Distributed Deep Neural Networks Over the Cloud, the Edge and End Devices

Surat Teerapittayanon, Bradley McDanel and H. T. Kung

2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS), Atlanta, GA, 2017, pp. 328-339.

Introduction

- 컴퓨터 비전 분야를 위한 높은 성능의 CNN 모델이 꾸준히 개발됨.
  - LeNet(1998), AlexNet(2012), VGGNet, GoogLeNet(2014), ResNet(2015)

- 많은 수의 IoT 장치가 사용되고 있으며, 더욱 더 많아질 것으로 전망.
  - IoT 장치는 입력 데이터를 수집하는 센서에 직접 연결되기 때문에 기계학습 응용에 매력적인 대상.

- 현재 엔드 디바이스에서 데이터를 수집한 데이터를 처리하는 방식은 좋지 않음.
  - 입력 데이터를 클라우드로 전송해 대형 NN 모델로 처리.
    - 통신 비용, 레이턴시, 개인정보 문제 발생.
  - 간단한 기계학습 모델을 사용해 엔드 디바이스에서 직접 분류하면 정확도가 떨어짐.

Introduction

- 기존 방식의 문제를 해결하기 위해 분사 처리 방식이 등장.
  - 클라우드-엣지-엔드 디바이스로 구성된 계층적 분산처리.
    - 엔드 디바이스에서 작은 모델을 통해 신속한 추론을 제공하며,
      복잡한 샘플의 경우 클라우드의 대형 모델을 통해 추론을 제공.
    - 항상 데이터를 클라우드에 전송하는 것과 비교할 때, 통신 비용이 적게 들며
      엔드 디바이스에서만 추론하는 것보다 높은 정확도를 얻음.
    - 엔드 디바이스에서 처리한 데이터를 클라우드에 전송하기 때문에
      기존 방식과 달리 개인정보를 보호할 수 있다.

- 이러한 분산 처리 방식에도 문제가 있음.
  - 엔드 디바이스는 제한된 성능을 가지고 있기 때문에
    제한된 성능에도 불구하고 충분한 정확도를 구현할 수 있는 모델이 필요.
  - 계층간 신경망 모델을 파티셔닝 하기 때문에
    노드간 중간 결과를 전송하는데 높은 통신 오버헤드가 발생할 수 있음.

