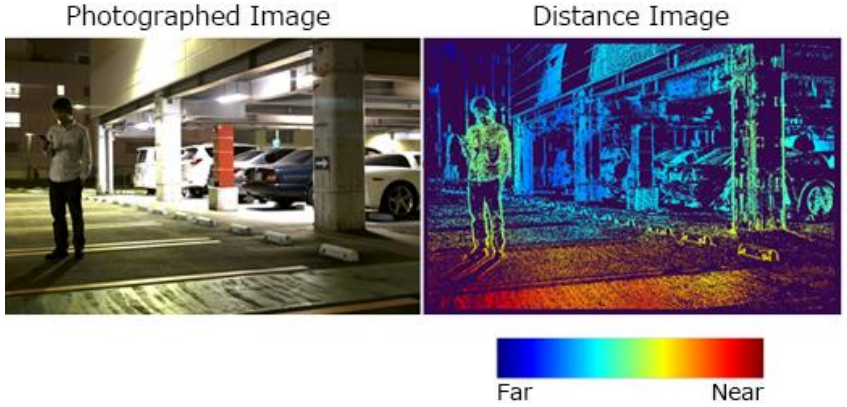
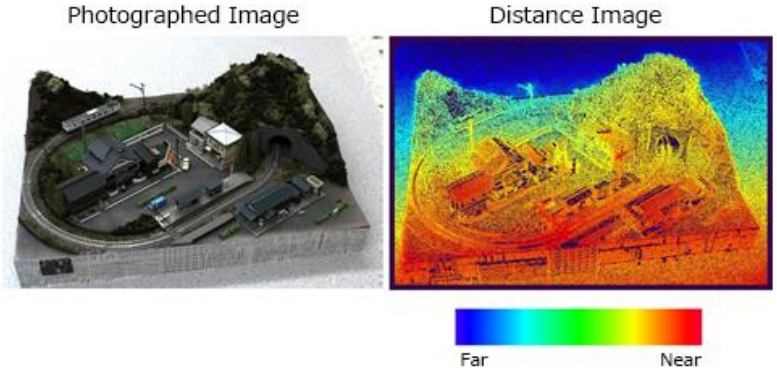
Cosmo: Contrastive Fusion Learning with Small Data for Multimodal Human Activity Recognition

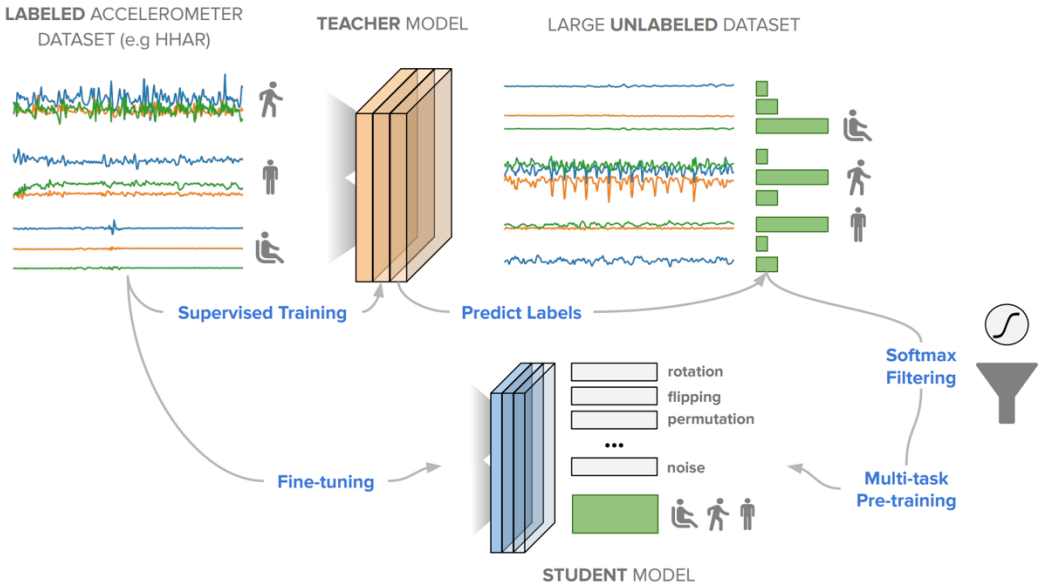
Presenter: Yunxiang Chi
09/23/2025

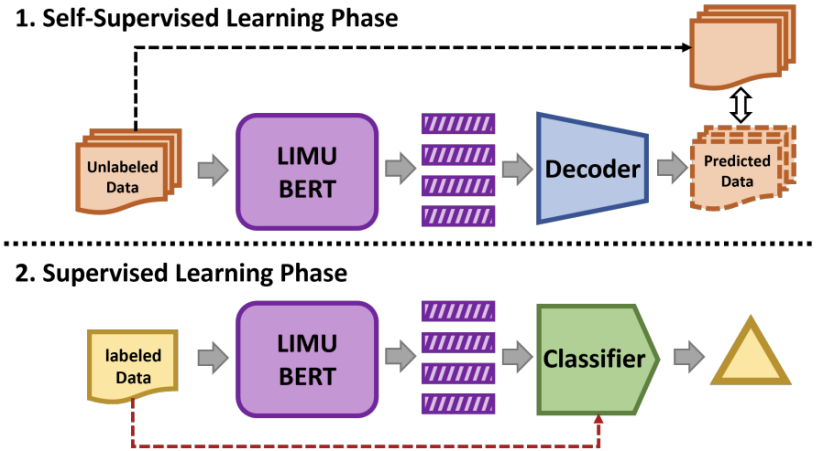
Human Activity Recognition(HAR)





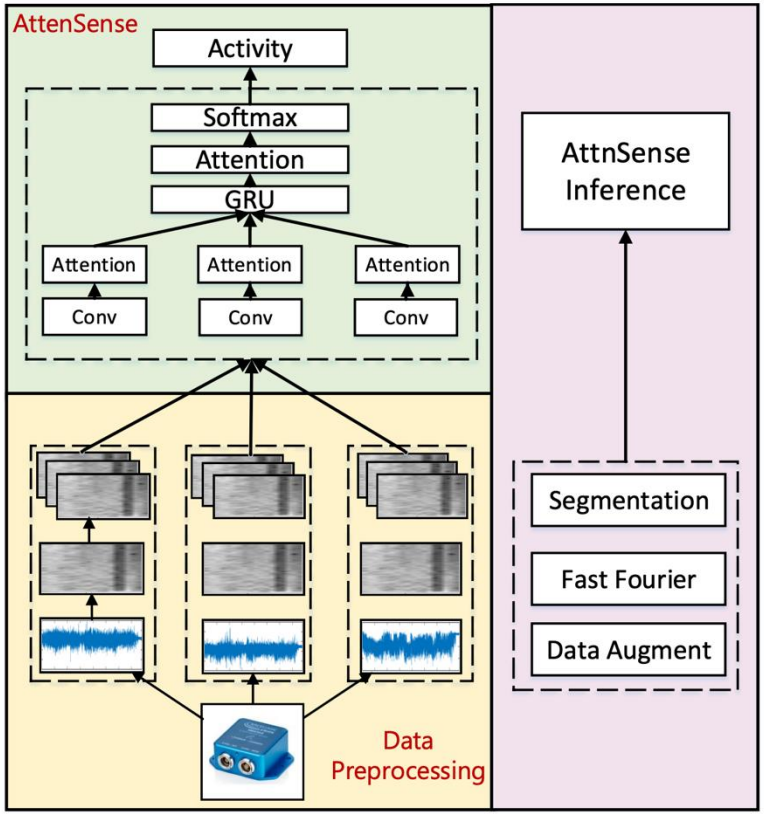
Single-modal HAR

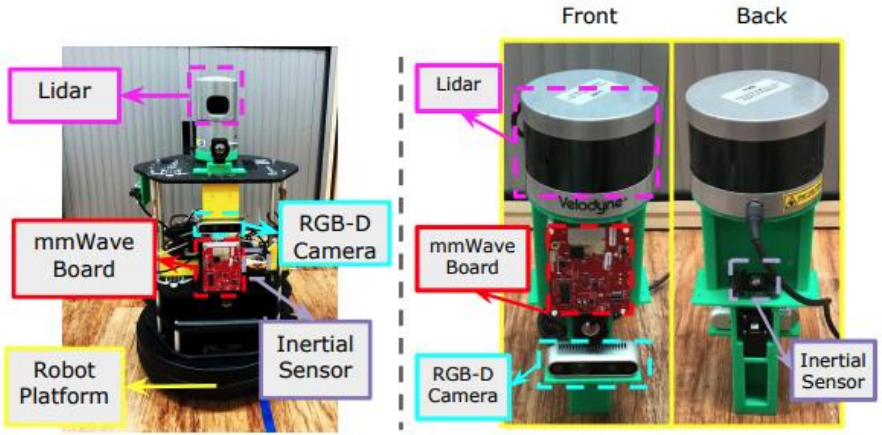


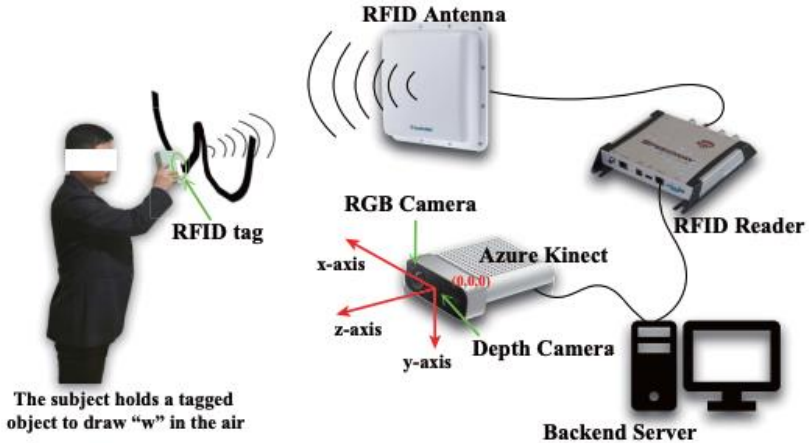


[1] LIMU-BERT: Unleashing the Potential of Unlabeled Data for IMU Sensing Applications; Xu et al.
[2] SelfHAR: Improving Human Activity Recognition through Self-training with Unlabeled Data; Tang et al.

