School of Computing and Information Systems

DiTMoS: Delving into Diverse Tiny-Model Selection on Microcontrollers

Xiao MA, Shengfeng HE, Hezhe QIAO, Dong MA

Singapore Management University

PerCom'24

Enabling DNN on Microcontrollers is Attractive

- IoT devices → Microcontrollers(MCUs).
- Deep learning has become the **state-of-art solution** for most mobile applications.
- Offload computations to cloud server → not always realistic(latency, privacy).

Microcontroller

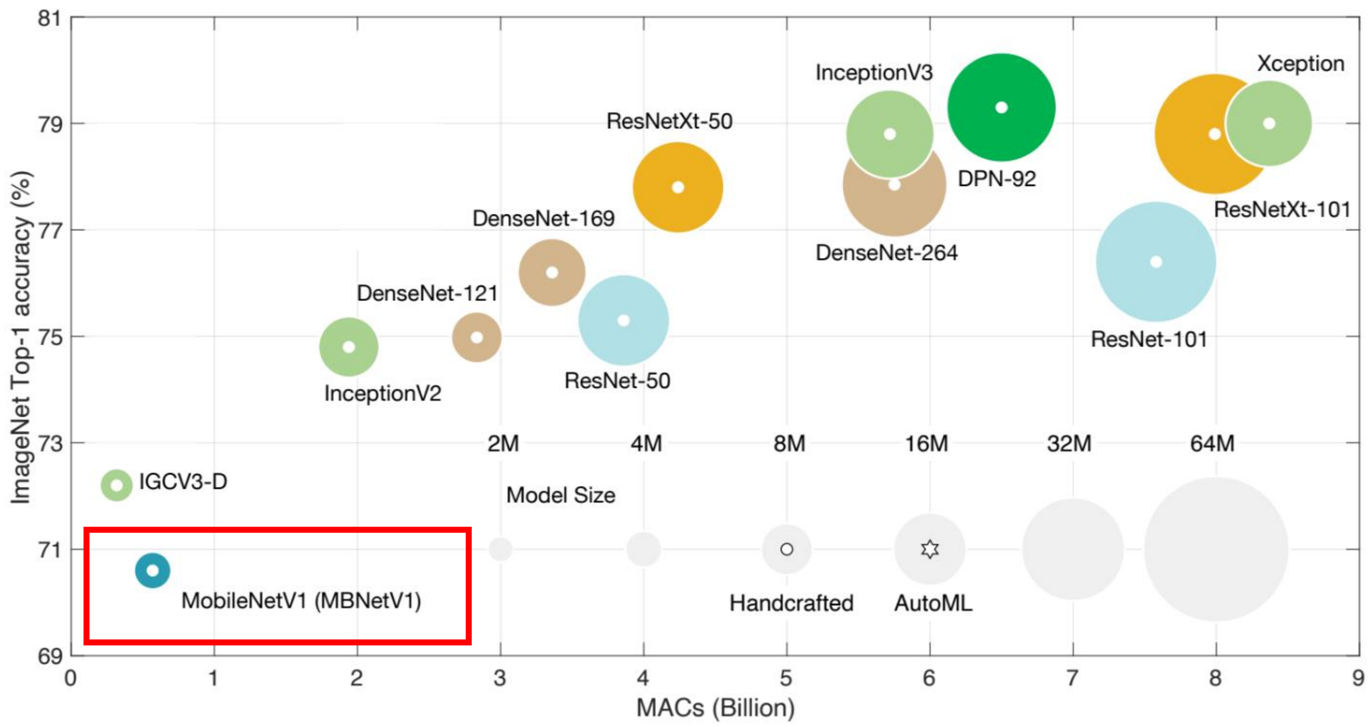
Smart City

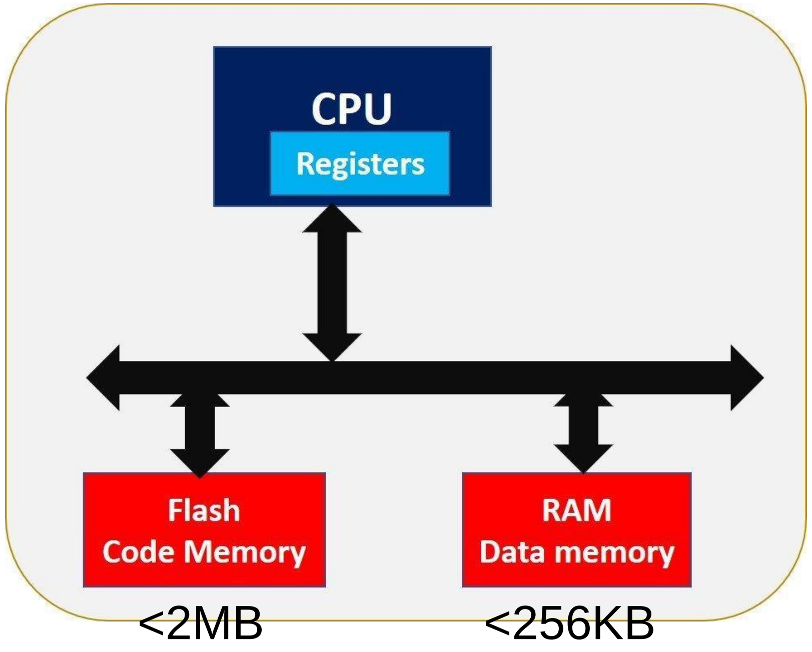
Smart Healthcare

Smart Retail

Smart Home

Challenges

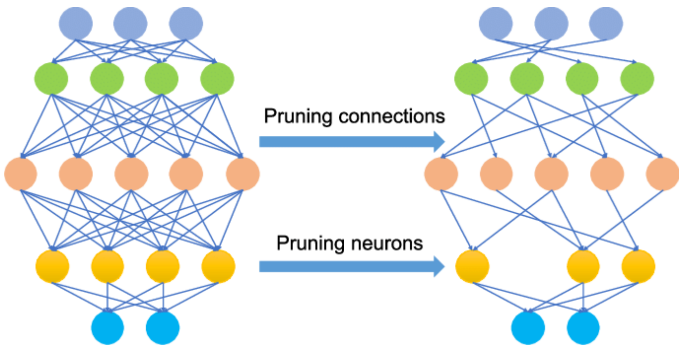




1. Deep learning models are too **large**.
2. MCU is usually resource-constraint. (Flash, Memory)

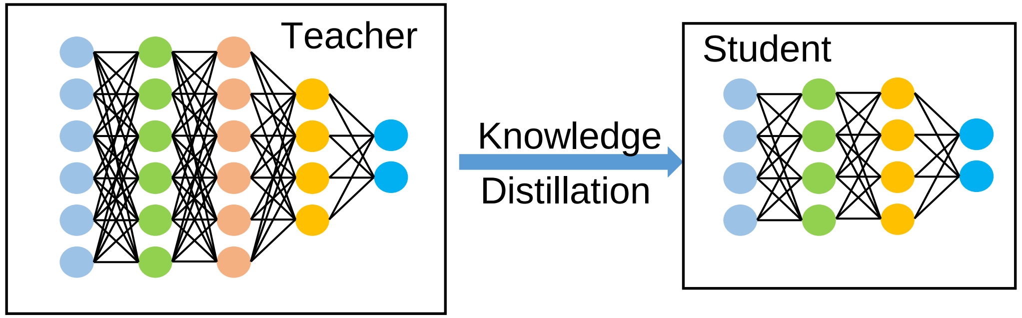
Existing Optimization Strategy

- Model compression: convert a large model to a tiny version.