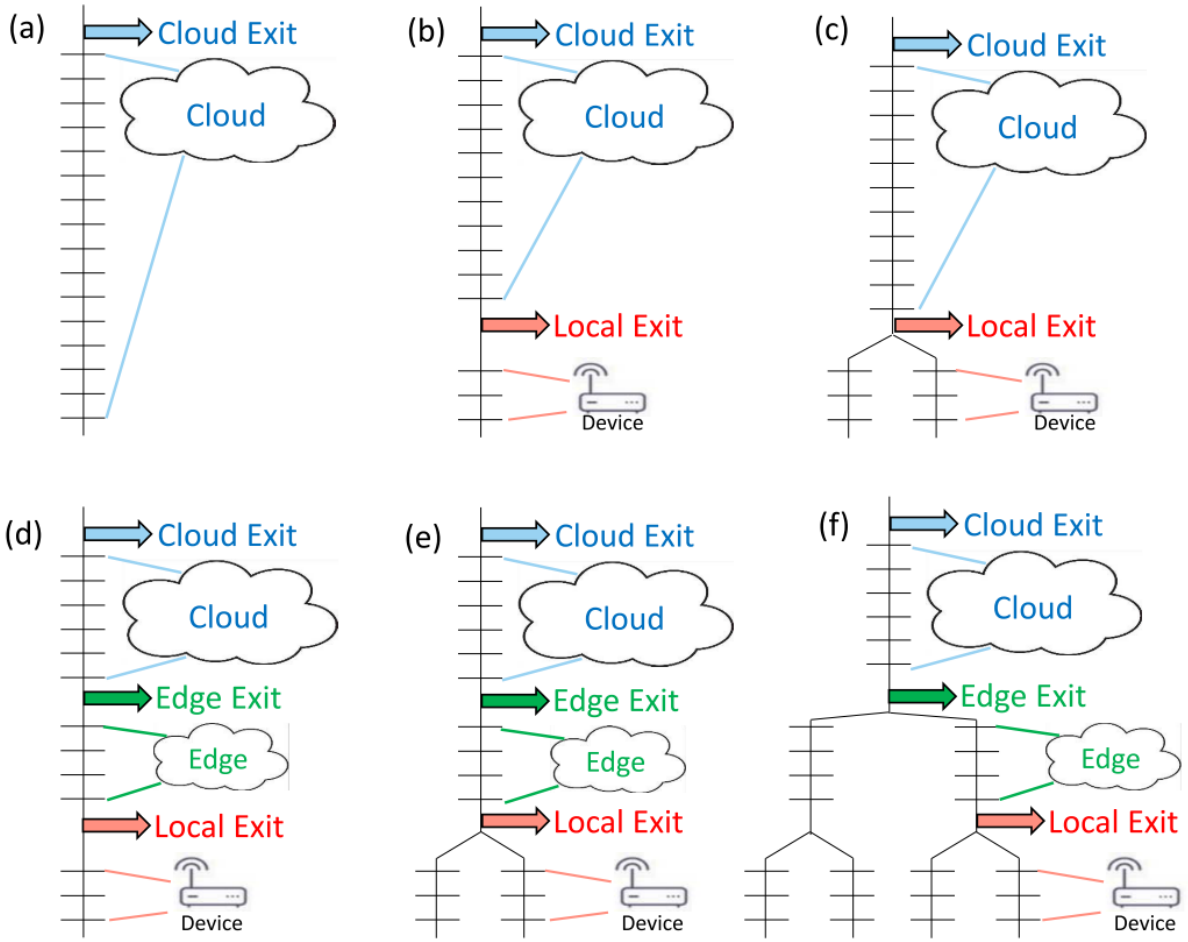
Introduction

- 클라우드-엣지-엔드 디바이스로 구성된 분산 컴퓨팅 구조(DDNN)를 제안.
  - 단일 DNN 모델을 분산 컴퓨팅 계층에 매핑.
  - 빠르고 지역화 된 추론과 클라우드의 최종 추론을 제공.
  - 확장 가능한 분산 컴퓨팅 계층 구조.

- BranchyNet을 기반으로 한 모델을 사용.
  - early exit 포인트를 DNN에 배치하고
    엔트로피에 기초한 신뢰도를 기준으로 출력 여부를 판단한다.
  - 로컬 추론에서 충분한 신뢰성을 갖는다면 로컬에서 출력.
  - 추가 연산이 필요하면 엣지, 클라우드로 전송해 연산.
    - S. Teerapittayanon, B. McDanel, and H. Kung,
      “Branchynet: Fast inference via early exiting from deep neural networks”

- 메모리가 부족한 엔드 디바이스에서 심층 신경망을 활용할 수 있도록
  BNN을 활용.

DDNN 계층 구조


(a)
Cloud Exit
Cloud
(b)
Cloud Exit
Cloud
Local Exit
Device
(c)
Cloud Exit
Cloud
Local Exit
Device
(d)
Cloud Exit
Cloud
Edge Exit
Edge
Local Exit
Device
(e)
Cloud Exit
Cloud
Edge Exit
Edge
Local Exit
Device
(f)
Cloud Exit
Cloud
Edge Exit
Edge
Local Exit
Device

DDNN 학습

- DDNN의 훈련은 단일 서버에서 진행할 수 있음.

- 여러 개의 exit point를 사용하며,
  각 exit point의 손실은 역전파 과정에서 결합되어
  전체 신경망을 공동으로 훈련할 수 있게 함.
  - 각 exit point가 깊이에 비해 좋은 정확도를 가질 수 있게 함.

- 소프트맥스 크로스 앤트로피 손실함수를 최적화 목적으로 사용.

