JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

山东大学 人工智能研究中心

TIME 机器学习与数据挖掘实验室

MetaViewer: Towards A Unified Multi-View Representation

Ren Wang[§], Haoliang Sun[§], Yuling Ma[∮], Xiaoming Xi[∮], Yilong Yin[§,∮]

§ Shandong University, China. ∮ Shandong Jianzhu University, China.

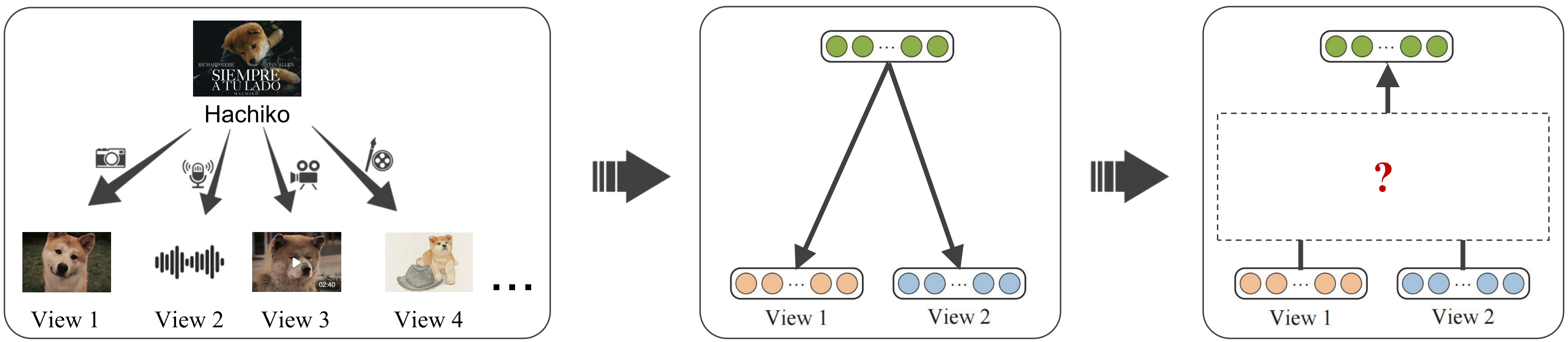
Poster: Wed-AM-320

Project page: https://xxlifelover.github.io/MetaViewerProjectPage/

Summary

Topic—Multi-view representation learning

learning a unified entity representation from its multiple observable views, which is critical for solving downstream tasks.



Real-world entity and its multiple views

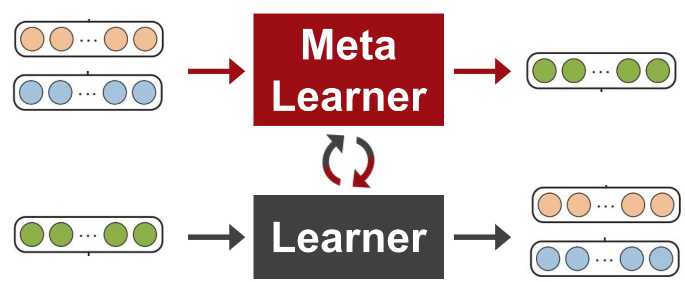
Intrinsic relation: from uniform to specific