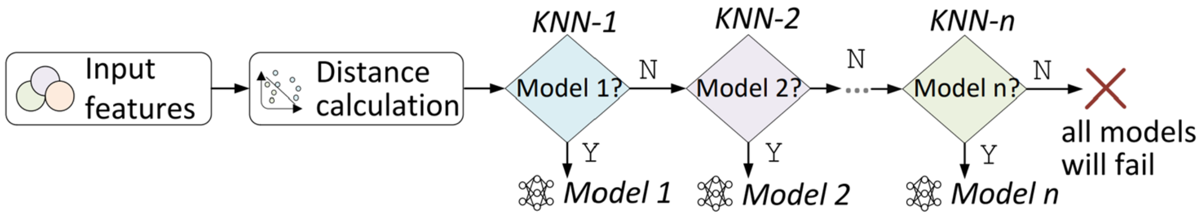


Pruning

Quantization



Knowledge Distillation



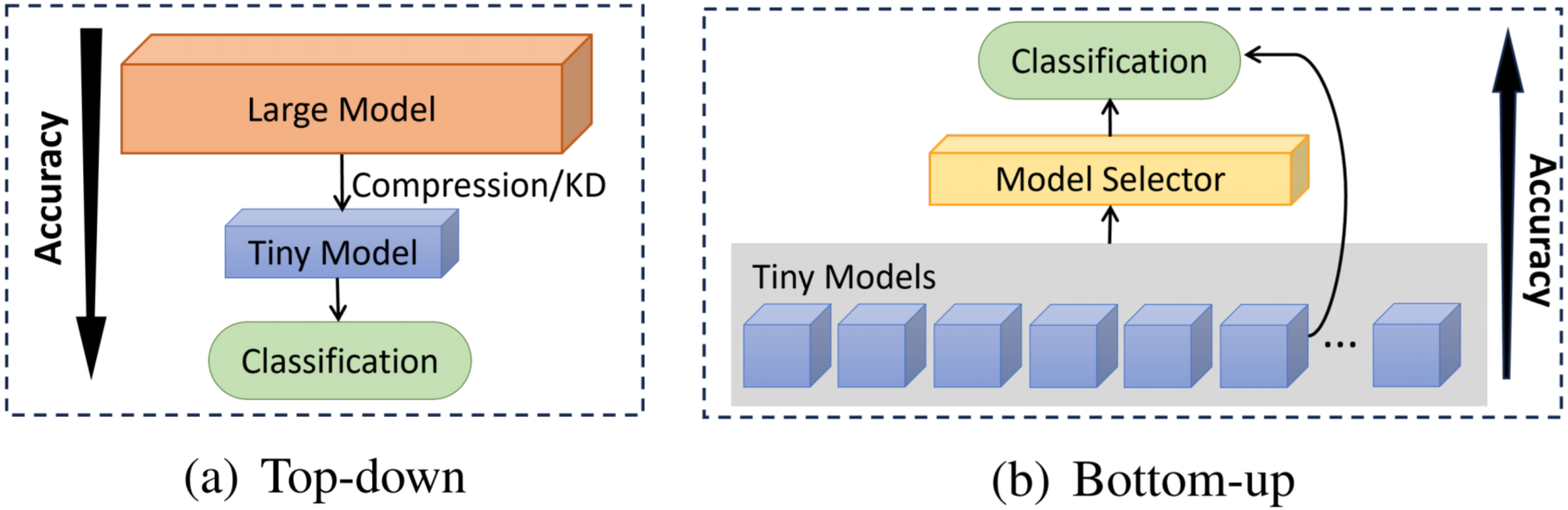
Model Selection [1]

Model compression compromises the accuracy.(High compression ratio)

[1] Taylor B, Marco V S, Wolff W, et al. Adaptive deep learning model selection on embedded systems[J]. ACM SIGPLAN Notices, 2018

Rethinking the Methodology from a Different Perspective

- Top-down vs. Bottom-up Methodology



Memory: Expensive(128KB)

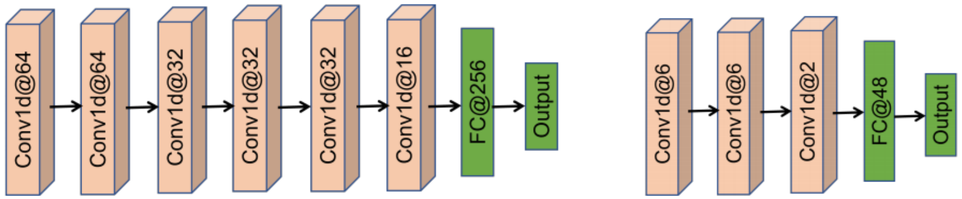
Flash: Cheap(1MB)

- Limited model **capacity** ⟹ Limited learned **knowledge** ⟹ Higher **diversity**
- The effectiveness of the bottom-up method relies on two **insights**:

1. Tiny models can perform **higher diversity than larger models**.
2. Aggregating multiple weak models promises a **higher upper bound** on classification accuracy.

Model Diversity

- Tiny(weak) Models vs Large(strong) models



(a) Strong model (b) Weak model

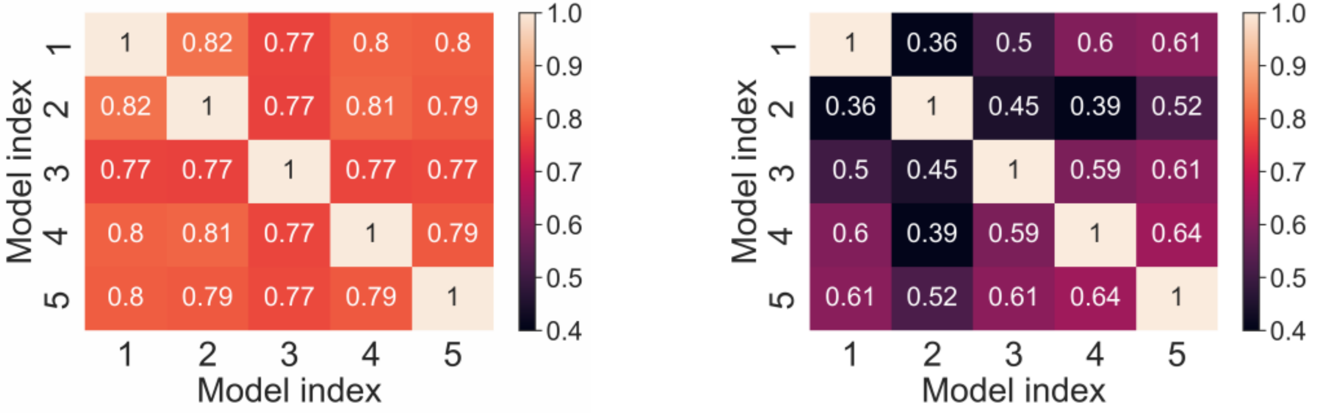
UniMiB-SHAR HAR dataset

Model Index	1	2	3	4	5
Strong Model (484KB)	95.3%	95.9%	95.2%	96.1%	95.9%
Weak Model (28KB)	64.7%	57.8%	58.2%	61.4%	60.9%

Note: Each model has different initialization.