DDNN 추론

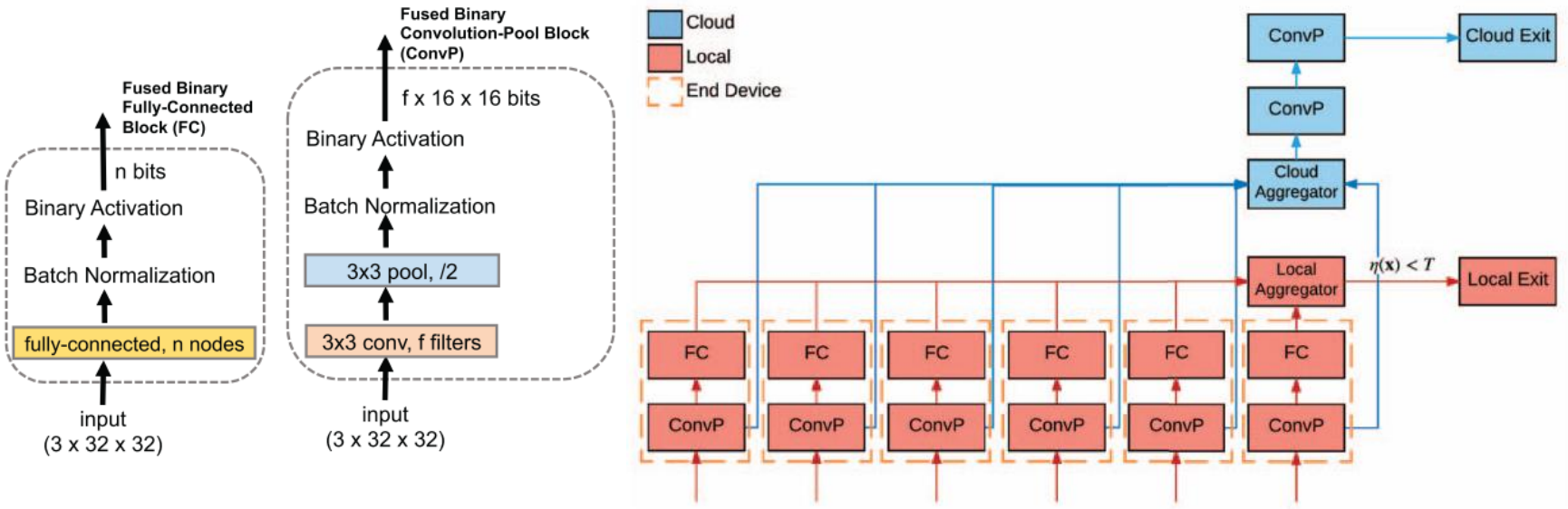
- 각 exit point에 미리 임계값(T)을 설정하고, 예측에 대한 신뢰도를 측정.

- 임계값과 비교할 값으로 정규화된 엔트로피(n)를 사용.
  - 정규화된 엔트로피는 0과 1 사이의 값을 가짐.
    - 대응하는 임계값을 더 쉽게 해석하고 검색할 수 있게 함.
    - 예측에 대한 신뢰도가 높을 수록 0에 가깝고 낮을 수록 1에 가까움.

- 각 exit point에서 정규화된 엔트로피와 임계값을 비교해
  예측 값에 대한 신뢰도가 떨어지는 경우(n > T),
  충분한 신뢰도가 확보될 때까지 상위 계층으로 보냄.

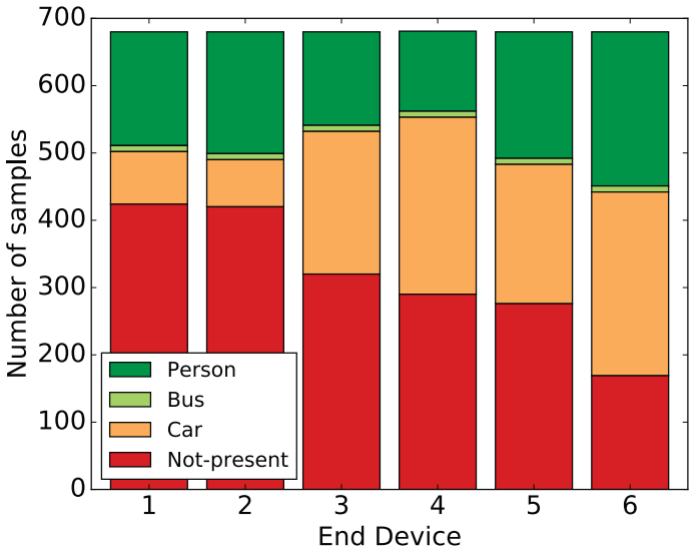
DDNN 평가

- 모델은 BNN을 기반으로 한 FC 블록과 ConvP 블록으로 구성.
- 6개의 엔드 디바이스와 클라우드를 사용.