Task: from specific to uniform

Existing methods

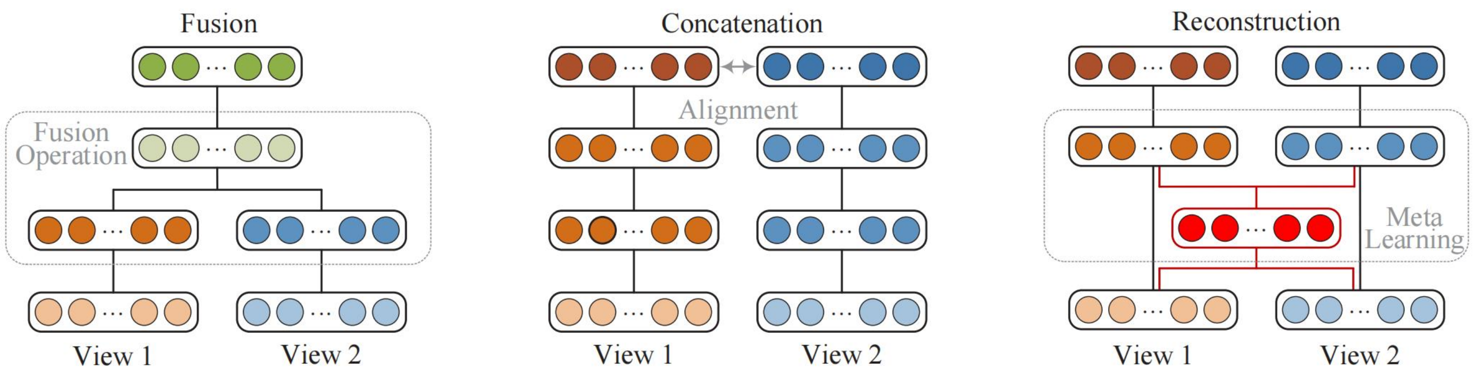
specific-to-unifrom (*S2U*) pipeline

Our MetaViewer



unifrom-to-specific (*U2S*) pipeline

Ⅰ. Backgorund



(a) Fusion-based (b) Alignment-based (c) MetaViewer

Exitsting methods (a) and (b) follow the *S2U* pipeline, where the unified representation is obtained by fusing or concatenating view-specific features. They suffer from

1) manually pre-specified fusion strategies;
2) meaningless view-private information.

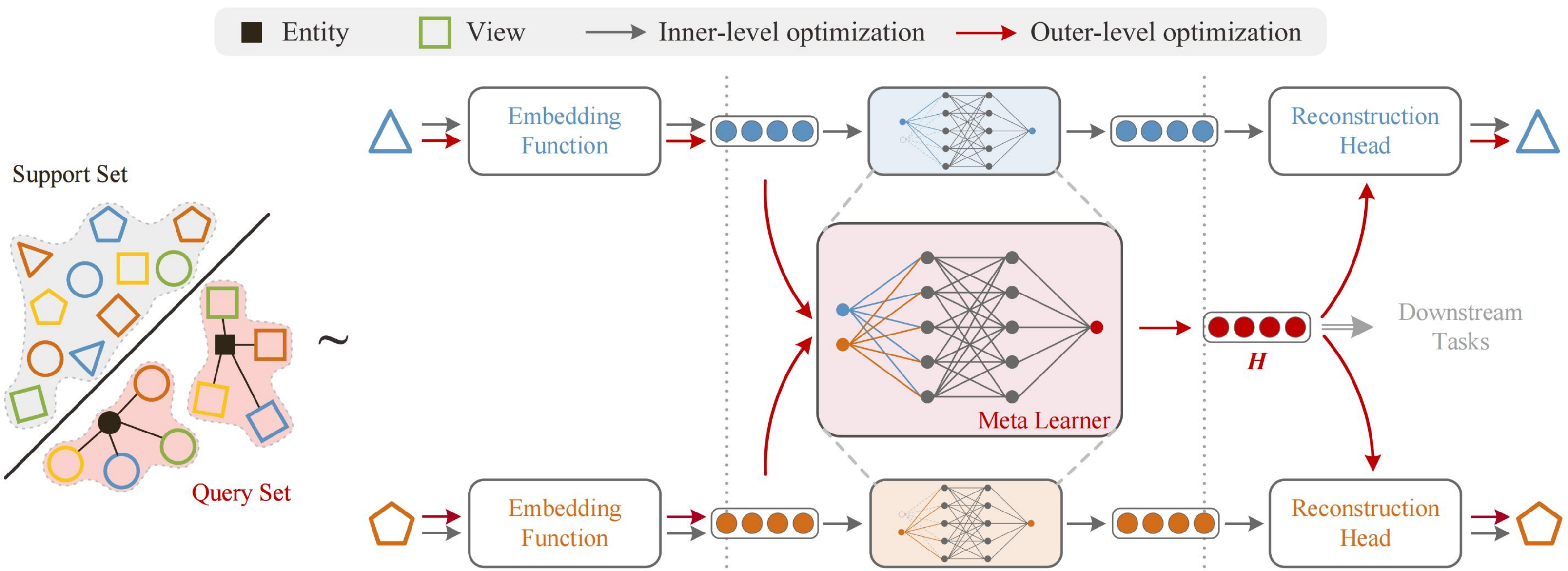
MetaViewer follows a novel *U2S* pipeline, which builds a meta-learner to learn "how to transform from a unified representations to view-specific features". It

