신경망 프레임워크와의 통합 운용을 위한 임베디드 하이브리드 AI 플랫폼 및 의사결정나무 구조 제안

마준익, 이세인, 곽준호, 김규민, 조정훈*

경북대학교, *경북대학교

macs6848@knu.ac.kr, lsin07@knu.ac.kr, junho7513@knu.ac.kr, kyumin747@knu.ac.kr, *jcho@knu.ac.kr

Proposal of an Embedded Hybrid AI Platform and Decision Tree Structure for Integration with Neural Network Frameworks

Jun Ik Ma, Se In Lee, Jun Ho Kwak, Kyu Min Kim, Jung Hun Cho*

Kyungpook National Univ., *Kyungpook National Univ.

요 약

본 논문은 자원이 제한된 MCU 환경에서 온디바이스 학습 및 추론을 지원하는 하이브리드 AI 플랫폼을 제안한다. 특히, 기존 인공신경망 프레임워크와 호환 가능한 '의사결정나무(Decision Tree) 레이어' 구현을 위해 인덱스 기반 데이터 분할 기법과 더미 노드(Dummy Node)를 활용한 정적 트리 구조화 방식을 제안한다. 이를 통해 동적 메모리 할당이 제한된 임베디드 환경에서 메모리 파편화를 방지하고, 사용자 패턴을 실시간으로 반영하는 개인화된 TinyML 서비스 구현 가능성을 제시한다.

Ⅰ. 서 론

최근 사물인터넷(IoT) 기술의 급격한 발전으로 디바이스 수가 폭발적으로 증가함에 따라, 수집된 데이터를 클라우드로 전송하지 않고 엣지(Edge) 단말에서 직접 처리하는 TinyML(Tiny Machine Learning) 기술이 핵심 트렌드로 부상하고 있다[1]. TinyML은 클라우드 전송 시 발생하는 지연 시간(Latency)과 대역폭 소모를 최소화하고, 민감한 데이터의 프라이버시를 보호할 수 있다는 강점이 있다. 그러나 TinyML이 주로 적용되는 마이크로컨트롤러(MCU) 환경은 메모리와 연산 능력이 극도로 제한되어 있어, 고성능 인공지능 모델을 그대로 적용하기에는 물리적인 한계가 따른다[2].

특히 생활 가전 분야는 가격 경쟁력 확보를 위해 고성능 프로세서 대신 저사양 MCU를 주로 채택하며, 비교적 단순한 입력 데이터를 처리한다. 그러나 사용자마다 가전제품을 사용하는 빈도나 패턴이 상이하므로, 클라우드나 고성능 서버에서 사전 학습된 고정 모델(Pre-trained Model)을 단순히 탑재하여 추론(Inference)만 수행하는 방식은 사용자별 특성을 반영하지 못해 개인화(Personalization) 성능이 저하되는 문제가 있다[3]. 진정한 사용자 맞춤형 서비스를 제공하기 위해서는 디바이스가 운용되는 현장의 데이터를 기반으로 모델을 재학습하거나 갱신할 수 있는 '온디바이스 학습(On-device Learning)' 기능이 필수적이다[4].

하지만 기존 임베디드 AI 솔루션들은 대부분 Python 기반 라이브러리에 의존하거나 고성능 하드웨어를 전제로 하고 있어, 제한된 MCU 환경에서 학습과 추론을 동시에 수행하기에는 어려움이 있다[5].

이에 본 논문에서는 자원이 제약된 임베디드 환경에서도 효율적으로 동작할 수 있는 머신러닝-인공신경망 통합 플랫폼을 제안하며, 특히 신경망의 레이어(Layer) 구조를 차용하여 의사결정나무(Decision Tree) 알고리즘을 탑재하기 위한 구조적 설계를 제시한다. 이를 통해 저사양 MCU 기반의 생활 가전 기기에서도 사용자 패턴을 실시간으로 반영하는 개인화된 인공지능 서비스를 제공하고자 한다.

Ⅱ. 기존 연구

임베디드 환경의 추론 라이브러리인 NNoM(Neural Network on Microcontroller)은 순수 C언어 기반의 계층 구조를 제공하여 MCU 최적화가 용이하다는 특징이 있다[6]. 본 연구팀은 NNoM의 유연한 레이어 설계 방식을 계승하되, 추론에 국한된 기존 기능을 확장하여 학습 기능과 머신러닝 모델인 의사결정나무를 동시에 지원하는 플랫폼을 개발하였다[7]. 본 논문은 해당 플랫폼의 핵심 모듈인 의사결정나무 레이어의 구조적 설계에 집중하여 기술한다.

Ⅲ. 본론

제안하는 플랫폼은 DNN, RNN 등 기존 신경망 레이어 사이에 의사결정나무를 자유롭게 배치할 수 있는 하이브리드 구조를 갖는다. 하드웨어 제약을 고려한 핵심 설계 전략은 다음과 같다.

3-1. 인덱스 기반 데이터 분할 및 공유 버퍼 활용 (In-place Strategy)

동적 메모리 할당이 불가능한 환경을 고려하여, 레이어 간 데이터 전달 시 공유 데이터 버퍼(Shared Data Buffer)를 사용한다.

- **In-place Sorting**: 기존 라이브러리와 달리 새로운 노드 객체를 생성하지 않고, 버퍼 내에서 분할 기준에 따라 데이터를 좌우로 재배치(In-place)하는 방식을 적용한다.