DDNN 평가

- Multi-view multi-camera dataset
  - 서로 다른 위치에서 동시에 동일한 영역을 바라보는 6대의 카메라에서 획득한 이미지
  - 32 x 32 RGB 픽셀 이미지
  - 680개의 훈련 샘플과 171개의 테스트 샘플로 분할



DDNN 평가

- Aggregator의 종합 방식에 따른 결과
  - MP: max pooling
    - 각 구성요소의 최대값을 취해 입력 벡터를 집계
  - AP: Average pooling
    - 각 구성요소의 평균값을 취해 입력 벡터를 집계
  - CC: Concatenation
    - 입력 벡터를 연결해 모두 사용

DDNN 평가

- Aggregator의 종합 방식에 따른 결과

Schemes	Local Acc. (%)	Cloud Acc. (%)
MP-MP	95	91
MP-CC	**98**	**98**
AP-AP	86	98
AP-CC	75	96
CC-CC	85	94
AP-MP	88	93
MP-AP	89	97
CC-MP	77	87
CC-AP	80	94

MP-MP는 로컬에서는 높은 분류 정확도를 보이지만 클라우드에서는 떨어지는 정확도를 보여줌

엔드 디바이스에서 클라우드로 보낸 정보는 각 장치에서 활성화 함수를 거친 결과
이 피쳐를 다시 MP 하는 것은 좋은 성능을 보이지 못함.

DDNN 평가

- Aggregator의 종합 방식에 따른 결과

Schemes	Local Acc. (%)	Cloud Acc. (%)
MP-MP	95	91
MP-CC	**98**	**98**
AP-AP	86	98
AP-CC	75	96
CC-CC	85	94
AP-MP	88	93
MP-AP	89	97
CC-MP	77	87
CC-AP	80	94

MP-CC는 가장 높은 정확도를 보임.

모델을 훈련할 때, MP-MP는 역전파가 진행되는 동안 최대값을 제공한 장치에만 그래디언트를 전달.
하지만 MP-CC는 모든 장치에 그래디언트를 전달.

DDNN 평가

- Aggregator의 종합 방식에 따른 결과

Schemes	Local Acc. (%)	Cloud Acc. (%)
MP-MP	95	91
MP-CC	**98**	**98**
AP-AP	86	98
AP-CC	75	96
CC-CC	85	94
AP-MP	88	93
MP-AP	89	97
CC-MP	77	87
CC-AP	80	94

CC-CC는 상대적으로 로컬에서 떨어지는 성능을 보임.

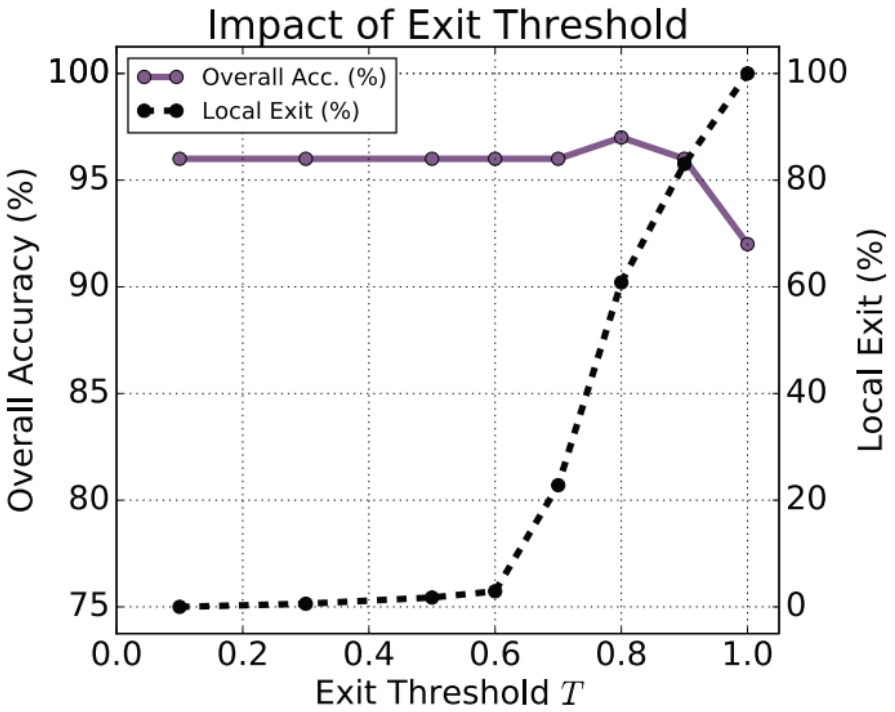
클라우드에서는 NN 레이어를 처리한 정보가 모두 유지되어 좋은 성능을 보여준다.
로컬에서는 여러 장치에서 처리된 동일한 클래스에 대한
출력 사이의 관계가 만들어지지 않아 성능이 저하됨.