Multi-modal HAR

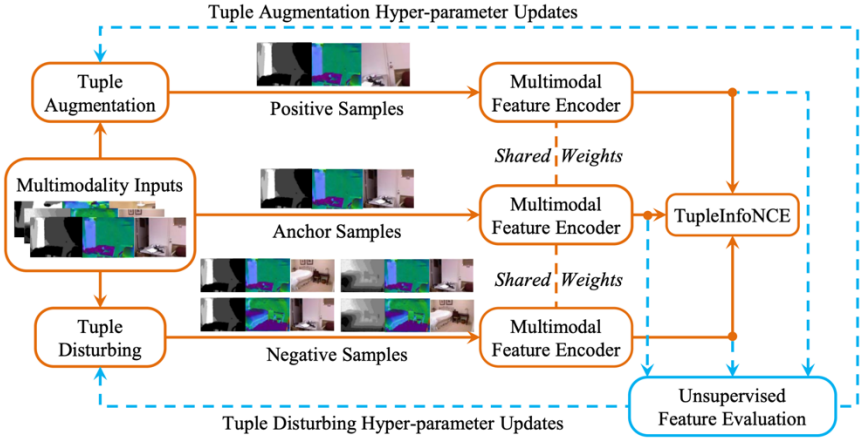


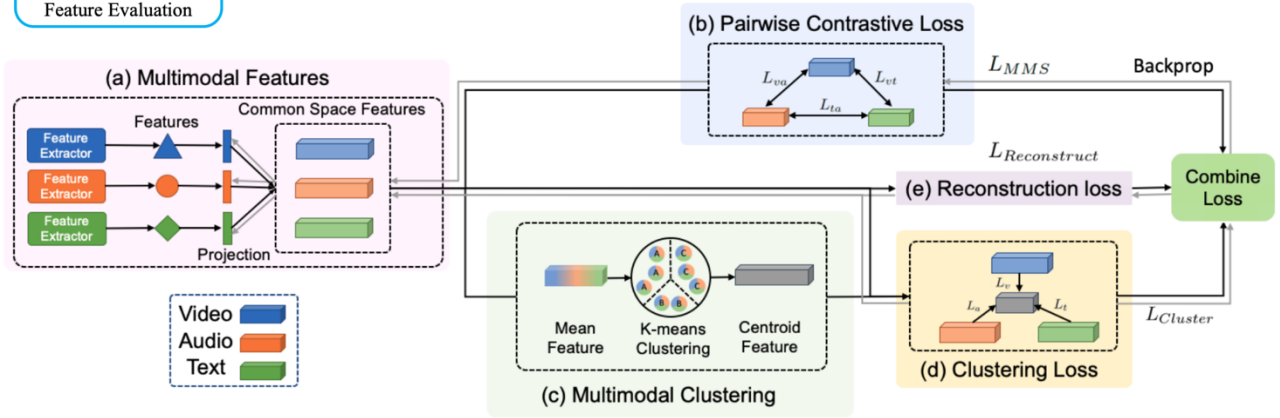




[1] RFID and Camera Fusion for Recognition of Human-object Interactions; Liu et al.
[2] milliEgo: Single-chip mmWave Radar Aided Egomotion Estimation via Deep Sensor Fusion; Lu et al.
[3] AttnSense: Multi-level Attention Mechanism For Multimodal Human Activity Recognition; Ma et al.

Multi-modal learning w/ limited labeled data





[1] Multimodal Clustering Networks for Self-supervised Learning from Unlabeled Videos; Chen et al.
[2] Contrastive Multimodal Fusion with TupleInfoNCE; Liu et al.

Challenges

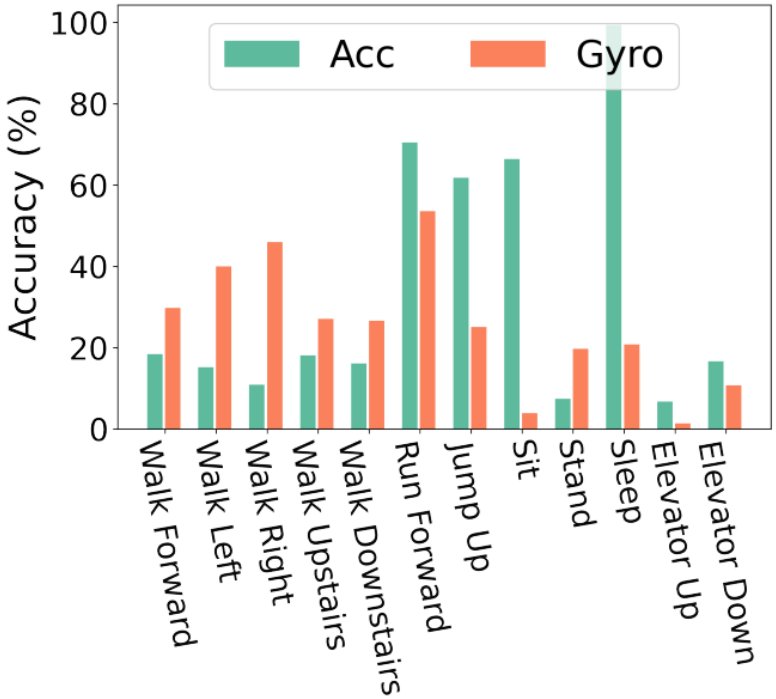
1. Heterogeneous info(e.g. Different dimensions) from different modalities about the same events
    a. Difficult Synchronization
    b. Difficult Fusion
2. Limited amount of labeled data for training(very labor-intensive for labeling multi-modalities data)
3. Privacy issue: can only process on-device and delete, can’t upload to cloud
4. Computational cost concern: on-device training for dynamic characteristics

Design

Key term for context

Complementary information

Acc is better for walking-related activities while Gyro is better for the rest



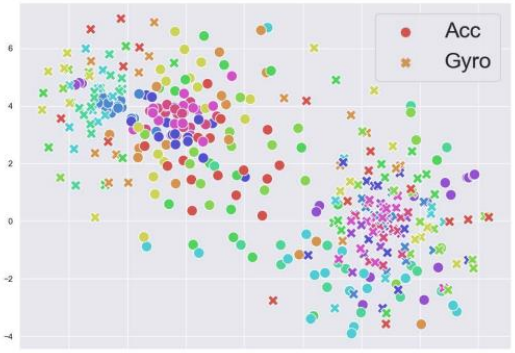
Key term for context

Complementary Information

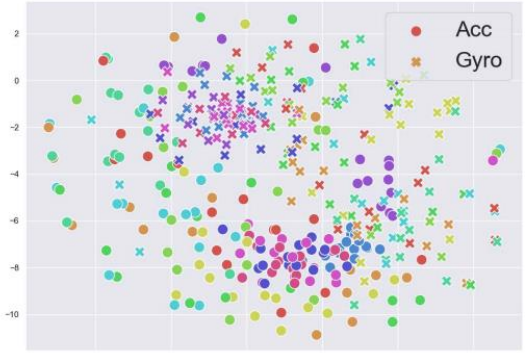
- Acc is better for walking-related activities; Gyro is better for the rest
- exploits strength of different sensors and promotes fusion performance

Consistent Information

- The model has learned to project both modalities into similar regions of feature space for the same underlying activity.
- helps align features, making the fusion more robust to noise



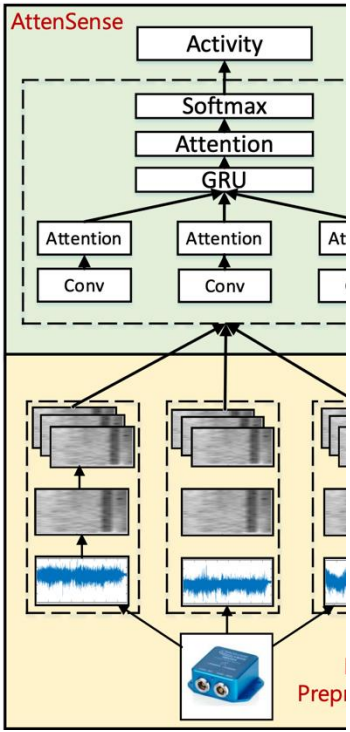
(a) ***Deepsense*** ($\bar{D} = 0.7288$).

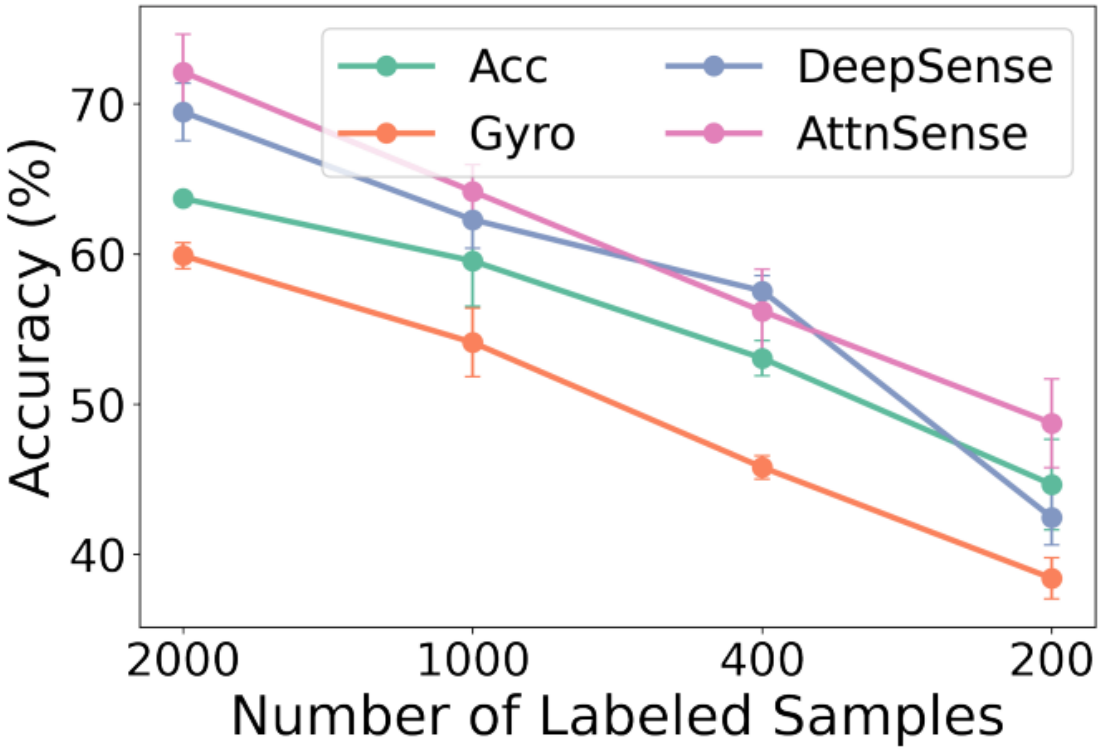


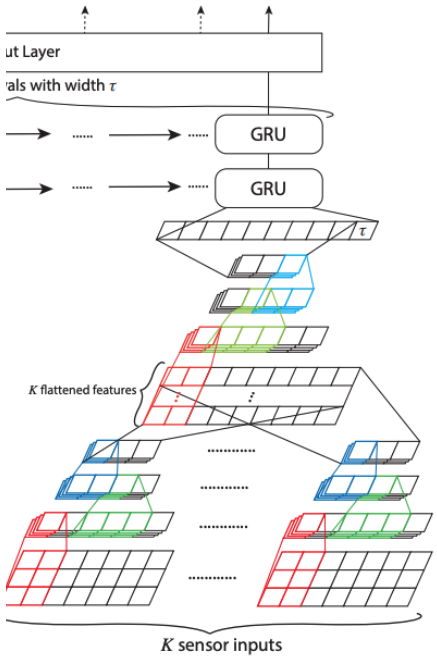
(b) ***Attnsense*** ($\bar{D} = 0.7685$).

