing and Networking*, Los Cabos, Mexico, 2019, pp. 9–11.
- [15] A. Farhad, D.-H. Kim, and J.-Y. Pyun, "R-ARM: Retransmission-assisted resource management in lorawan for the internet of things," *IEEE Internet of Things Journal*, vol. 9, no. 10, pp. 7347-7361, May. 2021.
- [16] A. Farhad, J.-S. Yoon, and J.-Y. Pyun, "Feasibility study of the LoRaWAN blind adaptive data rate," in *International Conference on Ubiquitous and Future Networks (ICUFN)*, Jeju, South Korea, 2021, pp. 67-69.
- [17] A. Farhad, and J.-Y. Pyun, "HADR: A Hybrid Adaptive Data Rate in LoRaWAN for Internet of Things," in *ICT Express*, vol. 8, no. 2, pp. 283-289, Jun. 2022.
- [18] A. Farhad, and J.-Y. Pyun, "AI-ERA: Artificial Intelligence-Empowered Resource Allocation for LoRa-Enabled IoT Applications," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 12, pp. 11640-11652, Dec. 2023.
- [19] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, Nov. 1997.
- [20] A. Farhad, D.-H. Kim, J.-S. Yoon, and J.-Y. Pyun, "Deep Learning-Based Channel Adaptive Resource Allocation in LoRaWAN," in *International Conference on Electronics, Information, and Communication (ICEIC)*, Jeju, South Korea, 2022, pp. 1-5.
- [21] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929-1958, Jun. 2014.

- [22] O. Brocaar, *ChirpStack network server*. (2023), Accessed on: Nov. 2023. [Online]. Available: <https://www.chirpstack.io>
- [23] RAK, *RAK831 WisLink LPWAN Concentrator Datasheet*. (2023), Accessed on: Nov. 2023. [Online]. Available: <https://docs.rakwireless.com/Product-Categories/WisLink/RAK831/Datasheet/>
- [24] Semtech, *SX1301 Datasheet V2.4 (2017)*, Accessed on: Nov. 2023. [Online]. Available: <https://www.semtech.com/products/wireless-rf/lora-gateways/sx1301>
- [25] Semtech, *SX1272/3/6/7/8 LoRa Modem Design Guide*. (2019), Accessed on: Nov. 2023. [Online]. Available: <https://www.semtech.com/uploads/documents/LoraDesignGuideSTD.pdf>
- [26] ARM MBED, *Mbed OS 6.10*. (2021), Accessed on: Nov. 2023. [Online]. Available: <https://github.com/ARMmbed/mbed-os>
- [27] LoRa alliance, *LoRaWAN Regional Parameters Specification v1.0.2*. (2020), Accessed on: Jun. 2023. [Online]. Available: <https://resources.lora-alliance.org/technical-specifications/rp2-1-0-2-lorawan-regional-parameters>
- [28] 권혁, 진경복, 오창현, “NLOS 환경에서 LPWA 통신 커버리지 측정,” in *Proc. 한국정보통신학회*, Daejeon, South Korea, 2019, pp. 591-593.
- [29] A. Farhad, D.-H. Kim, and J.-Y. Pyun, “Scalability of lorawan in an urban environment: A simulation study,” in *International Conference on Ubiquitous and Future Networks (ICUFN)*, Zagreb, Croatia, July. 2019, pp. 677-681.