1) meta-learns a data-oriented fusion manner;
2) decouples the mining and fusion of view features via a bi-level optimization process.

Ⅱ. Method

➢ Overall Architecture

1 data processing + **2** optimization levels + **3** main modules

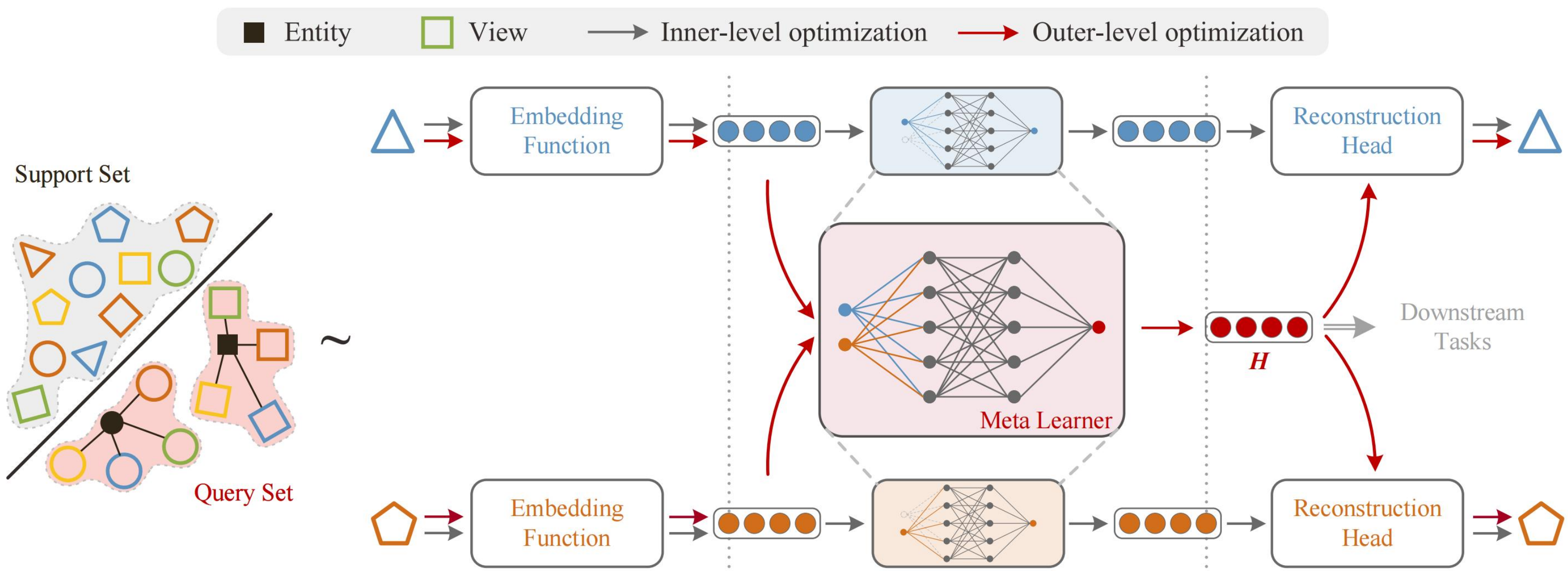


(a) Meta Split (b) Embedding Module (c) Representation Learning Module (d) Self-Supervised Module