- Similarity of the model representations(CKA similarity).

$$CKA(X,Y) = \frac{\|Y^T X\|_F^2}{\|X^T X\|_F \|Y^T Y\|_F}$$

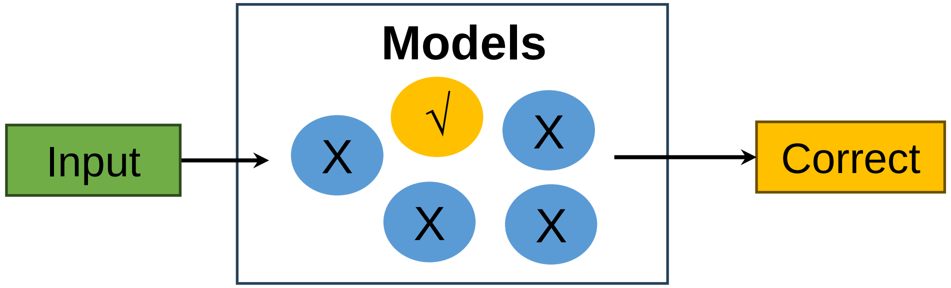


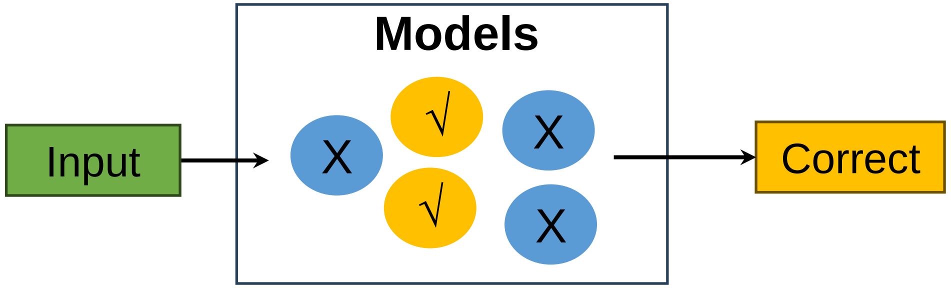
(a) Strong models (b) Weak models

[1] Kornblith S, Norouzi M, Lee H, et al. Similarity of neural network representations revisited[C]//International conference on machine learning. PMLR, 2019.

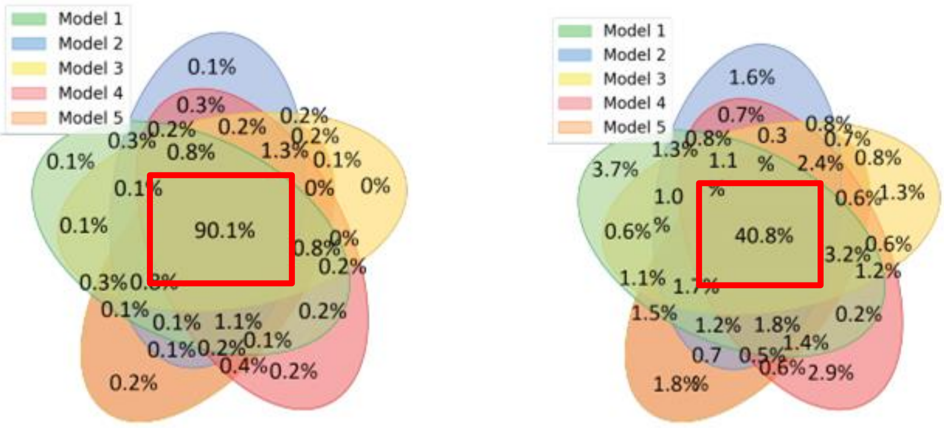
Key insight: union accuracy

Union accuracy: the percentage of samples that can be correctly classified by **at least** one model.





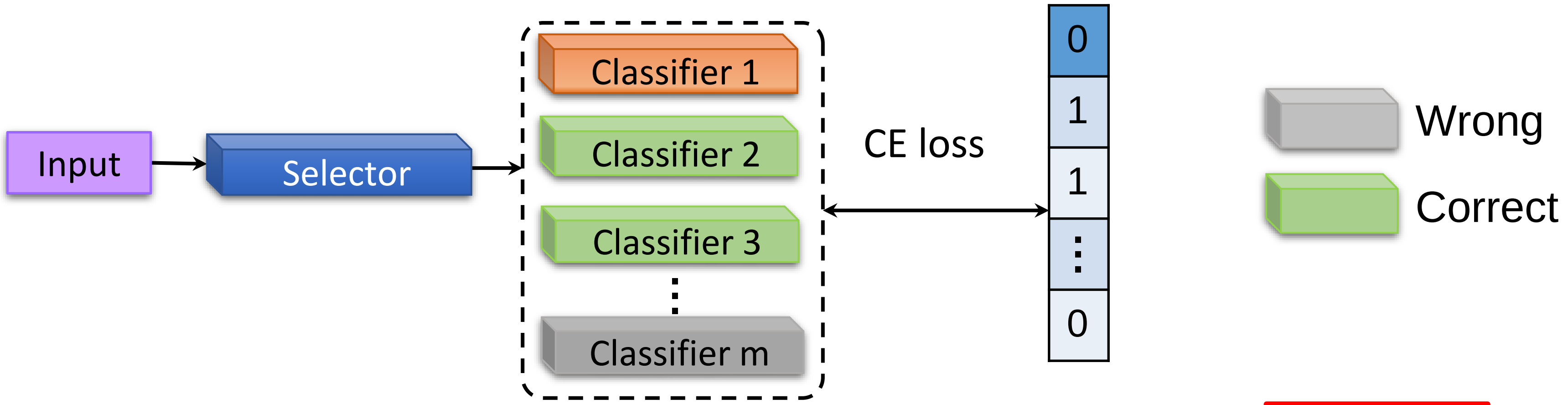
Model Index	1	2	3	4	5	Union Accuracy
Strong Model (484KB)	95.3%	95.9%	95.2%	96.1%	95.9%	**98.9%** (↑ 2.8)
Weak Model (28KB)	64.7%	57.8%	58.2%	61.4%	60.9%	**81.5%** (↑ 16.8)



(a) Strong models (b) Weak models

We can benefit from the **union accuracy** if we can select the correct model.