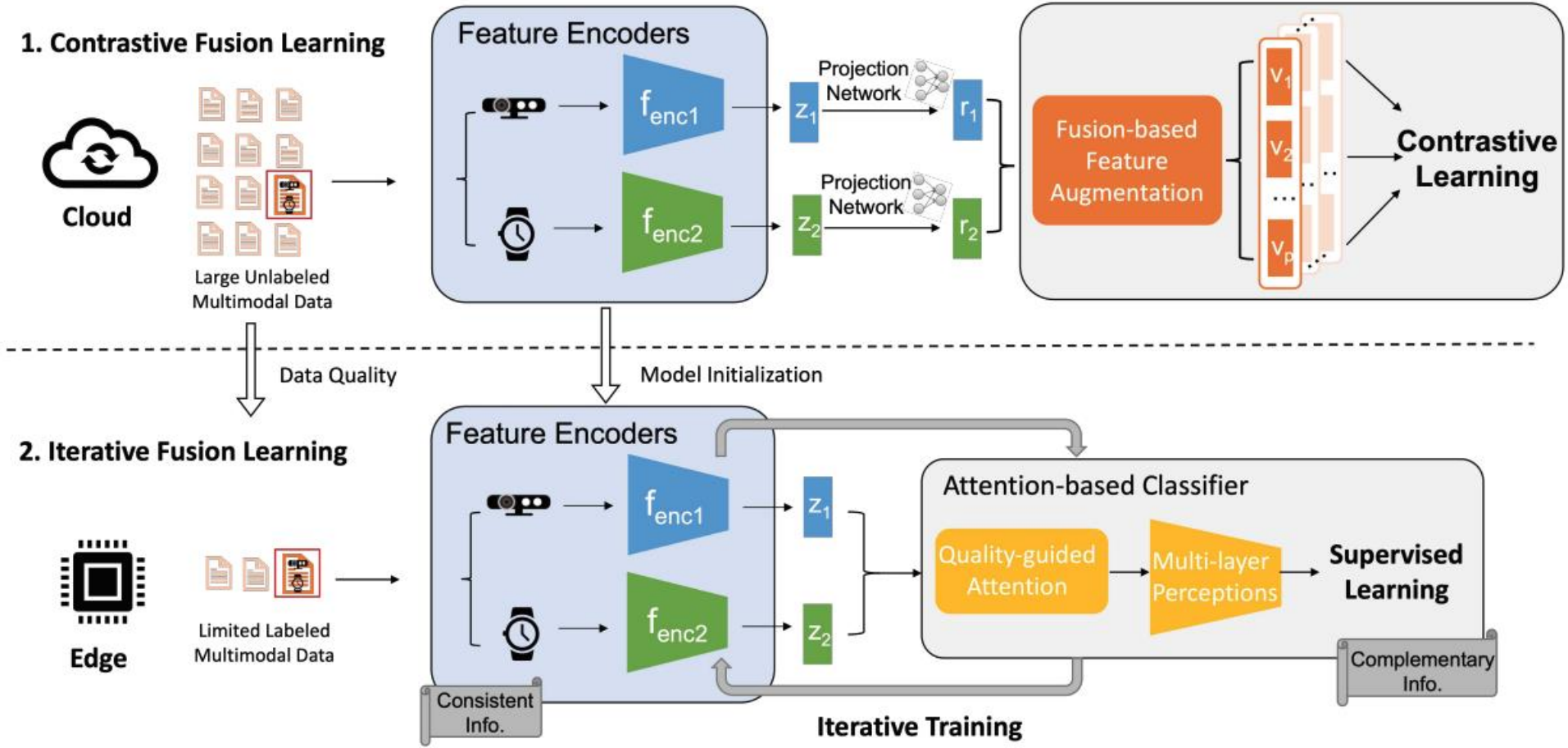
Figure 2: Visualization of Acc and Gyro features generated by different fusion approaches. Here $\bar{D}$ denotes the mean cosine distance between Acc and Gyro features. *Deepsense* learns more consistent information (the features of two modalities have a smaller mean distance and are more aligned), while *AttnSense* combines more complementary information.

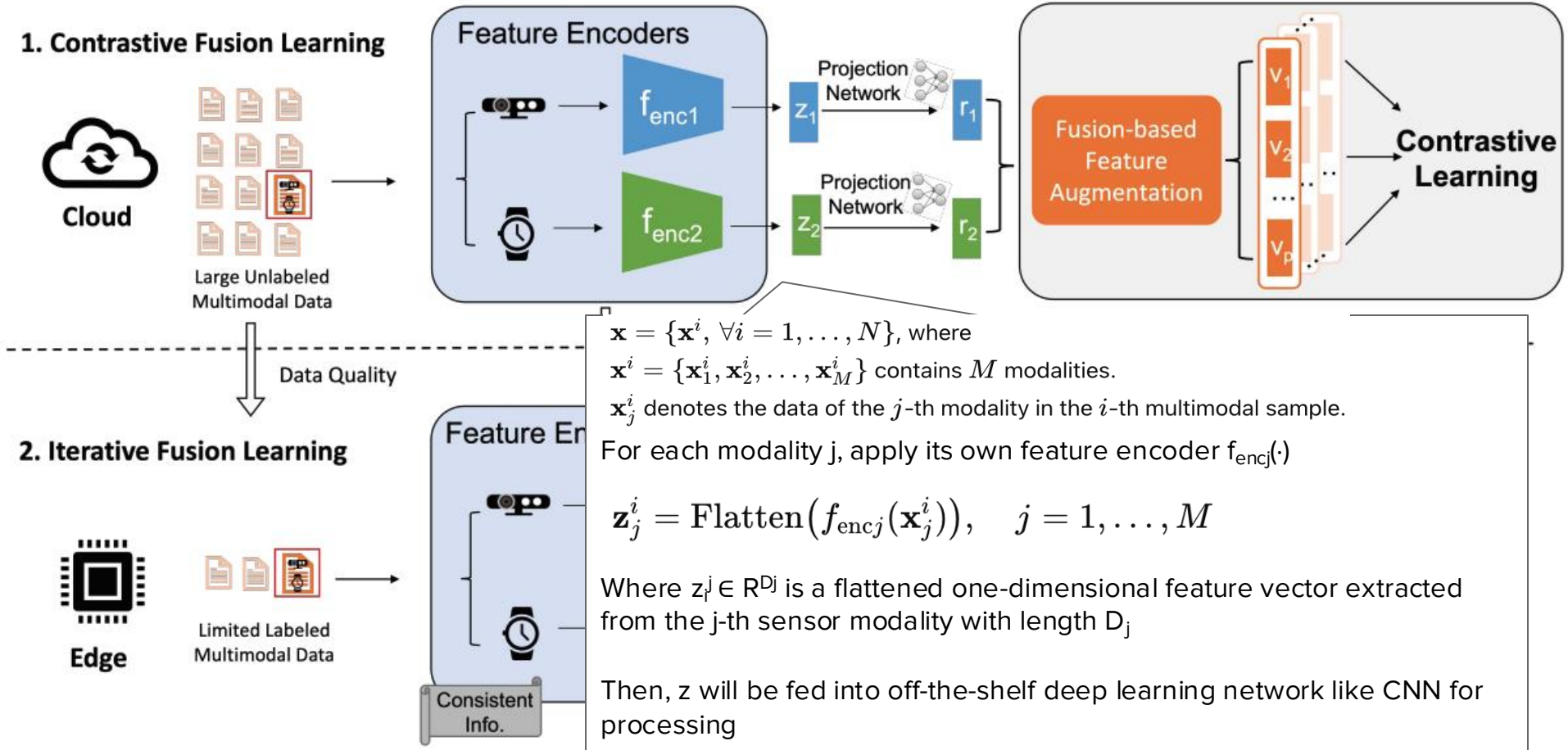
Key term for context








1. Contrastive Fusion Learning
Cloud
Large Unlabeled Multimodal Data
Feature Encoders
f_{enc1}
f_{enc2}
z_1
z_2
Projection Network
Projection Network
r_1
r_2
Fusion-based Feature Augmentation
v_1
v_2
v_p
Contrastive Learning
Data Quality
Model Initialization
2. Iterative Fusion Learning
Edge
Limited Labeled Multimodal Data
Feature Encoders
f_{enc1}
f_{enc2}
z_1
z_2
Attention-based Classifier
Quality-guided Attention
Multi-layer Perceptions
Supervised Learning
Consistent Info.
Complementary Info.
Iterative Training



$\mathbf{x} = \{\mathbf{x}^i, \forall i = 1, \ldots, N\}$, where

$\mathbf{x}^i = \{\mathbf{x}_1^i, \mathbf{x}_2^i, \ldots, \mathbf{x}_M^i\}$ contains M modalities.

