과제 2 :: Edge Device를 위한 TensorFlow Lite CNN 모델 최적화

Ⅰ. Overview

1. Objectives

- 강의 자료 5와 6에서 배운 내용을 기반으로, TensorFlow 및 TensorFlow Lite와 Arduino Nano 33 BLE를 사용하여 EMNIST 데이터셋 기반 "대문자 영문 알파벳" 손글씨 인식 모델을 최적화합니다.
- 제한된 하드웨어 리소스에서 효율적으로 동작하는 모델 개발을 통해, DL 모델의 Edge Device에서의 응용 가능성을 탐구합니다.

2. Implementation

- TensorFlow 및 TensorFlow Lite를 사용하여 영문 알파벳 손글씨 인식 모델 설계 및 Arduino 보드 포팅
- 보드에서 모델의 추론 정확도와 추론 시간을 측정하고, 다양한 최적화 기법을 적용하여 결과를 비교

Ⅱ. Tasks

Task 1. Development and hardware porting of DL model

- TensorFlow 및 TensorFlow Lite를 이용하여 대문자 영문 알파벳(Uppercase letters of Latin alphabet) 손글씨 인식 모델 설계 및 학습
- 손글씨 숫자와 영문자를 포함하는 확장된 MNIST 데이터셋인 EMNIST (Extended Modified National Institute of Standards and Technology) 데이터셋을 활용하여 과제 진행
- 모델의 크기와 성능을 하드웨어 요구 사항에 맞게 설계. 이후 설계한 모델을 Arduino Nano 33 BLE에 포팅하여 추론 정확도와 추론 시간 확인 (**추론 정확도는 Colab에서 획득, 추론 시간은 보드에서 획득**)

Task 2. Optimize inference time

- Task 1에서 개발한 모델에 대해 추론 시간을 줄이는 동시에 정확도를 유지하기 위한 최적화 기법 (예: pruning, quantization, convolution 최적화 등)을 적용
- 최적화된 모델을 Arduino Nano 33 BLE에 다시 포팅하여, **평균 추론 정확도가 0.99 이상이면서, 동시에 "평균 추론 정확도 / 평균 추론 시간" 비율이 최대가 될 수 있도록 최적화**

- **References for performing the task**

https://www.tensorflow.org/api_docs/python/tf/keras/layers
https://keras.io/api/layers/
https://ssongnote.tistory.com/13
https://truman.tistory.com/188
https://laboputer.github.io/machine-learning/2020/03/12/mnist995/

Ⅲ. Submission

1. Assignments

- 모델 최적화 및 포팅 과정**(Task 1)**이 담긴 Jupyter Notebook 파일(.ipynb) 및 Arduino 소스 코드 파일
- 추론 시간 최적화 과정**(Task 2)**이 담긴 Jupyter Notebook 파일(.ipynb) 및 Arduino 소스 코드 파일
- 결과 및 분석을 담은 보고서(PDF 형식)

  *** 보고서는 다음의 내용을 포함해야 합니다.**

  - 시뮬레이션 결과 스크린샷
  - 설계한 딥러닝 모델의 설계 및 실행 방식
  - 측정된 추론 성능 및 적용된 최적화 기법
  - 적용 방법 및 결과

2. Submission Guidelines

- 모든 파일은 팀 대표 한 명이 HW2_팀명.zip(예: HW2_3팀.zip) 형식으로 압축하여 제출
- 과제의 재제출 역시 초기 제출자가 담당 (초기 제출자의 재제출 불가 시 사전 고지 필요)
- 제출 양식 미준수 시 5% 감점 (예: 보고서 확장자 오류, 재제출 시 사전 고지 없는 제출자 변경 등)
- AjouBb를 통해서만 제출 (메일 제출 불가)

3. Delayed submission

- 지각 제출 시 매일 15% 감점, 최대 50% 감점
- 미제출 과제는 0점 처리

Ⅳ. Criteria (Total points: 100 pt.)

1. 정확성 및 완성도 [60점]

가. 모델 설계 및 학습 **[20점]**

1) 영문 알파벳 손글씨 인식 모델의 설계 및 학습 과정의 완성도
2) 모델의 정확도 및 효율성

나. 모델 최적화 및 하드웨어 포팅 **[20점]**

1) 하드웨어 요구 사항에 맞는 모델 최적화 및 성능 보완
2) Arduino Nano 33 BLE에 포팅된 모델의 성공적 실행, 추론 정확도 및 시간 평가

다. 추론 시간 최적화 **[20점]**

1) Task 1 모델에 적용된 최적화 기법의 선택 및 적용 과정의 적절성과 효과
2) 최적화 적용 후, 모델의 추론 시간 단축 정도 및 추론 정확도에 미친 영향

2. 코드 품질 [10점]

가. 일관적인 코드 스타일 및 구조 **[5점]**

나. 주석을 통한 코드 설명 및 일관적인 명명 규칙 **[5점]**

3. 보고서 완성도 [30점]

가. 프로젝트 목적, 개발 및 구현 내용에 대한 명확한 서술 **[10점]**

나. 분석 및 최적화 과정의 상세한 설명 **[10점]**

다. 정보의 구조화, 가독성 등 명확한 보고서 포맷 및 표현 **[10점]**