The Failure of Model Selection

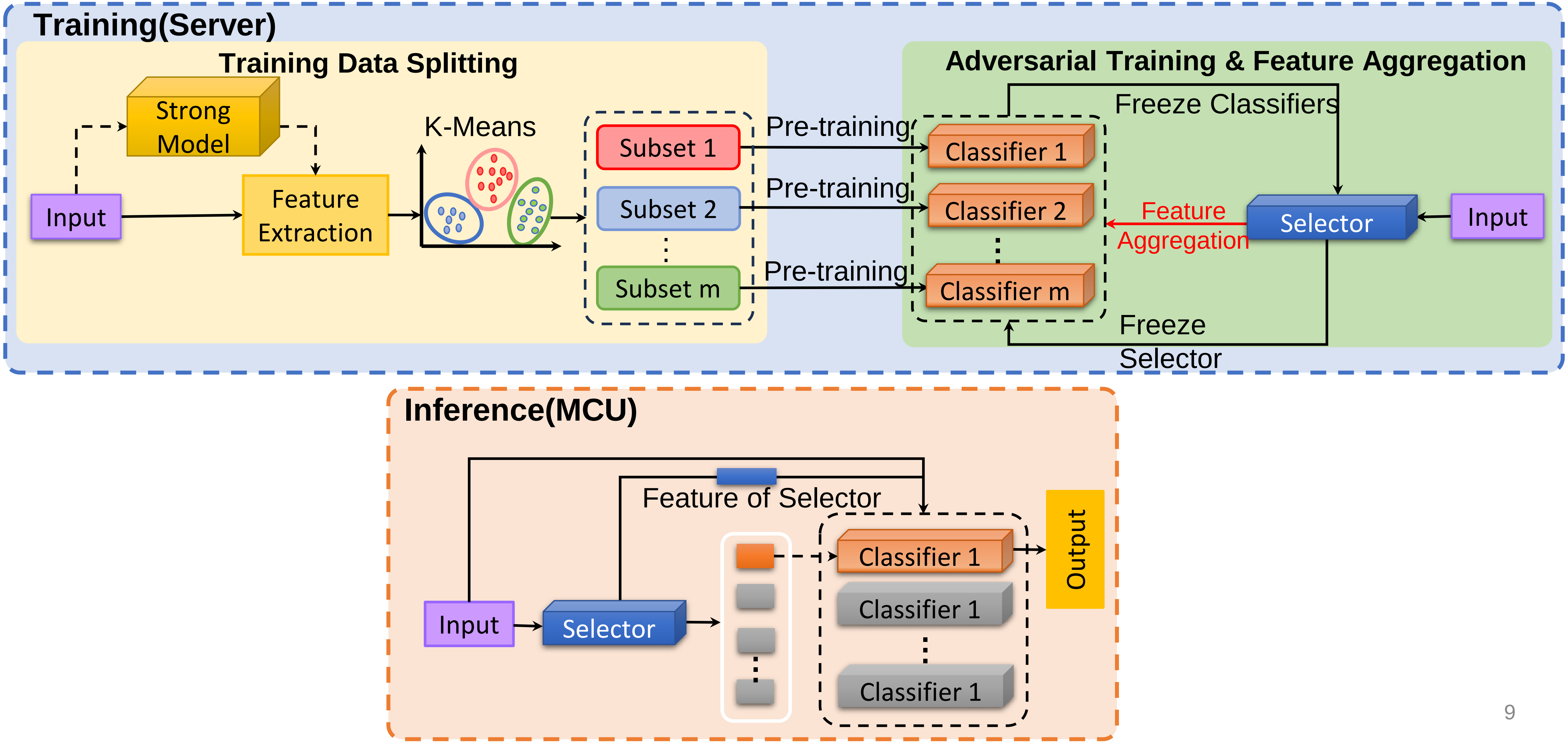


Model Index	1	2	3	4	5	Union Accuracy	Selection Accuracy
Strong Model (484KB)	95.3%	95.9%	95.2%	96.1%	95.9%	**98.9%**	
Weak Model (28KB)	64.7%	57.8%	58.2%	61.4%	60.9%	**81.5%**	**66.6%**

- Naïve model selection failed because of **two reasons**.

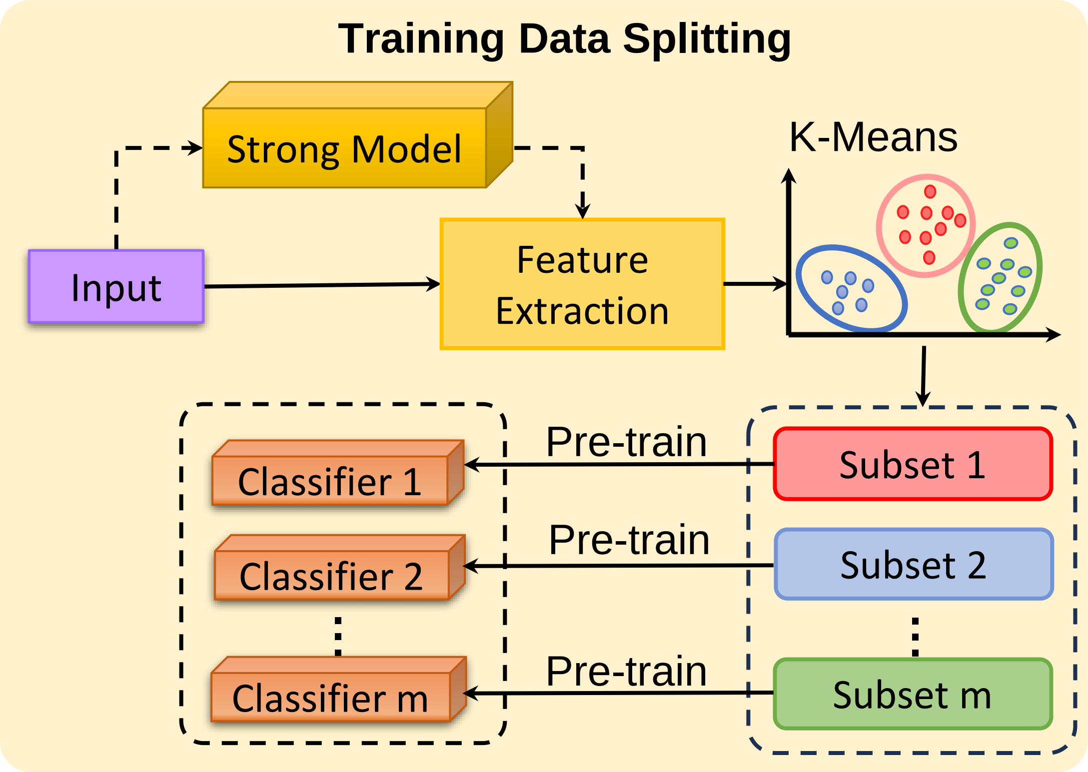
1. Independent training classifiers cannot provide enough **diversity(multi-label)**.
2. The selector and the classifiers are **mutually related**, but naïve model selection fails to capture the relationship.

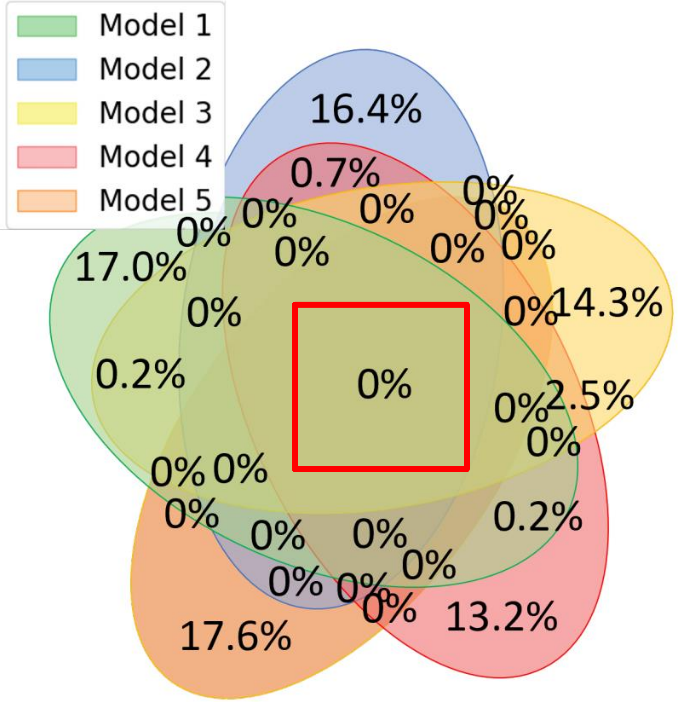
DiTMoS Framework Overview



Training Stage 1: Training Data Splitting

- Splitting the dataset to **several subsets** to encourage the **model diversity**.
- (multi-label problem)