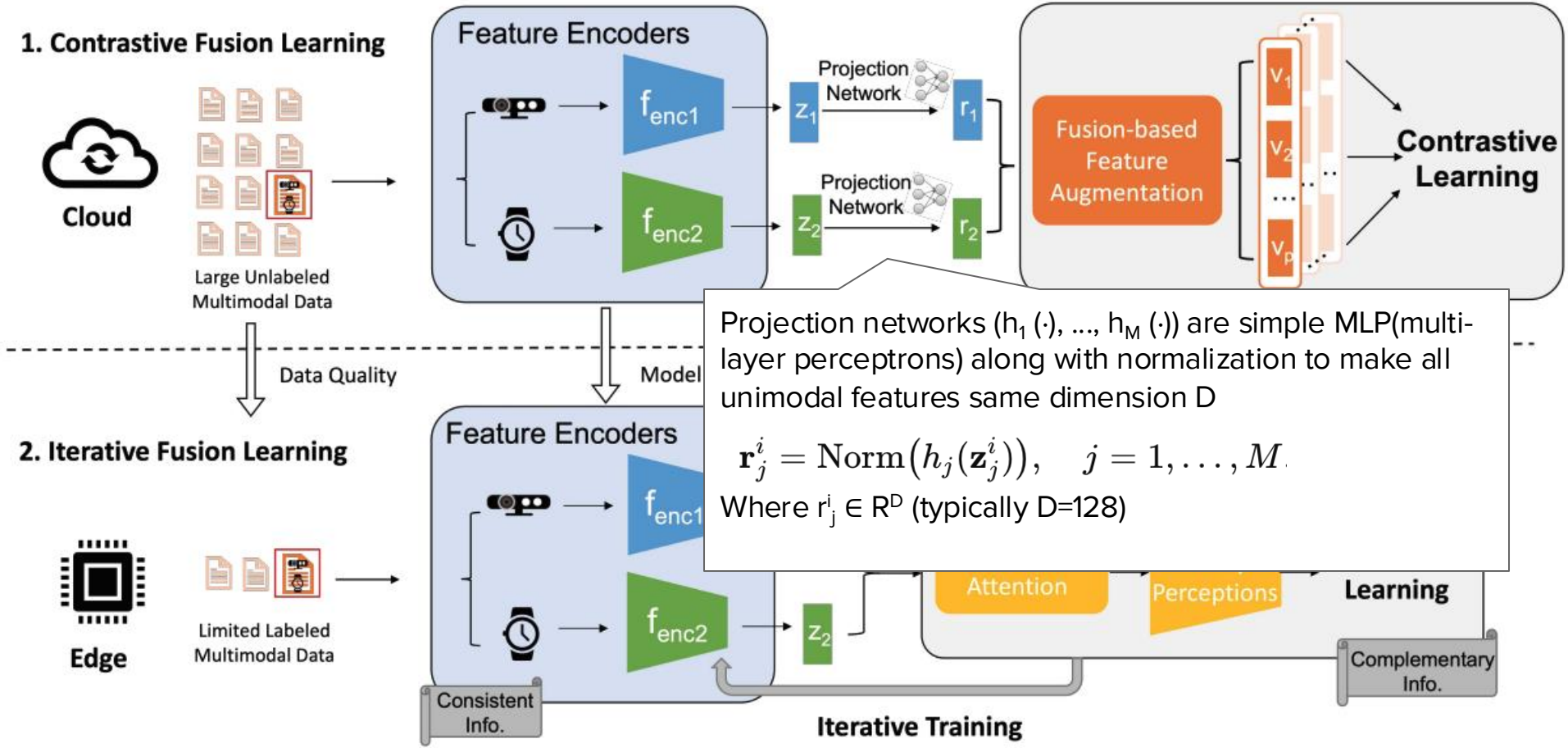
$\mathbf{x}_j^i$ denotes the data of the j-th modality in the i-th multimodal sample.

For each modality j, apply its own feature encoder $f_{encj}(\cdot)$

$$\mathbf{z}_j^i = \mathrm{Flatten}\left(f_{\mathrm{enc}j}(\mathbf{x}_j^i)\right), \quad j = 1, \ldots, M$$

Where $z_l^j \in R^{Dj}$ is a flattened one-dimensional feature vector extracted from the j-th sensor modality with length D_j

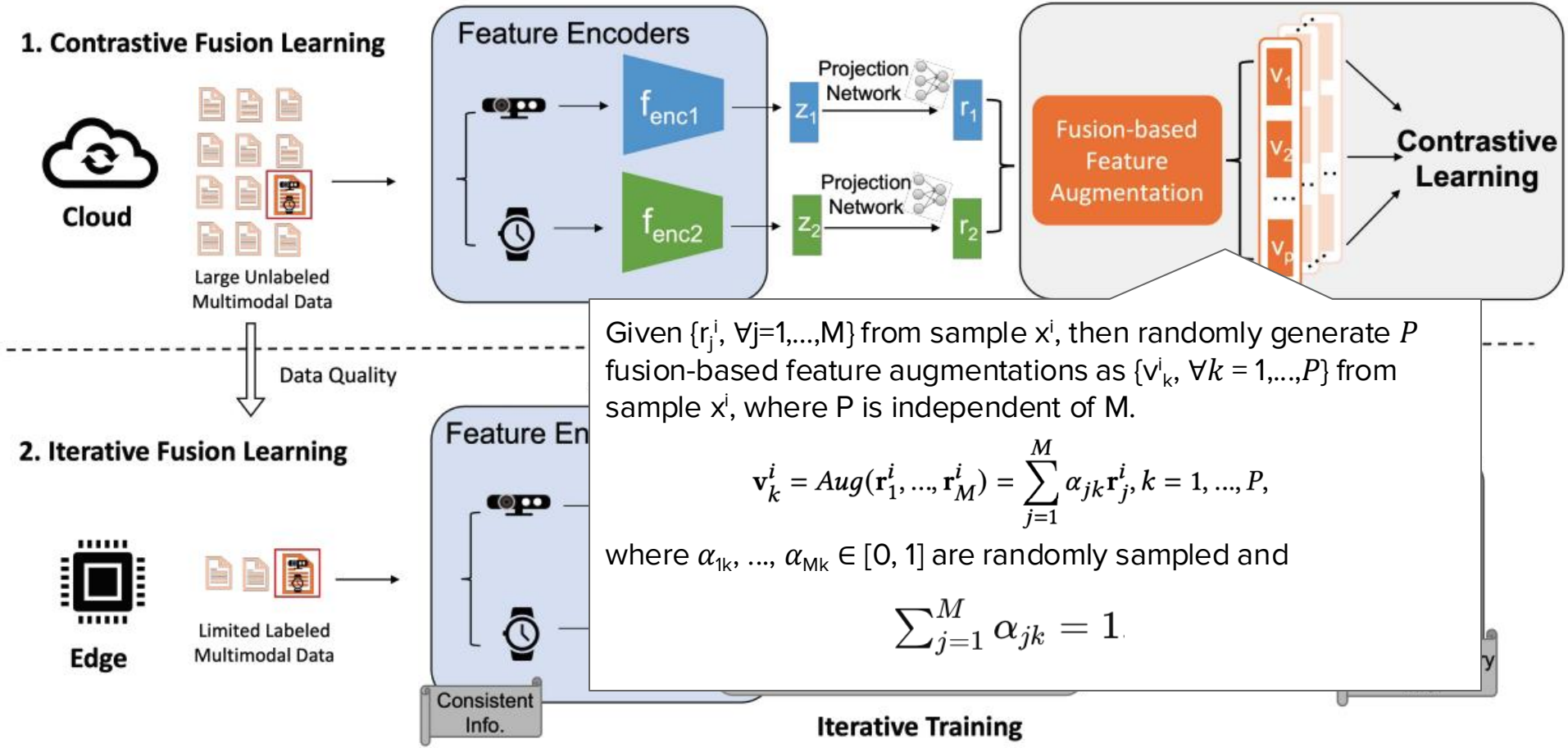
Then, z will be fed into off-the-shelf deep learning network like CNN for processing



Projection networks ($h_1(\cdot)$, ..., $h_M(\cdot)$) are simple MLP(multi-layer perceptrons) along with normalization to make all unimodal features same dimension D

$$\mathbf{r}_j^i = \mathrm{Norm}\left(h_j(\mathbf{z}_j^i)\right), \quad j = 1, \ldots, M$$

Where $r^i_j \in R^D$ (typically D=128)



Given $\{r_j^i, \forall j=1,...,M\}$ from sample x^i, then randomly generate P fusion-based feature augmentations as $\{v^i_k, \forall k = 1,...,P\}$ from sample x^i, where P is independent of M.

$$\mathbf{v}_k^i = Aug(\mathbf{r}_1^i, ..., \mathbf{r}_M^i) = \sum_{j=1}^{M} \alpha_{jk}\mathbf{r}_j^i, k = 1, ..., P,$$