Ⅱ. Method

➢ Overall Architecture

1 data processing + **2** optimization levels + **3** main modules



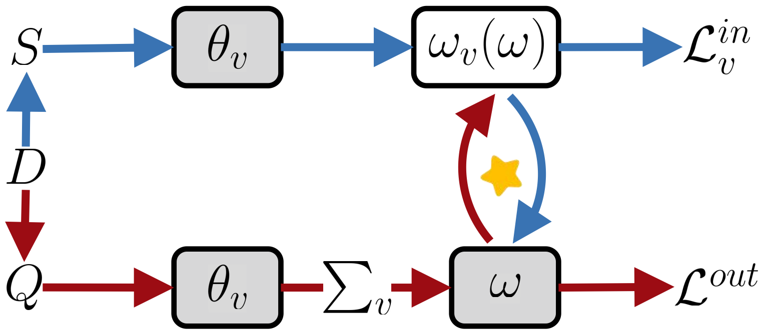
(a) Meta Split (b) Embedding Module (c) Representation Learning Module (d) Self-Supervised Module

θ_v denotes parameters specific to view v in (b) and (d) modules, ω and ω_v are parameters of the meta-learner and base learner in (c) module.

Ⅱ. Method

- Data flow

θ_v denotes parameters specific to view v in (b) and (d) modules, ω and ω_v are parameters of the meta-learner and base learner in (c) module.



Algorithm 1 Main flows in MetaViewer.

```
Initialize ω, {θ_v}_{v=1}^V;
while not done do
    # Outer-level
    Sample and meta-split a batch set from D:
    {D_v^batch}_{v=1}^V = {S_v}_{v=1}^V + {Q_v}_{v=1}^V.
    while not large than the inner-level step do
        # Inner-level
        for v = 1, ..., V do
            Initialize ω_v = ω;
            Optimize ω_v(ω) via inner-level objective.
        end for
    end while
    Optimize ω and {θ_v}_{v=1}^V via outer-level objective.
end while
```

Inner-level initializes ω_v from ω and updates it on Q to learn view-specific knowledge. Based on this, **Outer-level** computes the loss on S and updates ω to learn the uniform representation.

Ⅱ. Method

- Optimization objective

$$\omega^*, \{\theta_v^*\}_{v=1}^{V} = \arg\min \mathcal{L}^{out}\left(\{\omega_v^*(\omega), \theta_v\}_{v=1}^{V}; Q\right)$$ Outer-level

$$s.t., \quad \omega_v^*(\omega) = \arg\min \mathcal{L}_v^{in}\left(\omega_v(\omega), \theta_v; S_v\right)$$ Inner-level

- The instance of loss function

$$\mathcal{L}_v^{in} = \mathcal{L}_v^{rec}(S_v, S_v^{rec}) = \|S_v - S_v^{rec}\|_F^2$$

\+

$$\mathcal{L}^{out} = \sum_v \mathcal{L}_v^{rec}(Q_v, Q_v^{rec})$$

\|\|

MVer-R

a pure implementation of MetaViewer

\+

$$\mathcal{L}^{out} = \sum_v \left(\mathcal{L}_v^{rec} + \sum_{v',v'\neq v} \mathcal{L}_{v,v'}^{con}\right)$$

\|\|

MVer-C

a variant of MetaViewer with a contrastive loss

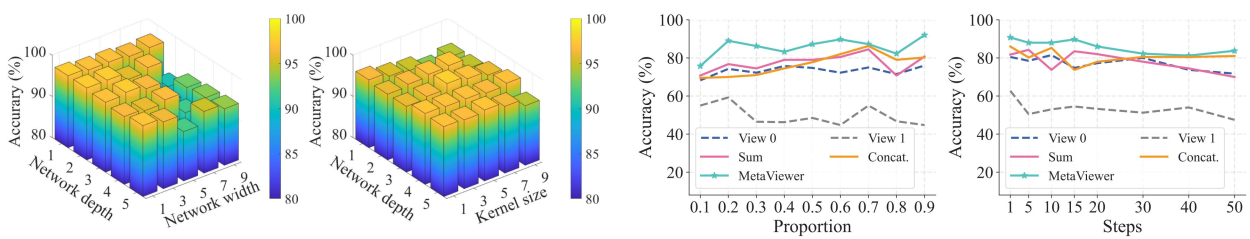
Ⅲ. Result

- Results on classification tasks (top) and clustering tasks (bottom) constructed from five datasets. Bold and underline denote the best and second-best results, respectively.