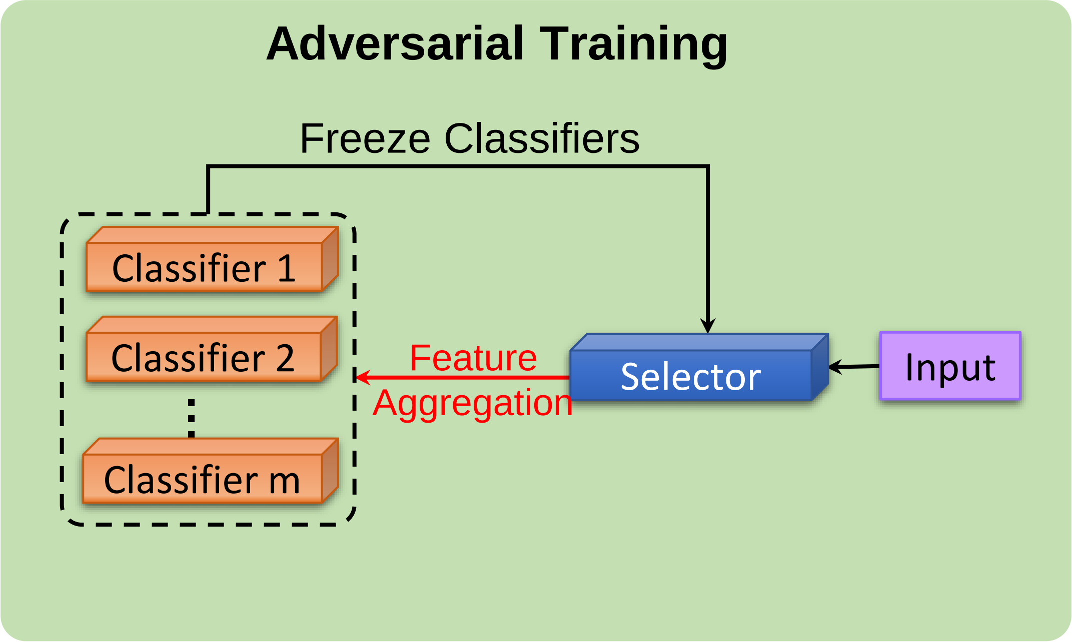




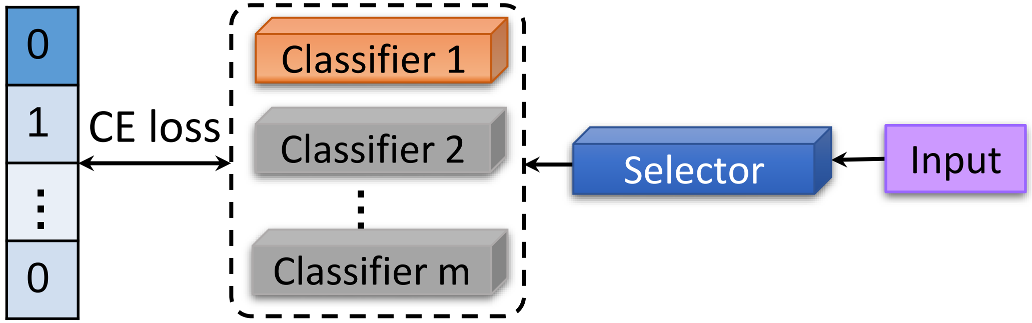
- Circle: different K-means clusters.
- Color: different classes.

Training Stage 2: Adversarial Training

- Adversarial training capture the relationship between selector and classifiers.
- Similar to train generator and discriminator in GAN **iteratively**.