where $\alpha_{1k}, ..., \alpha_{Mk} \in [0, 1]$ are randomly sampled and

$$\sum_{j=1}^{M} \alpha_{jk} = 1.$$

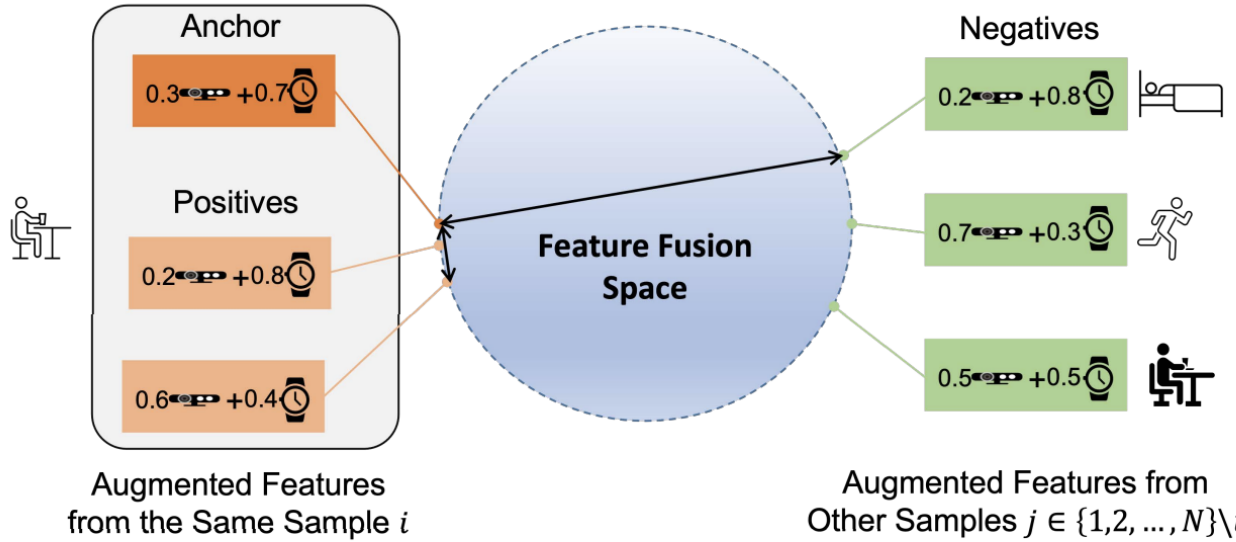
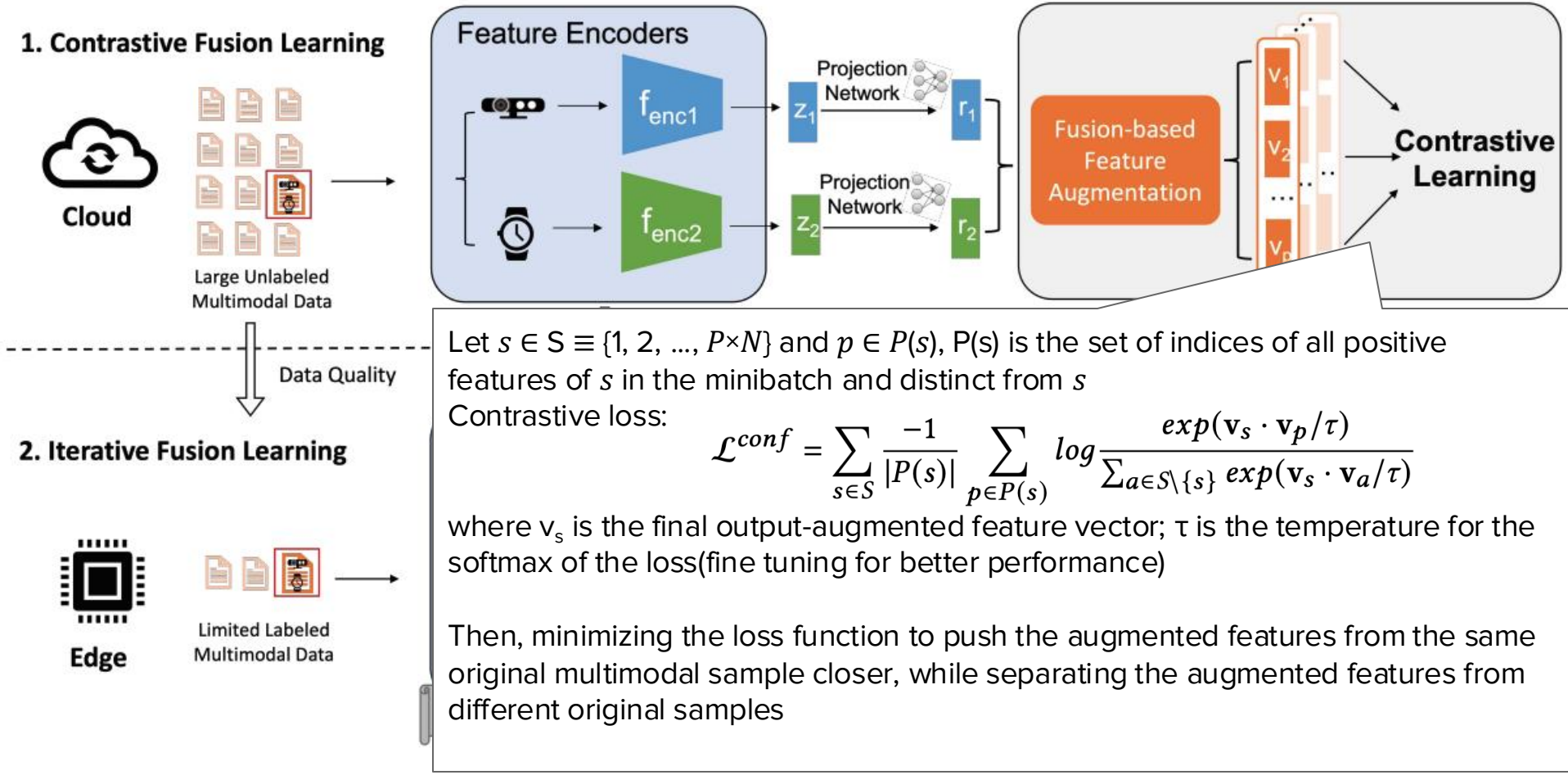


Figure 6: Illustration of fusion-based contrastive learning on the normalized feature space. The positive features are generated by sampling different weighted combinations of modalities from the same multimodal sample, while the negatives are augmented from the remaining multimodal samples in the batch. The contrastive fusion loss contrasts the positives to be closer to each other and pushes away the negative features.



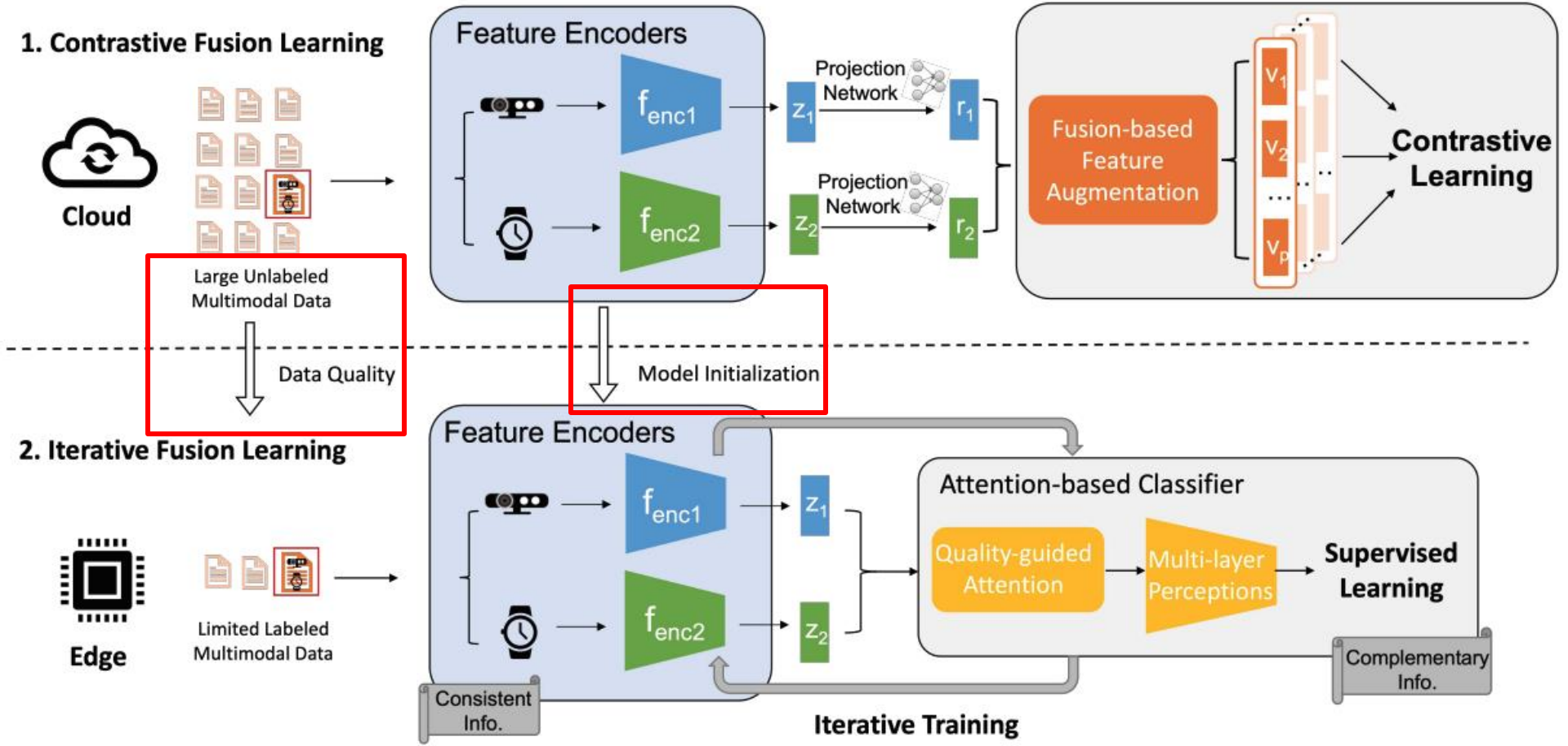
Let $s \in S \equiv \{1, 2, ..., P \times N\}$ and $p \in P(s)$, P(s) is the set of indices of all positive features of s in the minibatch and distinct from s

Contrastive loss:

$$\mathcal{L}^{conf} = \sum_{s \in S} \frac{-1}{|P(s)|} \sum_{p \in P(s)} log \frac{exp(\mathbf{v}_s \cdot \mathbf{v}_p / \tau)}{\sum_{a \in S \setminus \{s\}} exp(\mathbf{v}_s \cdot \mathbf{v}_a / \tau)}$$

where v_s is the final output-augmented feature vector; τ is the temperature for the softmax of the loss(fine tuning for better performance)

Then, minimizing the loss function to push the augmented features from the same original multimodal sample closer, while separating the augmented features from different original samples


1. Contrastive Fusion Learning
Cloud
Large Unlabeled Multimodal Data
Data Quality
Feature Encoders
f_{enc1}
f_{enc2}
z_1
z_2
Projection Network
Projection Network
r_1
r_2
Fusion-based Feature Augmentation
v_1
v_2
v_p
Contrastive Learning
Model Initialization
2. Iterative Fusion Learning
Edge
Limited Labeled Multimodal Data
Feature Encoders
f_{enc1}
f_{enc2}
z_1
z_2
Attention-based Classifier
Quality-guided Attention
Multi-layer Perceptions
Supervised Learning
Complementary Info.
Consistent Info.
Iterative Training

1. Contrastive

$$\mu_j = tanh(\mathbf{W} \cdot \mathbf{z_j} + \mathbf{b}),$$

$$\beta(Attn)_j = \frac{exp(\mu_j \cdot \mathbf{z_j})}{\sum_j exp(\mu_j \cdot \mathbf{z_j})}, j = 1, ..., M$$

z_j: the actual feature vector from the encoder.

μ_j: a learned transformation of z_j that acts like a "query" for its importance.