Methods	Handwritten ACC	Handwritten Prec.	Handwritten F-score	RGB-D ACC	RGB-D Prec.	RGB-D F-score	Animal ACC	Animal Prec.	Animal F-score	Fashion-MV ACC	Fashion-MV Prec.	Fashion-MV F-score	Caltech101-20 ACC	Caltech101-20 Prec.	Caltech101-20 F-score
Baseline	87.00	87.32	87.04	14.00	06.45	07.70	71.78	65.02	63.10	87.50	87.63	87.54	75.41	51.95	46.21
DCCA [2] (2013)	88.25	89.20	88.05	30.00	21.10	22.04	62.84	61.61	60.65	84.90	85.22	83.54	71.54	45.27	39.81
DCCAE [47] (2015)	90.00	90.48	89.92	24.00	16.00	16.91	66.82	62.53	62.83	85.35	85.97	83.84	71.54	60.57	43.25
MIB [8] (2020)	79.00	83.90	78.52	33.00	28.50	27.37	59.32	63.78	62.13	86.80	86.80	86.55	72.72	61.64	52.47
WTNNM [15] (2020)	96.77	96.20	96.36	45.00	47.18	43.90	69.90	67.40	65.48	94.50	94.50	94.58	83.27	80.49	74.78
TLRR [23] (2021)	97.00	97.05	97.17	49.00	53.32	47.66	71.90	69.52	68.35	96.35	96.28	96.40	85.55	77.30	75.24
MFLVC [51] (2022)	94.00	94.20	94.01	44.00	46.09	41.81	75.62	72.17	70.81	96.50	96.52	96.49	85.37	71.83	69.07
DCP [27] (2022)	97.25	97.30	97.24	37.00	28.87	30.78	77.95	73.43	70.01	89.25	82.06	82.90	**92.48**	89.41	84.58
MVer-R (ours)	97.00	97.08	97.00	51.00	53.65	48.73	77.69	73.91	71.22	96.85	96.37	96.48	92.28	89.46	84.21
MVer-C (ours)	**97.75**	**97.90**	**97.75**	**56.00**	**55.20**	**52.78**	**78.03**	**74.56**	**71.55**	**97.70**	**96.78**	**97.07**	92.16	**90.68**	**85.72**

Methods	Handwritten ACC	Handwritten NMI	Handwritten ARI	RGB-D ACC	RGB-D NMI	RGB-D ARI	Animal ACC	Animal NMI	Animal ARI	Fashion-MV ACC	Fashion-MV NMI	Fashion-MV ARI	Caltech101-20 ACC	Caltech101-20 NMI	Caltech101-20 ARI
Baseline	43.55	52.19	25.68	48.30	80.97	04.95	45.79	66.22	33.87	50.91	45.98	32.09	36.63	48.96	23.54
DCCA [2] (2013)	57.25	69.80	52.15	51.00	82.99	52.02	60.34	68.51	46.10	70.70	80.42	61.80	38.62	50.88	22.73
DCCAE [47] (2015)	63.00	75.04	59.29	48.00	81.58	48.34	65.17	66.82	48.83	71.05	81.12	62.34	36.59	52.24	25.25
MIB [8] (2020)	63.25	67.58	52.16	50.00	81.13	51.27	60.89	61.98	52.49	57.20	73.83	47.62	35.98	47.00	22.18
WTNNM [15] (2020)	75.69	77.62	61.51	52.54	81.95	52.46	71.92	71.43	61.06	83.33	86.30	77.99	39.74	55.58	26.92
TLRR [23] (2021)	78.60	77.99	66.40	52.87	82.38	53.37	69.79	73.01	61.23	79.54	84.01	78.54	40.18	52.38	27.60
MFLVC [51] (2022)	64.00	64.53	48.85	53.00	83.31	54.07	74.10	76.08	64.28	83.20	88.75	78.93	36.59	58.36	26.87
DCP [27] (2022)	66.25	70.56	56.10	52.00	82.04	52.64	72.77	74.02	65.48	62.60	68.38	54.30	36.79	44.37	23.50
MVer-R (ours)	75.00	78.53	67.21	53.00	82.41	53.04	**76.49**	**78.25**	65.28	80.80	88.13	75.05	41.87	58.52	29.19
MVer-C (ours)	**86.25**	**78.96**	**72.25**	**57.00**	**84.97**	**57.07**	75.92	78.01	**66.07**	**85.40**	**88.76**	**80.07**	**45.12**	**60.86**	**35.00**