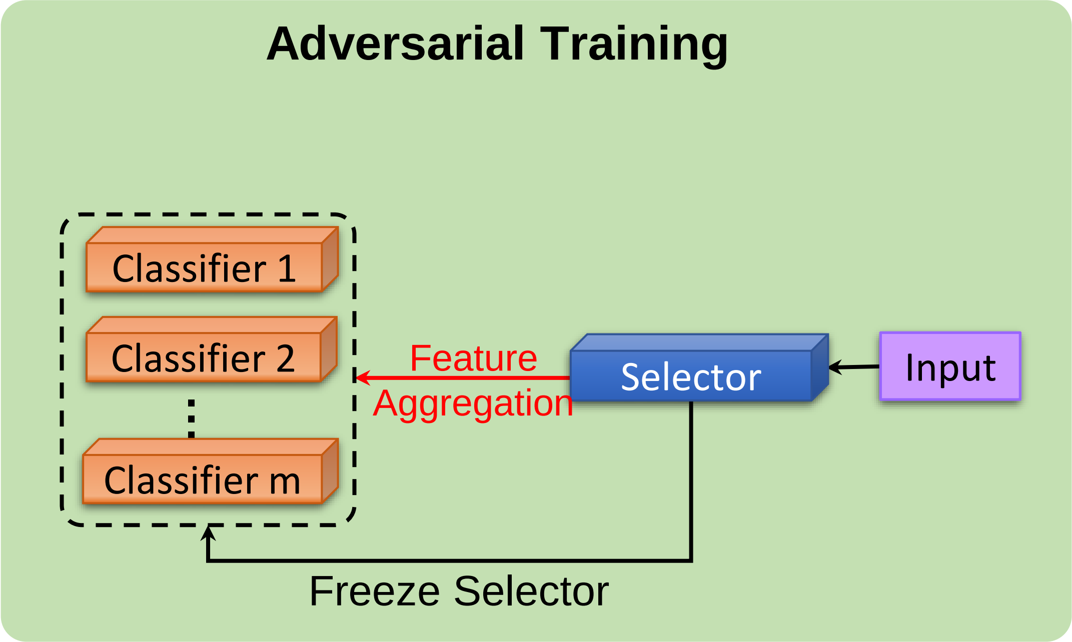


- Step 1. Freeze the classifiers, **train the selector.**

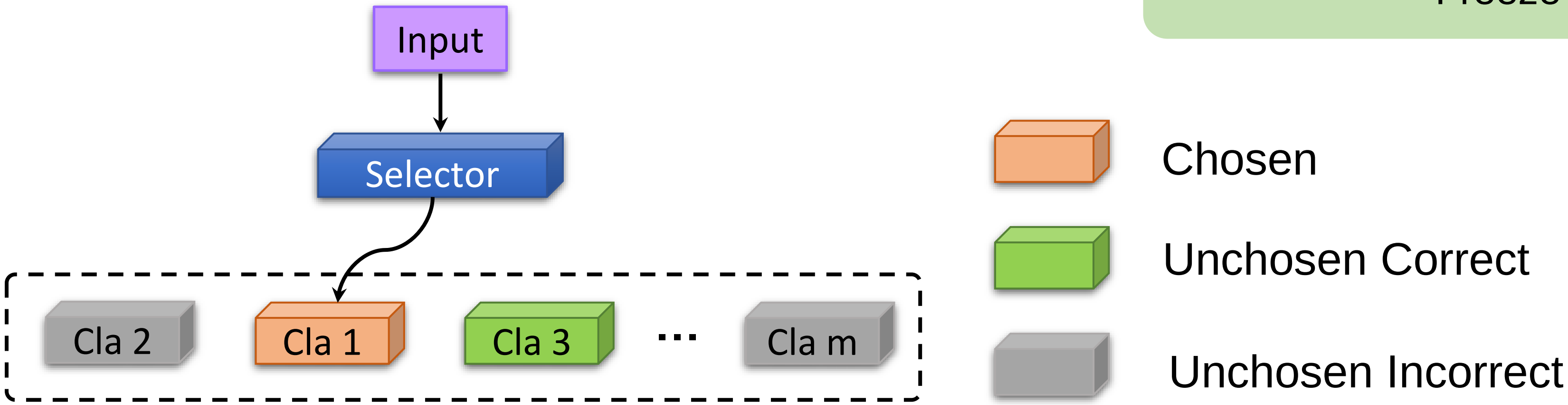


Training Stage 2: Adversarial Training

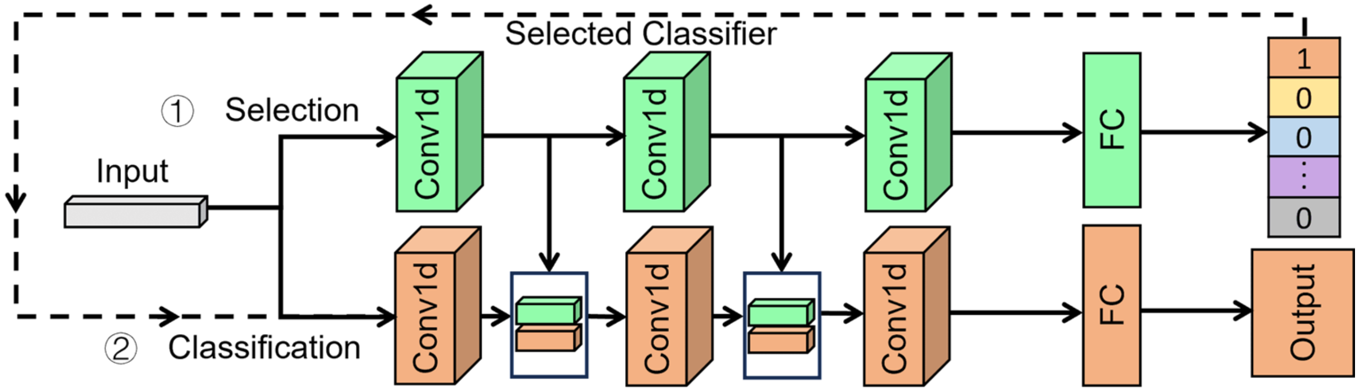
- Step 2. Freeze the selector, **re-train the classifiers**.
  - Train the classifier selected by the selector.
  - Reduce the overlap of classifiers.(diversity)
  - Improve the union accuracy.



$$Loss = \boxed{CE_{sel}} + \boxed{\alpha \cdot CE_{single} + \beta \cdot CE_{union} + \gamma \cdot CE_{overlap}}$$

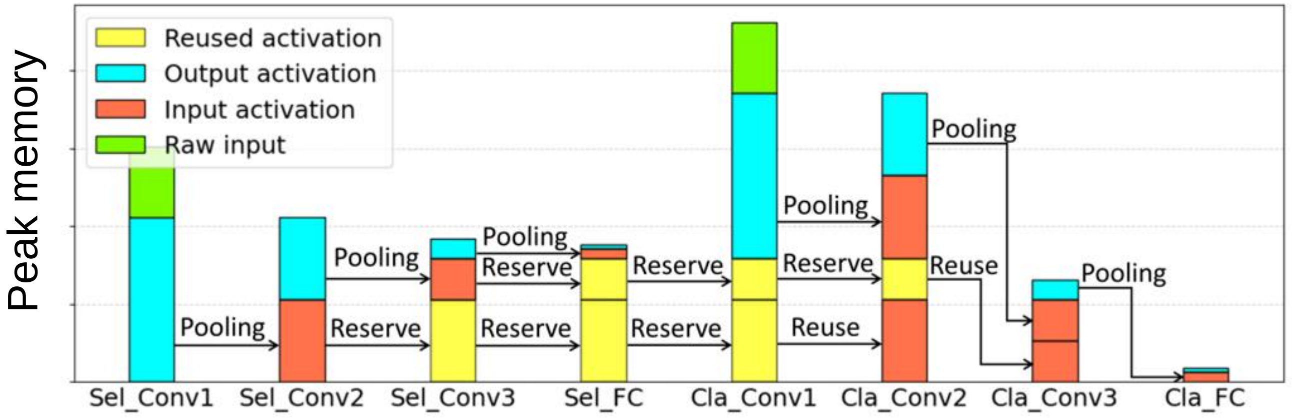


Training Stage 3: Feature Aggregation



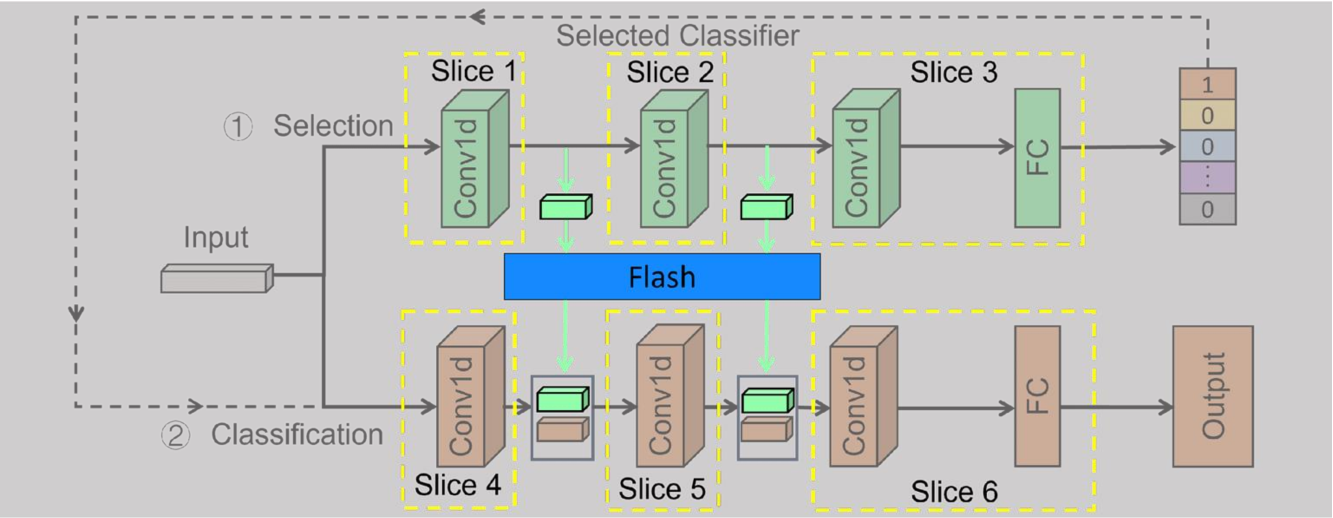
- Increase the representation capability.
- Add global information to the classifiers.

Problem: feature aggregation increase **memory consumption**. (DNN inference on MCU is layer-by-layer).