DDNN 평가

- 엔트로피 임계값에 따른 결과
  - 임계값 T가 0이면 어떤 샘플도 출력되지 못하고
    1이면 모든 샘플이 출력된다.
    - 예측에 대한 신뢰도가 높을 수록 0에 가깝고 낮을 수록 1에 가까움.
  - Local Exit의 비율이 증가하면 Overall Acc. 가 떨어짐.
    - 일반적으로 로컬이 클라우드보다 정확도가 낮기 때문.

DDNN 평가

- 엔트로피 임계값에 따른 결과

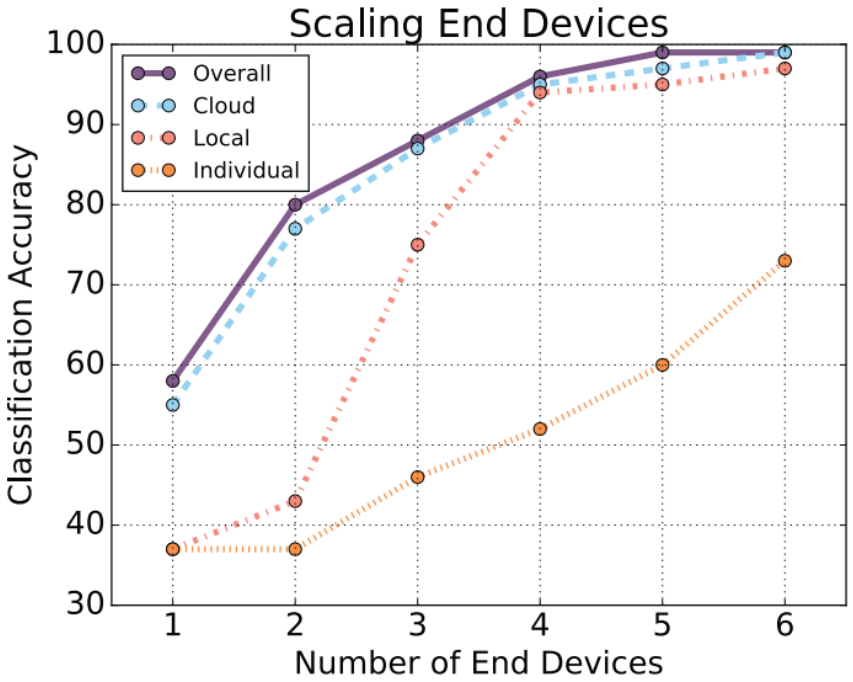


T	Local Exit (%)	Overall Acc. (%)	Comm. (B)
0.1	0.00	96	140
0.3	0.58	96	139
0.5	1.75	96	138
0.6	2.92	96	136
0.7	22.81	96	111
0.8	**60.82**	**97**	**62**
0.9	83.04	96	34
1.0	100.00	92	12

T가 0.8인 지점은 저레벨의 피처가 고레벨의 피처보다 정확한 분류 결과를 출력할 수 있는 지점.
이 임계값은 로컬과 클라우드의 분류기가 모두 가장 잘 작동하는 최적의 지점.
통신 비용과 정확도 간의 균형을 고려해 임계값 설정.