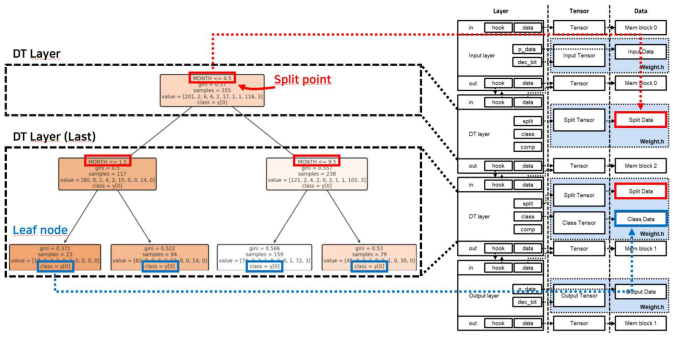
- **Index Boundary**: 분할된 데이터의 물리적 이동 없이 경계 인덱스(Split Index)만을 저장함으로써 노드를 추상화한다. 하위 레이어는 할당된 인덱스 범위 내에서만 재귀적으로 분할을 수행하여 메모리 사용량을 최소화한다.

3-2. 더미 노드(Dummy Node)를 활용한 정적 트리 구조화

학습 데이터에 따라 형태가 변하는 트리를 고정된 배열에 매핑하기 위해 더미 노드 기법을 도입한다.

- **완전 이진 트리(Complete Binary Tree) 정규화**: 특정 노드가 조기 종료(Early Stopping) 조건을 만족하더라도, 최대 깊이(Max_depth)까지 가상의 더미 노드를 생성하여 전파한다.

- 메모리 예측 가능성 확보: 이를 통해 트리는 항상 $2^{Max\_depth}-1$ 크기의 고정된 배열 구조를 갖게 된다. 이는 메모리 파편화를 원천 차단하며, 배열 인덱스 연산을 통한 O(1)의 노드 접근 성능을 보장한다.



<Figure 1. Layer-Tensor 기반의 의사결정나무 시스템 아키텍처 구성도>

Figure 1은 인덱스 기반 데이터 분할과 더미 노드 기법을 활용하여 의사결정나무의 논리적 구조를 MCU 환경에 최적화된 정적 Layer-Tensor 계층으로 매핑한 도식이다. 공유 버퍼 내에서의 In-place 데이터 재배치와 고정 배열 인덱스 연산을 통해 메모리 파편화를 방지하고 하드웨어 수준에서의 예측 가능한 추론 성능을 보장하는 구조이다.

Ⅳ. 결론

본 논문에서는 자원이 극도로 제한된 저사양 MCU 환경에서도 개인화된 인공지능 서비스를 제공하기 위해, 기존의 인공신경망(ANN)과 머신러닝(ML) 알고리즘을 단일 파이프라인에서 통합 운용할 수 있는 '학습형 하이브리드 AI 플랫폼'을 제시하고, 그 중 의사결정나무의 효율적 구조를 제안하였다.

본 연구는 동적 메모리 할당이 불가능한 임베디드 시스템의 제약을 극복하기 위해, 가변적인 트리 구조를 정적 배열 인덱싱과 더미 노드(Dummy Node) 기법을 통해 신경망의 고정 레이어(Fixed Layer) 형태로 추상화했다는 점에 의의가 있다.

향후 연구에서는 제안한 의사결정나무 구조를 실제 펌웨어로 구현하고 Scikit-learn의 표준 의사결정나무 결과와 비교 검증한 후, 실제 MCU 보드에 탑재하여 성능을 평가할 예정이다.

ACKNOWLEDGMENT

이 논문은 2025년도 정부(산업통상자원부)의 재원으로 한국산업기술진흥원의 지원(RS-2024-00415938, 2024년 산업혁신인재성장지원사업) 및 2025년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원(No.RS-2025-02216517, 재구성형 인공지능 프로세서 SW 프레임워크 기술개발)을 받아 경북대학교에서 수행된 연구 결과입니다.

참 고 문 헌

[1] R. Kallimani, K. Pai, P. Raghuwanshi, S. Iyer, and O. L. A. López, "TinyML: Tools, applications, challenges, and future research directions," Multimedia Tools and Applications, vol. 83, pp. 29015 - 29045, 2024.

[2] B. Jacob et al., "Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 2704-2713.

[3] J. S. Ren et al., "TinyOL: TinyML with Online Learning on Microcontrollers," 2021 International Joint Conference on Neural Networks (IJCNN), 2021, pp. 1-8.

[4] M. Faris et al., "On-Device Training of Machine Learning Models on Microcontrollers with a Handful of Data," in IEEE Access, vol. 9, pp. 113110-113120, 2021.

[5] L. Lai, N. Suda and V. Chandra, "CMSIS-NN: Efficient Neural Network Kernels for Arm Cortex-M CPUs," arXiv preprint arXiv:1801.06601, 2018.

[6] Jianjia Ma. (2020). A higher-level Neural Network library on Microcontrollers (NNoM) (v0.4.2). Zenodo. https://doi.org/10.5281/zenodo.4158710

[7] J. Ma, J. Kwak, and J. Cho, "Implementation of an RNN Model for Artificial Intelligence Training on an MCU-Based Platform," in Proc. of the KICS Summer Conference, 2024, pp. 479-480.