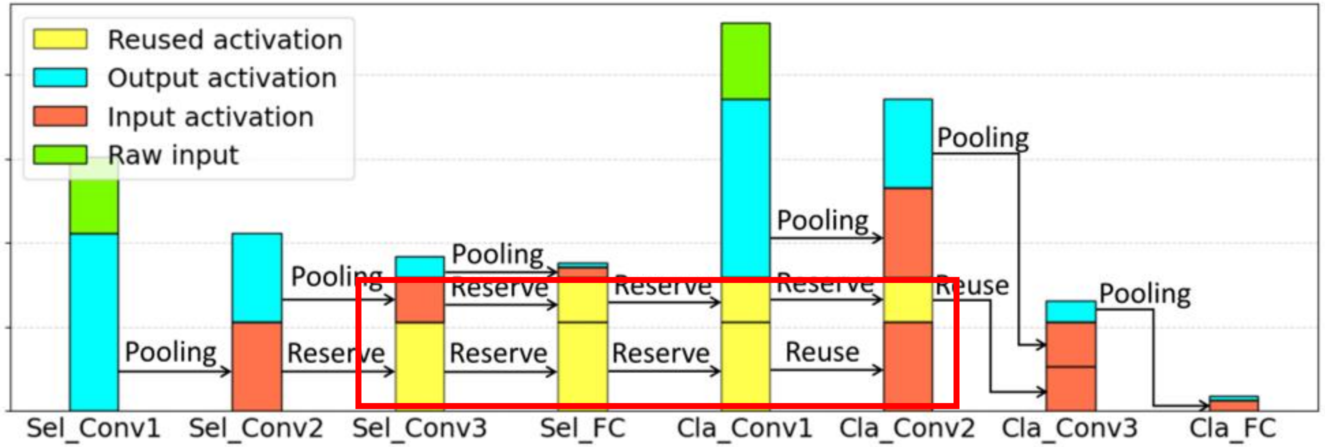


Implementation: Network Slicing

- Network slicing: Store the reused feature in the Flash.

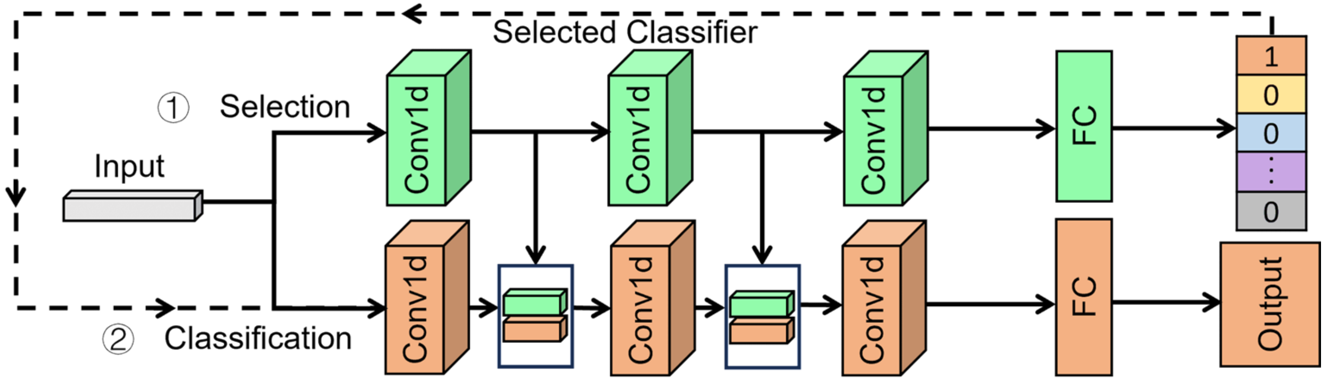


Flash is much cheaper than memory.
Flash is slower than memory.



Evaluation

- Datasets:
  - UniMiB-SHAR (human activity recognition, accelerometers)
  - Speech Commands(keyword spotting, microphone)
  - DEAP(Emotion recognition, EEG)
- Device
  - Hardware: STM32F767ZI (RAM: 512KB, FLASH: 2MB)
  - AI toolchain: STM32Cube.AI
- Model



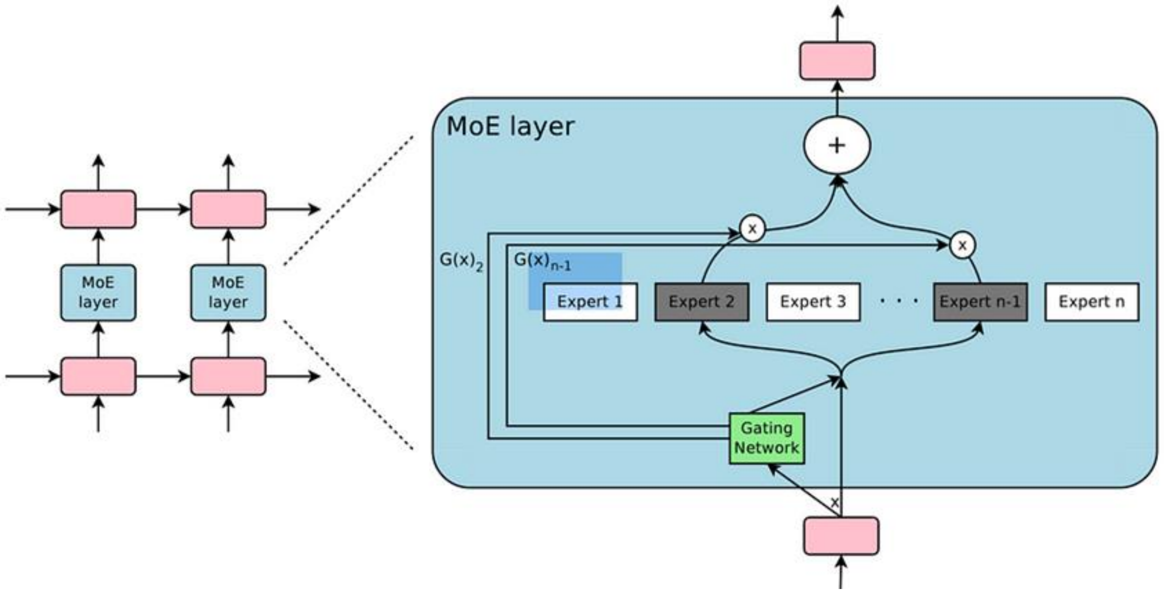
6 classifiers

Evaluation

- Baselines

Baseline	Description
SigCla	A 6-layer single CNN
SigCla-KD	A 6-layer single CNN with SOTA Knowledge Distillation[1]
Ada-MoE	A Mixture of Expert architecture using the same model as DiTMoS
Ensemble	Two 3-layer CNNs using an averaging ensemble

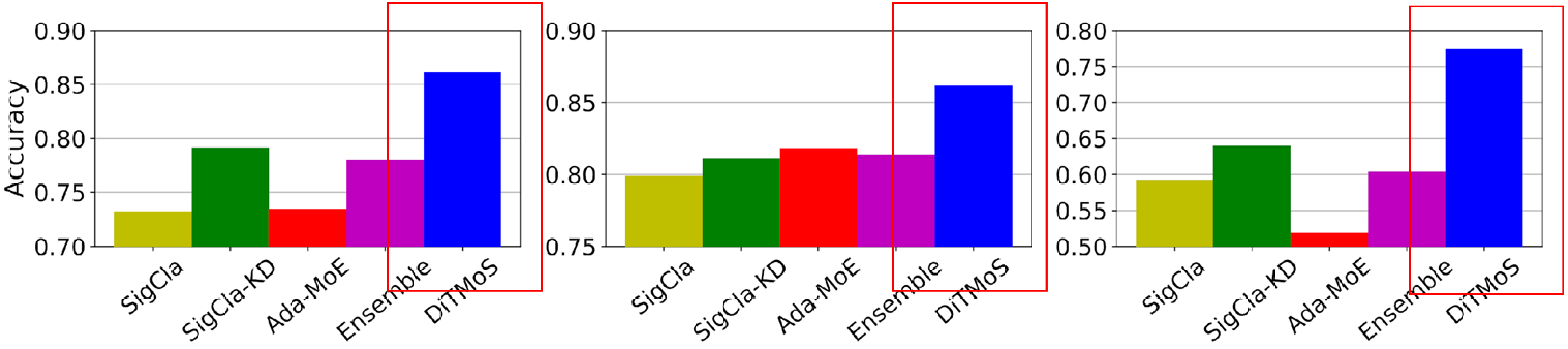
Mixture of Experts(MoE)



[1] Huang T, You S, Wang F, et al. Knowledge distillation from a stronger teacher[J]. Advances in Neural Information Processing Systems, 2022, 35: 33716-33727.
[2] Shazeer N, Mirhoseini A, Maziarz K, et al. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer[J]. arXiv preprint arXiv:1701.06538, 2017.