$\beta(Attn)_j$: the normalized importance weight for modality j

$$q_j = \frac{H(x_j)}{c_j}, \quad \beta_j^{\text{QoM}} = \frac{q_j}{\sum_j q_j} \qquad \beta(QoM)_j = q_j / \sum_{j=1}^{M} q_j$$

$H(x_j)$: Hopkins statistic, a statistical metric between 0 and 1 for clusterability

c_j: the absolute difference between the number of clusters and the ground truth

$\beta(Attn)_j$: the normalized quality weights for modality j

Cloud

Contrastive Learning

2. Iterative Fusion Learning

Edge

Limited Labeled Multimodal Data

Feature Encoders

f_{enc1}

f_{enc2}

z_1

z_2

Attention-based Classifier

Quality-guided Attention

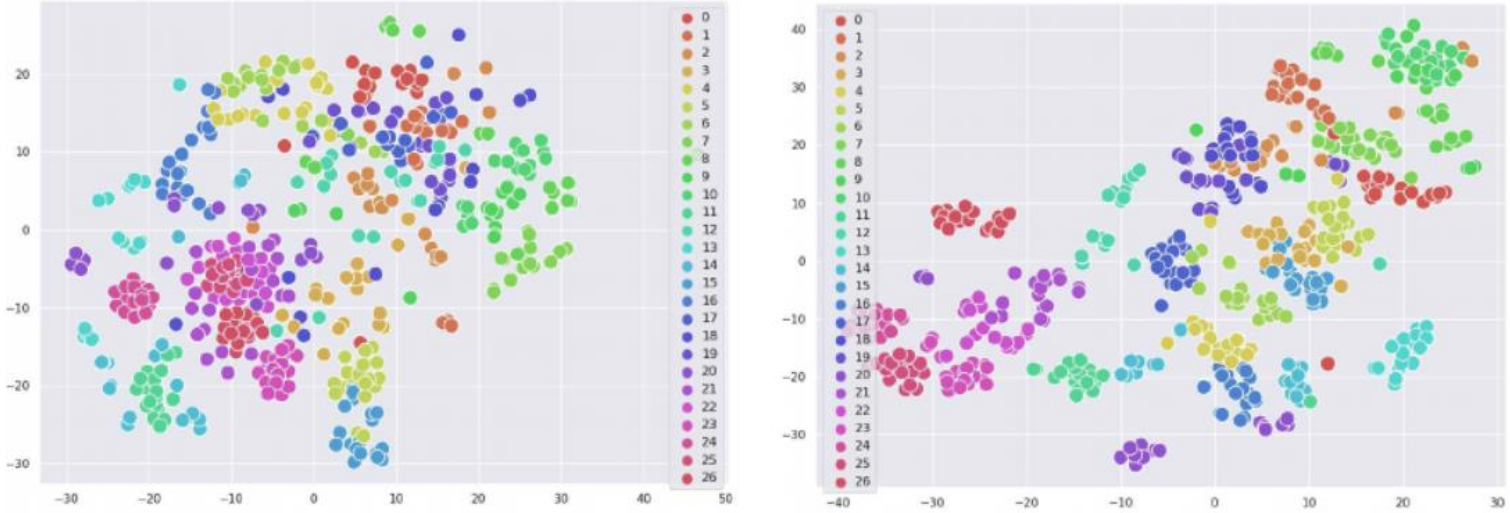
Multi-layer Perceptions

Supervised Learning

Complementary Info.

Consistent Info.

Iterative Training



(a) IMU ($q_1 = 0.26$). (b) Depth ($q_2 = 0.74$).

Figure 7: Measuring data quality from unlabeled data. $q_j (j = 1, 2)$ is the calculated quality weight. Compared with IMU, the depth data has a higher clusterability (0.8513), and the optimal number of clusters (24) is closer to the number of total classes (27).

$\beta_j = (1-\lambda)\beta(Attn)_j + \lambda\beta(QoM)_j$ $\beta_j = \beta_j / \sum_{j=1}^{M} \beta_j$

λ: tunable hyper-parameter to adjust the impact of quality-based weights

β_j: the normalized combined weights for modality j

Fusion Mechanism:

For sensors of similar modalities (e.g. accelerator and gyroscope) and when the unimodal features have the same dimension:

$$\mathbf{v}^i = SumAttn(\mathbf{z}_1^i, ..., \mathbf{z}_M^i) = \sum_{j=1}^{M} \beta_j \mathbf{z}_j^i$$

For sensors of extremely diverse modalities (e.g., the depth camera and IMU) or when the unimodal features have different dimensions:

$$\mathbf{v}^i = ConcateAttn(\mathbf{z}_1^i, ..., \mathbf{z}_M^i) = [\beta_1 \mathbf{z}_1^i, ..., \beta_M \mathbf{z}_M^i]$$