DDNN 평가

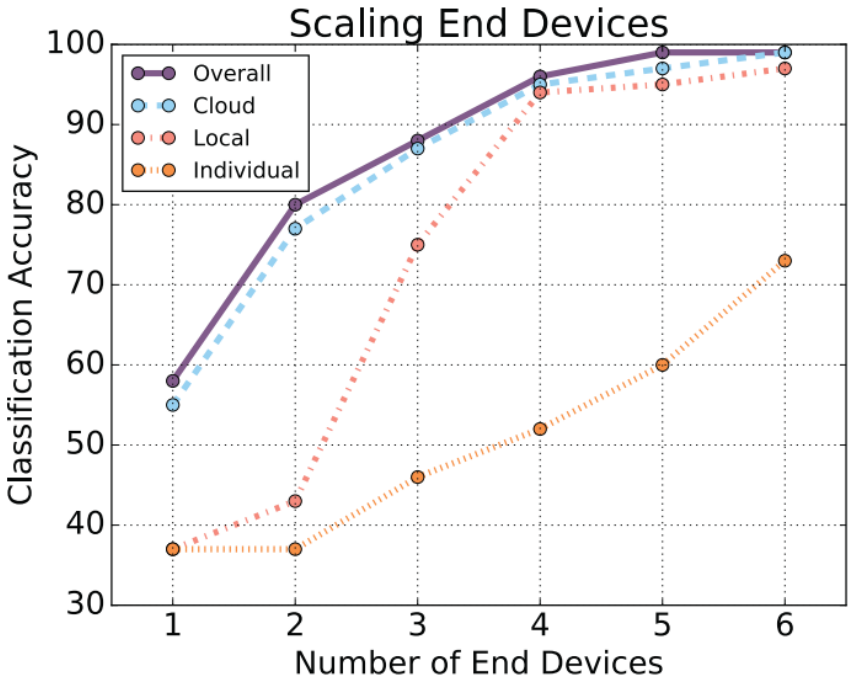
- 엔드 디바이스 수에 따른 결과



데이터셋의 배치가 불균형하기 때문에 개별 장치의 정확도가 크게 다름.

DDNN 평가

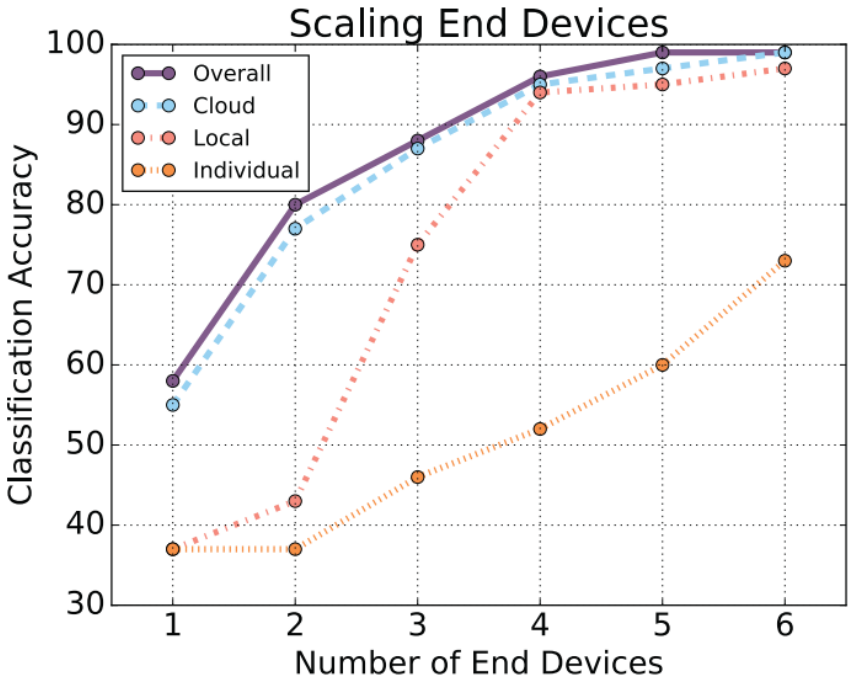
- 엔드 디바이스 수에 따른 결과



클라우드는 항상 로컬보다 높은 정확도를 보임.
하지만 엔드 디바이스가 4개 이상일 때부터 로컬과 클라우드 모두 높은 정확도를 출력.