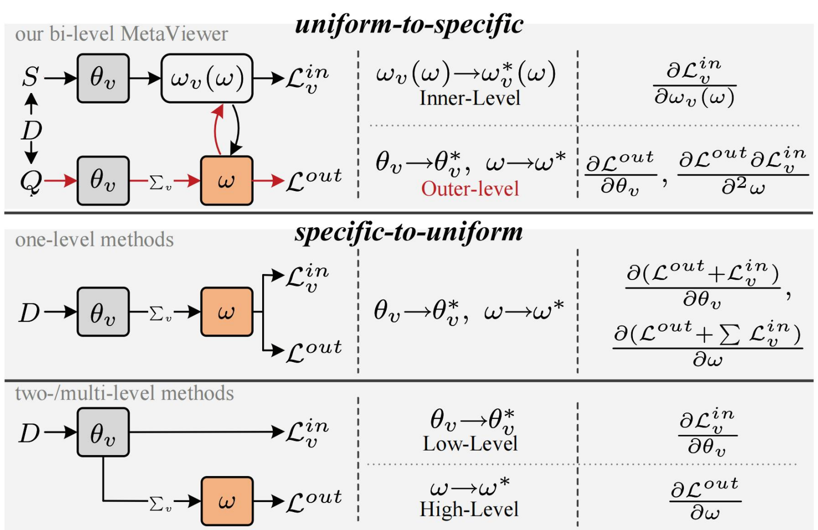
- Sensitivity studies on key hyperparameters



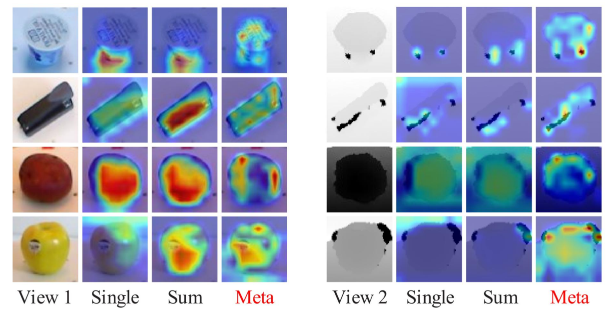
(a) Network width (b) Kernel size (c) Support proportion (d) Inner-level step

Ⅳ. Disscuss

➢ We analyze proposed MetaViewer in depth from two aspects: the pipeline (left) and the fusion strategy (right).



(a) Flowcharts (b) Steps (c) Gradients



Strategies	Rules	ACC↑	NMI↑	ARI↑	MSE↓
Sum	$z^x + z^y$	69.25	71.89	59.02	1.84
Max	$max(z^x, z^y)$	80.75	73.93	63.75	-
Concat.	$cat[z^x, z^y]$	78.75	72.02	61.52	**1.77**
Linear	$l(z^x, z^y, \theta_l)$	85.00	77.40	69.71	4.74
C-Conv	$m(z^x, z^y, \omega)$	69.75	65.21	51.33	2.37
MetaViewer	*meta-learning*	**86.25**	**78.96**	**72.25**	2.45

Left: *S2U* vs. *U2S*

U2S update parameters ω by observing "the learning process from a unified representation to view-specific features".

Right: manual design vs. meta-learning

MetaViewer balances information from all views instead of just the salient one (i.e., View 1).

Thanks

Project Page:
https://xxlifelover.github.io/MetaViewerProjectPage/

Towards **I**ntelligence **ME**chanism (TIME) Lab:
http://time.sdu.edu.cn/