1. Contrastive

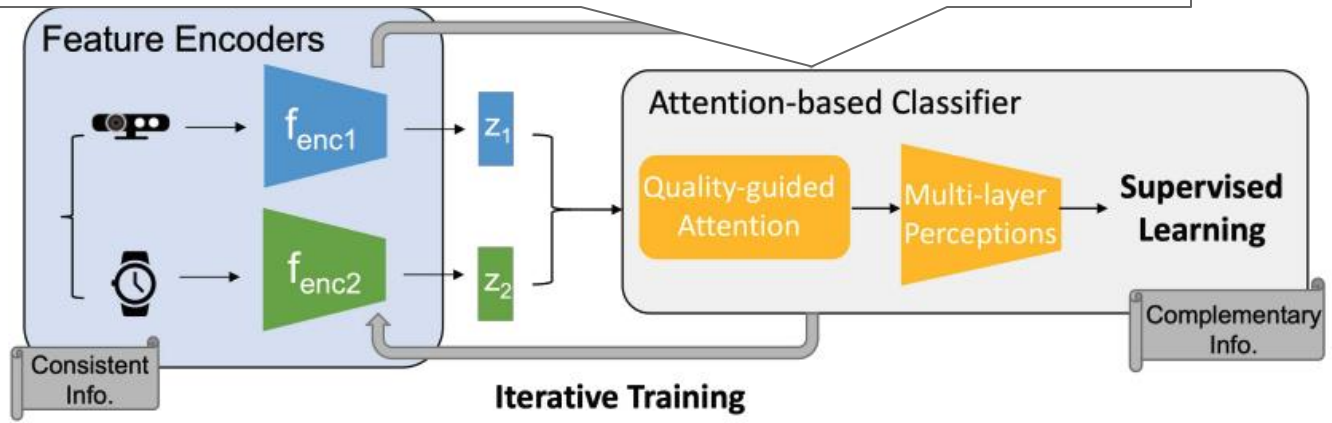
Cloud

Contrastive Learning

2. Iterative Fusion Learning

Edge

Limited Labeled Multimodal Data

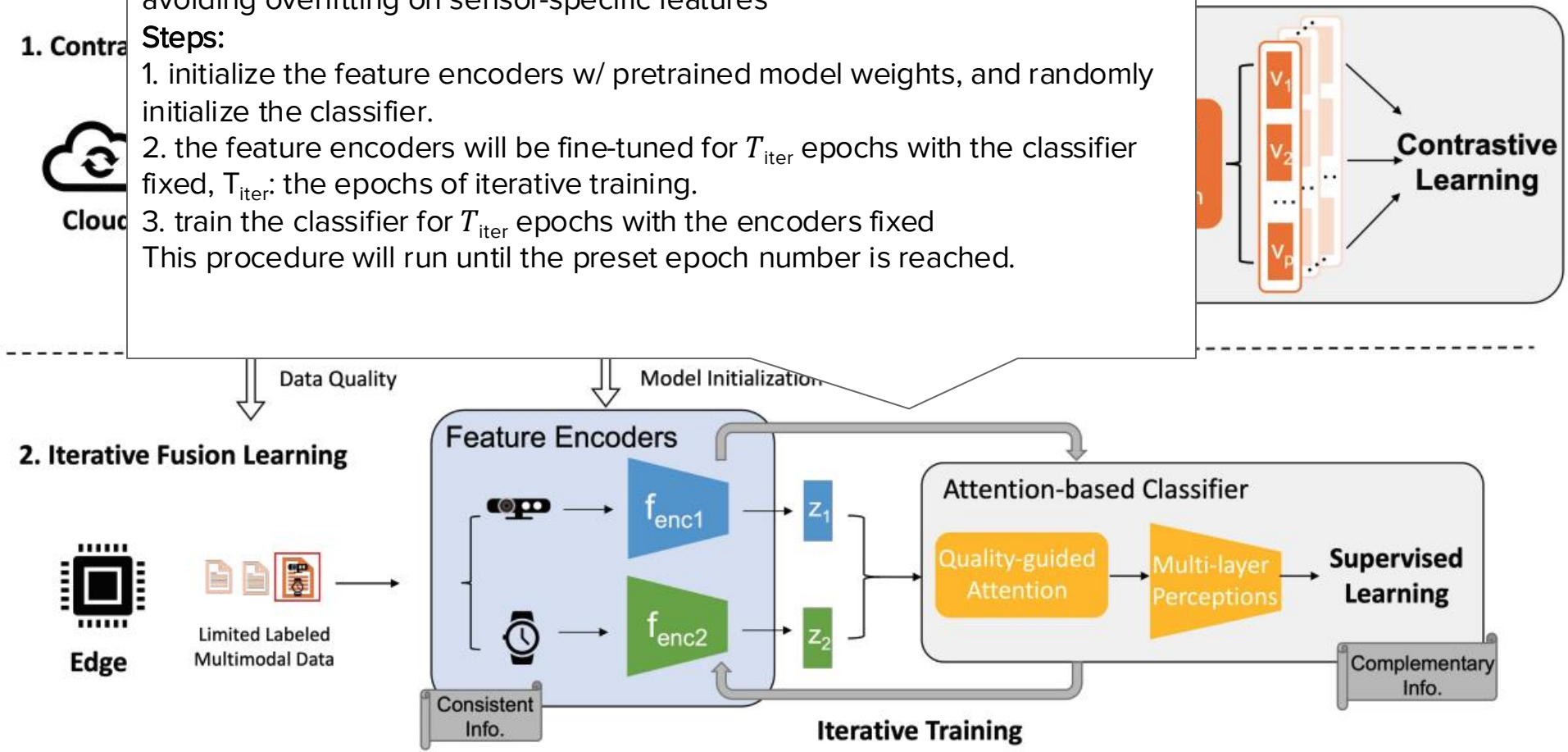


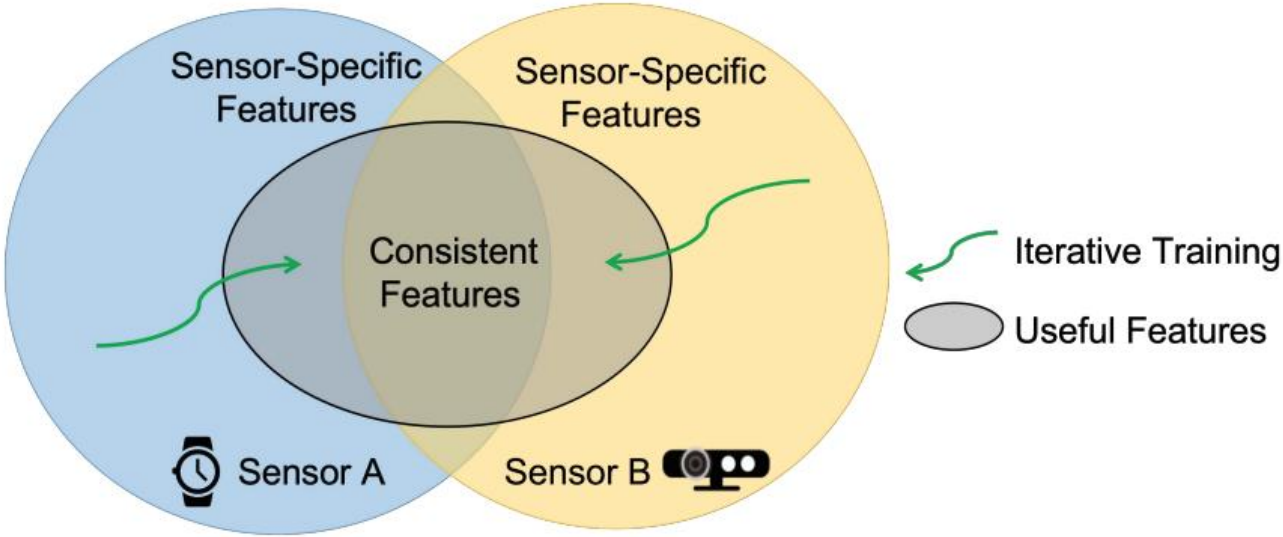
Goal: explore complementary information from labeled multimodal data while avoiding overfitting on sensor-specific features

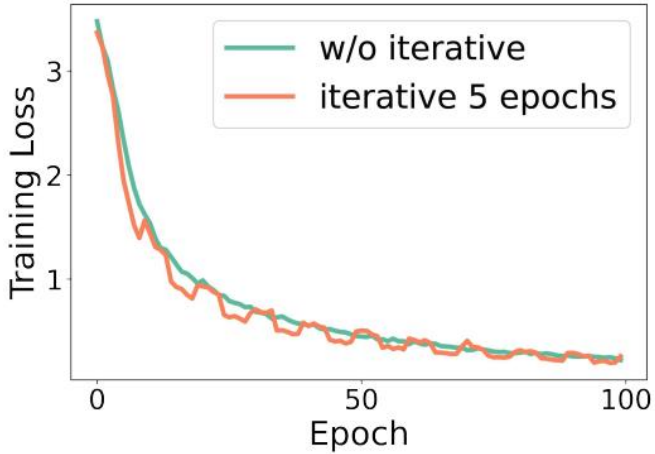
Steps:

1. initialize the feature encoders w/ pretrained model weights, and randomly initialize the classifier.
2. the feature encoders will be fine-tuned for T_{iter} epochs with the classifier fixed, T_{iter}: the epochs of iterative training.
3. train the classifier for T_{iter} epochs with the encoders fixed

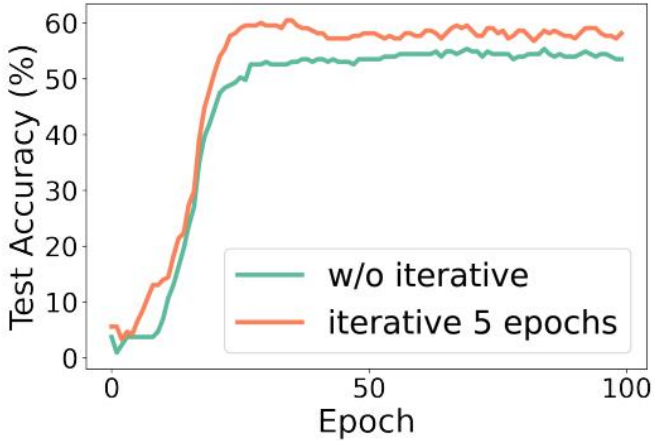
This procedure will run until the preset epoch number is reached.







(a) Training Loss.



(b) Testing Accuracy

Evaluations

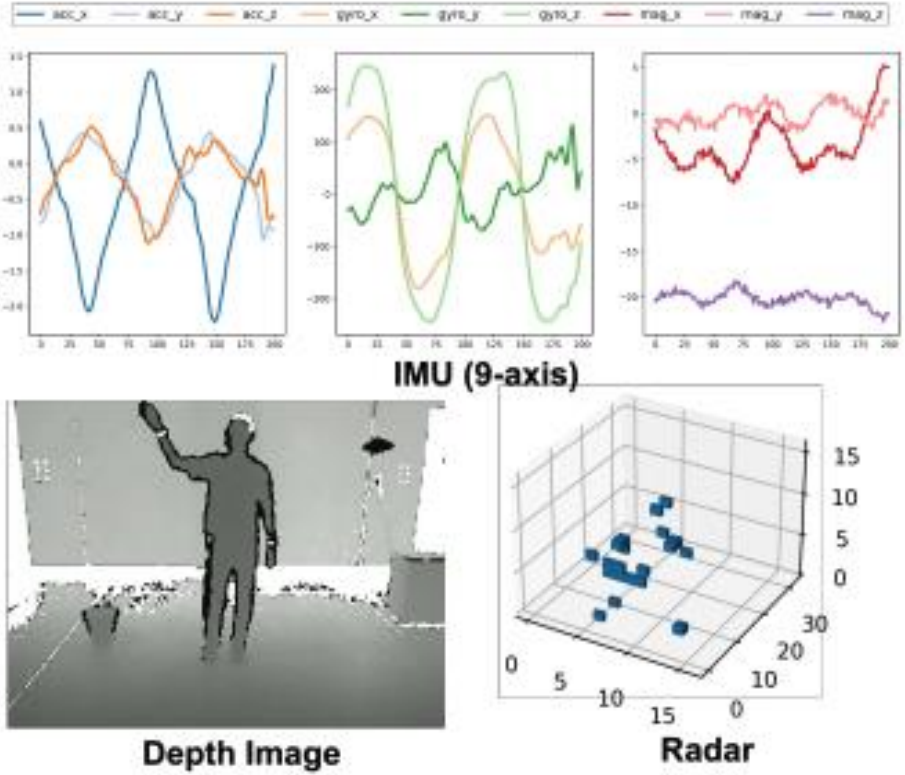
Setups

1. Python & Pytorch,
   a. 0.01 lr, 64 bs for contrastive learning; 0.01 lr, 16 bs for supervised learning
2. Cloud:
   a. 8 NVIDIA GEFORCE TITAN Xp GPUs
   b. 256 GB RAM
   c. two 16-core Intel Xeon E5-2620 (2.10GHz) CPUs
3. Edge: NVIDIA Jetson TX2
   a. 256-core NVIDIA Pascal™ architecture GPU
   b. Dual-core NVIDIA Denver™ 2 64-bit CPU and quad-core Arm® Cortex®-A57 MPCore processor
   c. 8GB 128-bit LPDDR4(59.7GB/s)
4. Baselines: SingleModal, DeepSense, AttnSense, Contrastive Predictive Coding (CPC), Contrastive Multi-view Learning (CMC)
5. Datasets:

Dataset	Modality	Activity	Subject	Samples
USC	Acc, Gyro	12	14	38312
UTD	IMU, Skeleton	27	8	864
Self-Collected	IMU, Depth, Radar	14	30	3434

Setups - self-collected datasets

(a) Experiment Setting.



(b) Collected Data.

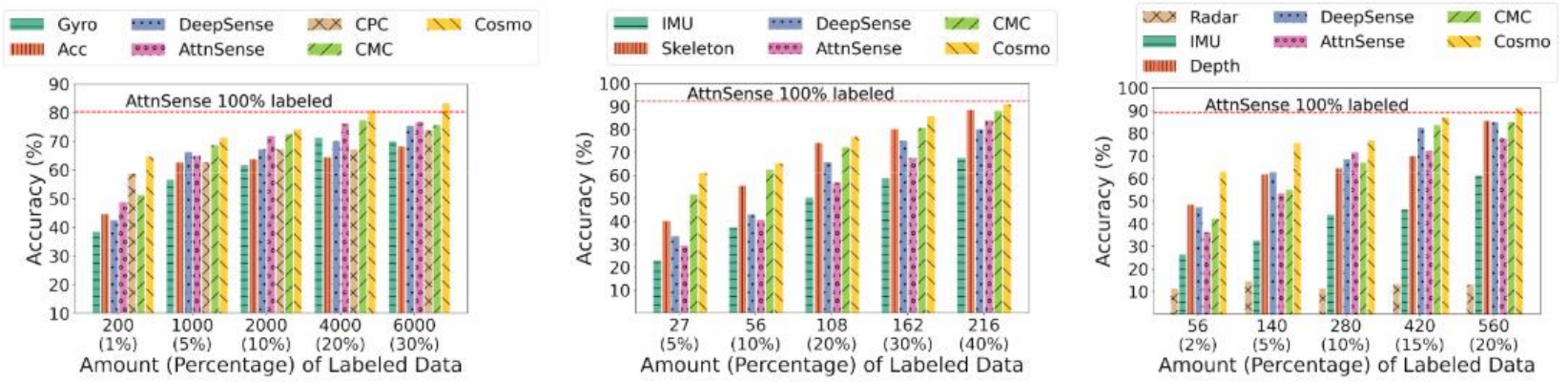


Figure 11: Accuracy comparison with different amounts of labeled data. Cosmo consistently outperforms other baselines, and can achieve comparable accuracy of AttnSense (with 100% labeled data) with only a small portion of labeled data.