Experimental Results

Overall performance

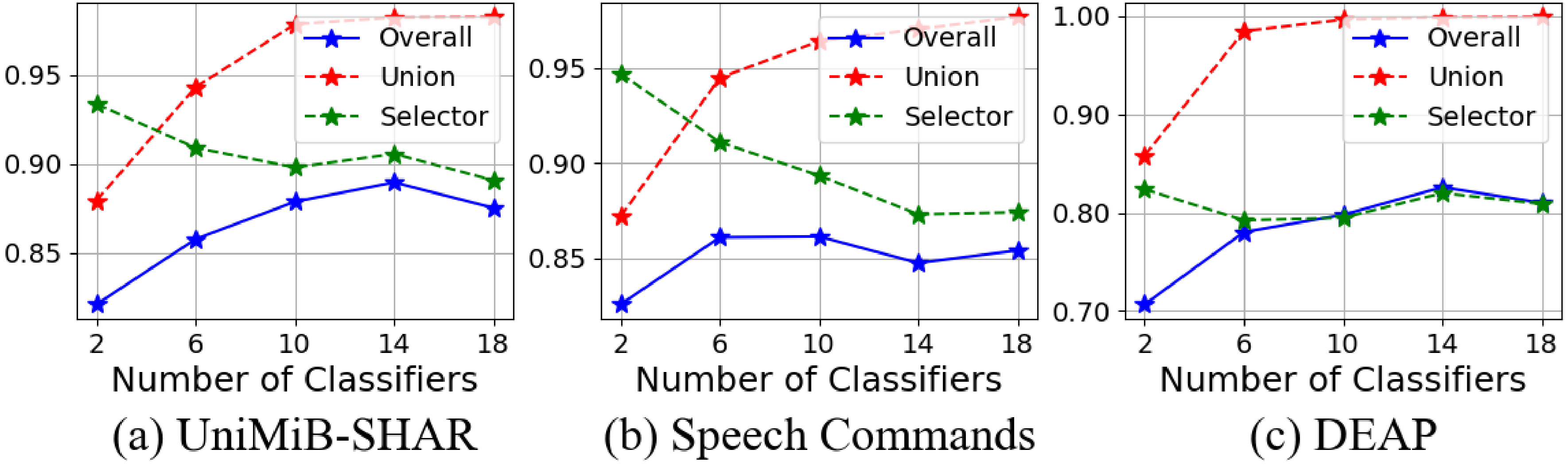


(a) UniMiB-SHAR (b) Speech Commands (c) DEAP

- DiTMoS achieves up to **13.4%** accuracy improvement compared to the best baseline.

Experimental Results

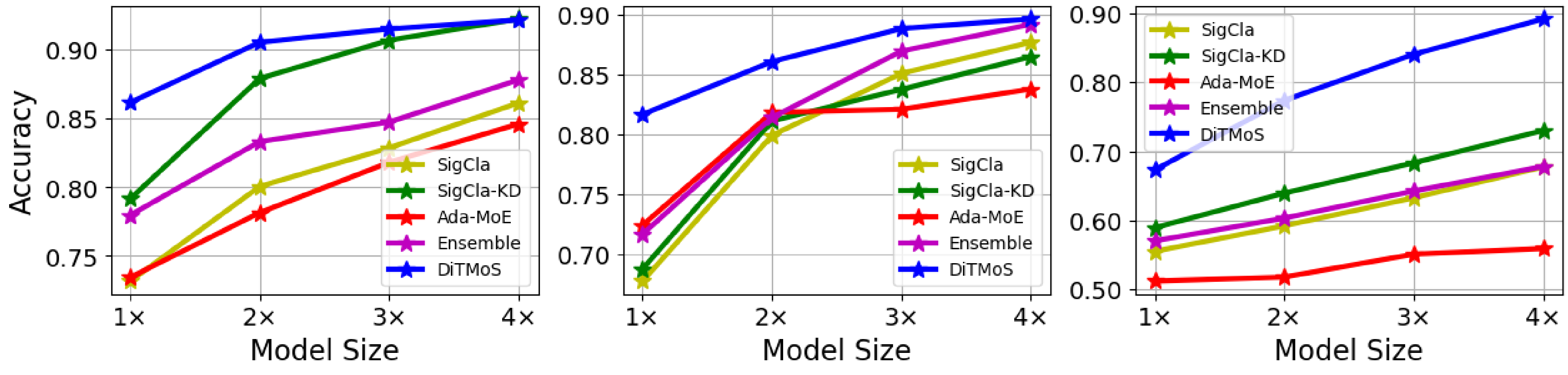
Impact of number of classifiers



- The **optimal** number of classifiers depends on the **datasets**.
- There will be a **tradeoff** between **selector performance** and **union accuracy**.

Experimental Results

Impact of model size



(a) UniMiB-SHAR (b) Speech Commands (c) DEAP

- DiTMoS consistently **outperforms** baselines under different model sizes.
- For UniMiB-SHAR and Speech Commands, DiTMoS shows higher improvement for smaller models.

Experimental Results

System Performance on UniMiB-SHAR

Approach	Memory Usage(KB)	Flash Usage(KB)	Latency (ms)	Energy (mJ)
SigCla	6.1	63.6	11.9	3.9
SigCla-KD	6.1	63.6	11.9	3.9
Ada-MoE	6.1	168.2	10.4	3.4
Ensemble	6.1	51.4	10.4	3.4
DiTMoS w/o Slicing	**8.5**	**166.9**	**10.9**	**3.6**
DiTMoS	**6.2**	**166.9**	**12.5**	**4.1**

- Without network slicing, the memory usage will be **higher** than other baselines.
- Network slicing will **reduce memory** usage but slightly **increase the latency**.

Experimental Results

Ablation Study

Ablation Study	UniMiB-SHAR	Speech Commands	DEAP
Random Splitting	84.9%	84.3%	75.8%
w/o Adversarial Training	72.6%	81.8%	56.3%
w/o Feature Aggregation	83.5%	86.0%	76.3%
DiTMoS	**86.2%**	**86.2%**	**77.4%**

- **Removing the feature aggregation** module can still achieve **higher performance** while maintain comparable **latency and memory usage**.

Takeaways

- We introduce the fresh concept of **Union Accuracy**, which is defined as the accuracy where a sample can be correctly classified by **at least** one weak model.
- Union accuracy provide another perspective to leverage the **model diversity** to reduce **computation overhead** of conventional ensemble and MoE approaches.

- DiTMoS consists of 3 major components: **training data splitting**, **adversarial training**, and **heterogeneous feature aggregation**.
- DiTMoS achieves up to 13.4% accuracy improvement compared to the best baseline.

- Future Works
  - Generalize DiTMoS to vision tasks.
  - Combine with neural architecture search(NAS) and model compression.

Thanks!

Presenter: Xiao MA

Email: xiaoma.2022@phdcs.smu.edu.sg

Code: https://github.com/TheMaXiao/DiTMoS