DDNN 평가

- 엔드 디바이스 수에 따른 결과



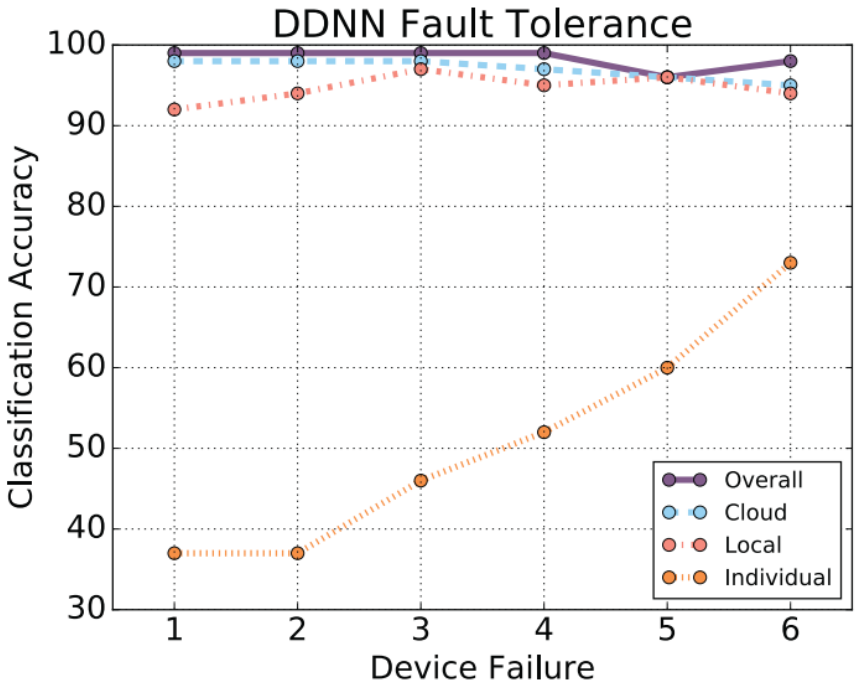
Overall accuracy는 클라우드보다 조금 높은 정확도를 보임.

- 저레벨의 피처가 고레벨의 피처보다 정확한 분류 결과를 출력할 수 있는 지점에 임계값을 설정했기 때문.

샘플의 60%가 로컬에서 출력되기 때문에 통신 비용이 크게 감소함.

DDNN 평가

- DDNN의 내결함성



엔드 디바이스가 손실되어도 연산의 정확도는 크게 떨어지지 않음.

DDNN 평가

- 통신 비용 절감
  - 32x32 RGB 픽셀 이미지를 전송하려면 3072b를 전송.
  - DDNN 모델은 비용이 가장 큰 모델을 사용해도 평균 140B만 전송
  - 통신 비용을 20배 이상 절약.

T	Local Exit (%)	Overall Acc. (%)	Comm. (B)
0.1	0.00	96	140
0.3	0.58	96	139
0.5	1.75	96	138
0.6	2.92	96	136
0.7	22.81	96	111
0.8	**60.82**	**97**	**62**
0.9	83.04	96	34
1.0	100.00	92	12

결론

- 클라우드-엣지-엔드 디바이스로 구성된
  수직적, 수평적으로 확장 가능한 분산형 DNN 구조를 제안.

- 대상 응용의 정확성, 통신 및 대기 시간에 대한 요구 사항을 충족하면서
  내결함성과 개인정보 보호와 같은 이점을 가짐.

- 로컬에서 많은 샘플을 추론하고, 추가 연산이 필요할 때만
  작은 바이너리 피처 데이터를 클라우드에 전송하기 때문에
  기존 방식보다 통신 비용을 크게 줄임.

결론

- 성능이 제한된 엔드 디바이스에서는 BNN이 유용하지만,
  클라우드에서는 그렇지 않음.

- 엔드 디바이스에서는 BNN을, 클라우드에서는 혼합 정밀도,
  또는 부동 소수점 신경망을 사용하는 방법을 추후 연구