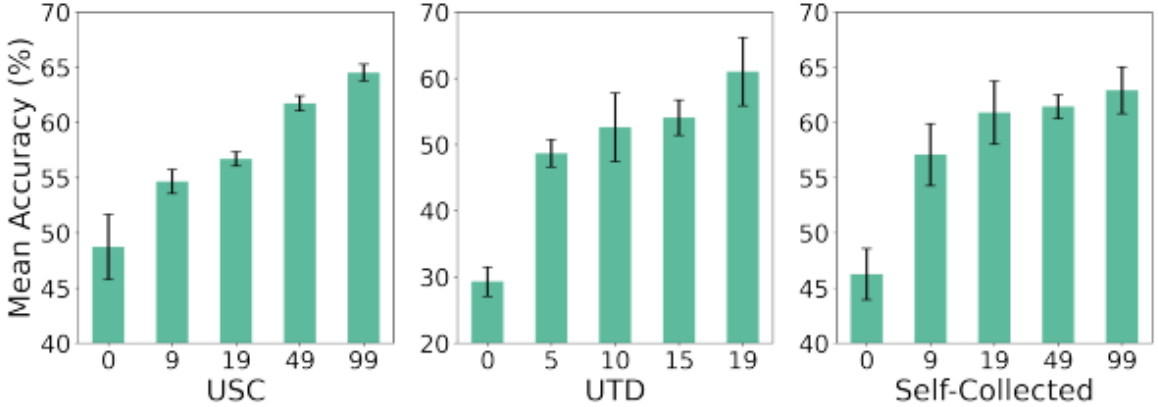
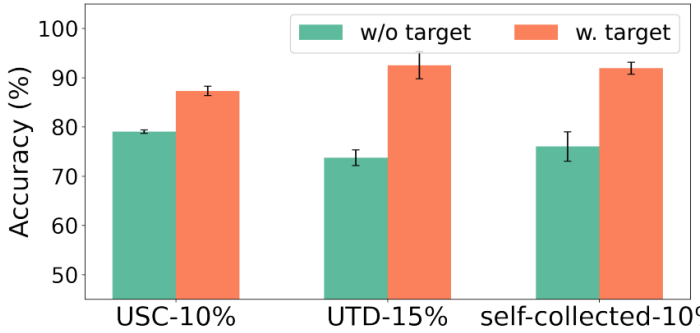


Figure 12: Accuracy of Cosmo with different amounts of unlabeled data. Values in X-axis are $\frac{\#Unlabeled\ data}{\#Labeled\ data}$.

Figure 12: Under the condition that num of labeled data is fixed

USC: 1%, 200
UTD: 5%, 27
Self-Collected: 2%, 56



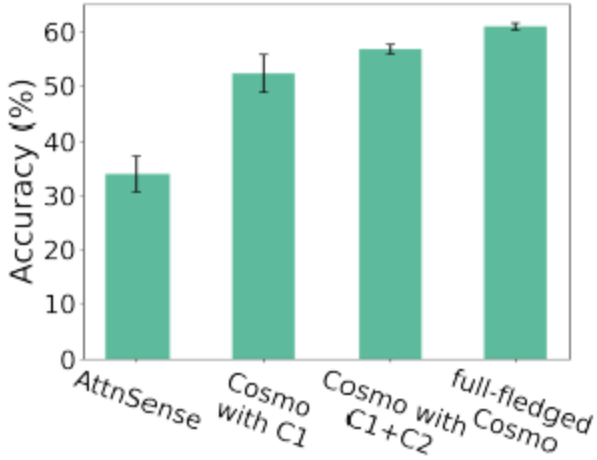
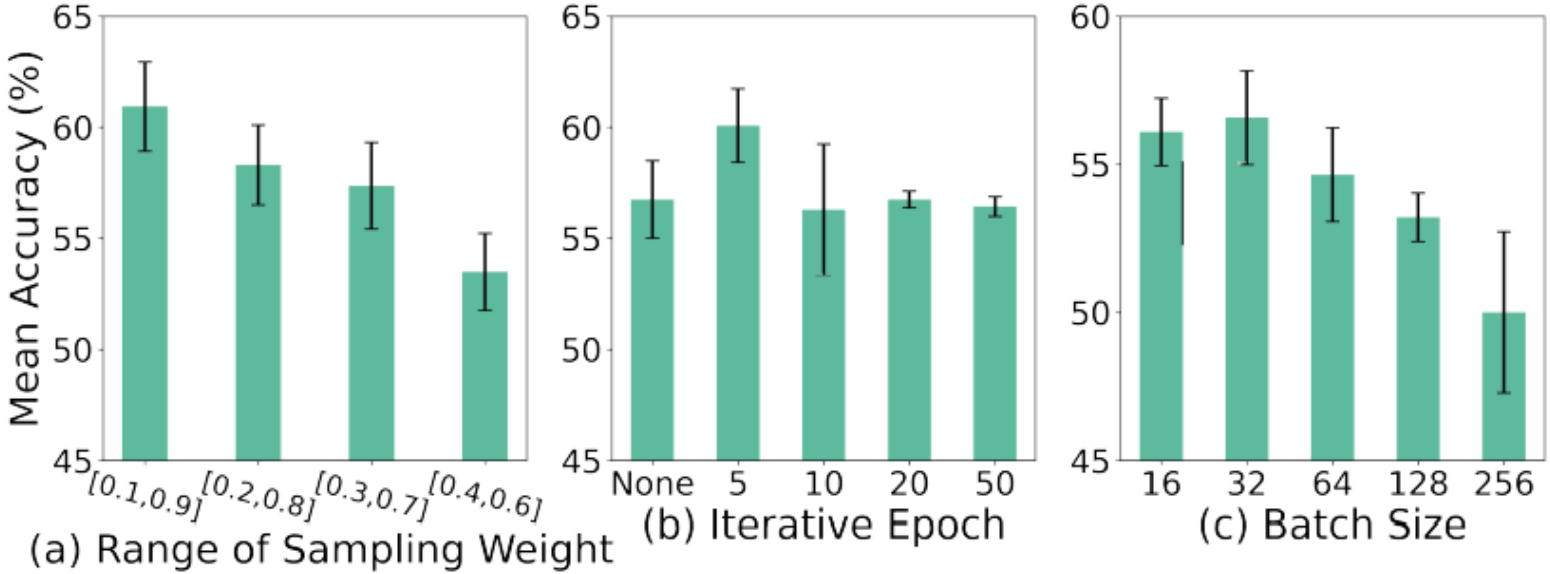
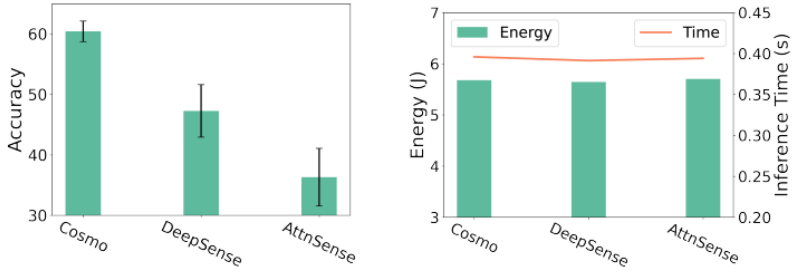


Figure 13: Accuracy with labeled data from target subjects.



Approach	Label rate	Cosmo Stage 1	Cosmo Stage 2	*DeepSense*	*AttnSense*
Time (min)	1%	101.19	**25.93**	38.38	62.52
	2%	90.13	49.87	74.42	120.98
Energy (KJ)	1%	7286.07	**21.34**	30.99	51.71
	2%	6489.70	40.35	60.90	98.58

Table 3: Training overhead. The first stage of Cosmo is trained on the server. The others are trained on Jetson TX2.



(a) Accuracy. **(b) Energy and time.**

Figure 17: Inference performance on Jetson TX2.

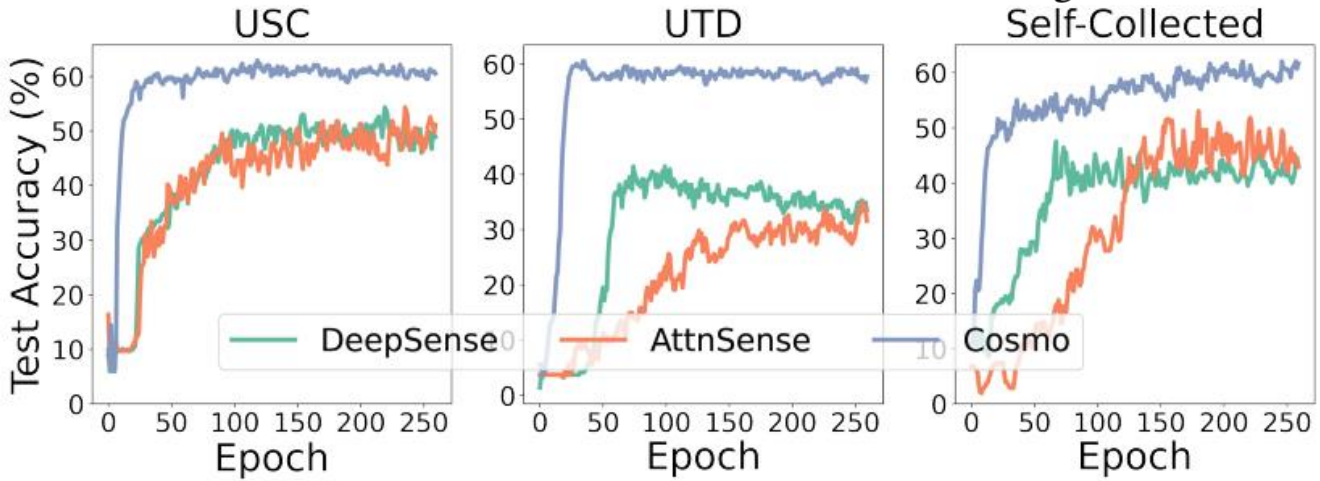


Figure 14: Convergence comparison on limited labeled data. Cosmo converges faster than supervised learning baselines.

Future Works & Opinions

Limitations & Future Works

1. Filtering techniques for throwing out irrelevant information
2. Synchronized errors-leverage the consistent information across the features of different modalities to align the multimodal data stream.
3. Faster inference
    a. Cache data from different sensors to enable faster convergence
    b. Dynamic sensor selection
4. Federated learning for further performance improvement and privacy protection in stage 1

Opinions

1. Question mark for the SOTA analysis
2. Strong assumption on the data availability
3. Paper writing is not perfectly clear
    a. Data quality confusion
    b. Missing loss function statement

Perusall