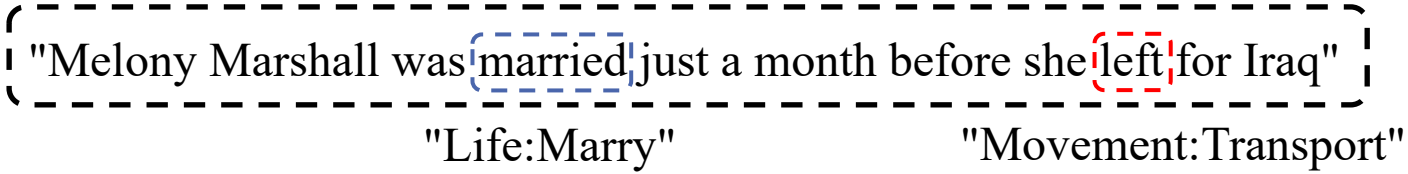
Continual Few-shot Event Detection via Hierarchical Augmentation Networks

Chenlong Zhang[1,2]*, **Pengfei Cao**[1,2]*, **Yubo Chen**[1,2]†, **Kang Liu**[1,2,3]
Zhiqiang Zhang[4], **Mengshu Sun**[4], **Jun Zhao**[1,2]

[1]The Laboratory of Cognition and Decision Intelligence for Complex Systems,
Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[3]Shanghai Artificial Intelligence Laboratory, Shanghai, China
[4]Ant Group, Hangzhou, China
zhangchenlong2023@ia.ac.cn
{pengfei.cao, yubo.chen, kliu, jzhao}@nlpr.ia.ac.cn

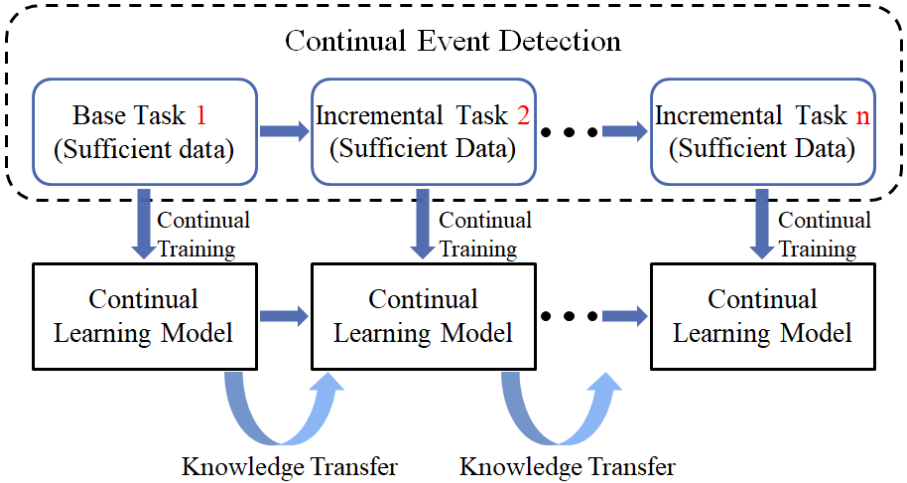
Background

- **Event Detection**
  - Detect event triggers and classify the corresponding event types.
  - Assume fixed data and pre-defined event types.
  - In real-world applications, new events emerge continually.



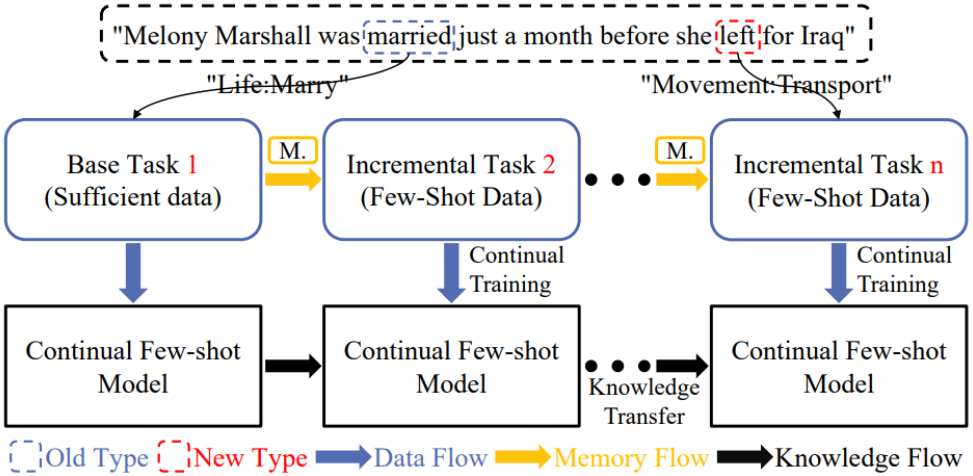
Background

- **Continual Event Detection**
  - Assume multiple detection tasks emerge continually and the training samples for each tasks are sufficient.
  - Obtaining high-quality samples for new tasks are expensive.
  - Training model in sequential way leads to catastrophic forgetting.



Background

- **Continual Few-shot Event Detection**
  - Only 5 or 10 training samples are available in incremental tasks.
  - Memorize previous tasks and learn new tasks with few-shot samples.
  - More challenging and realistic scenario.

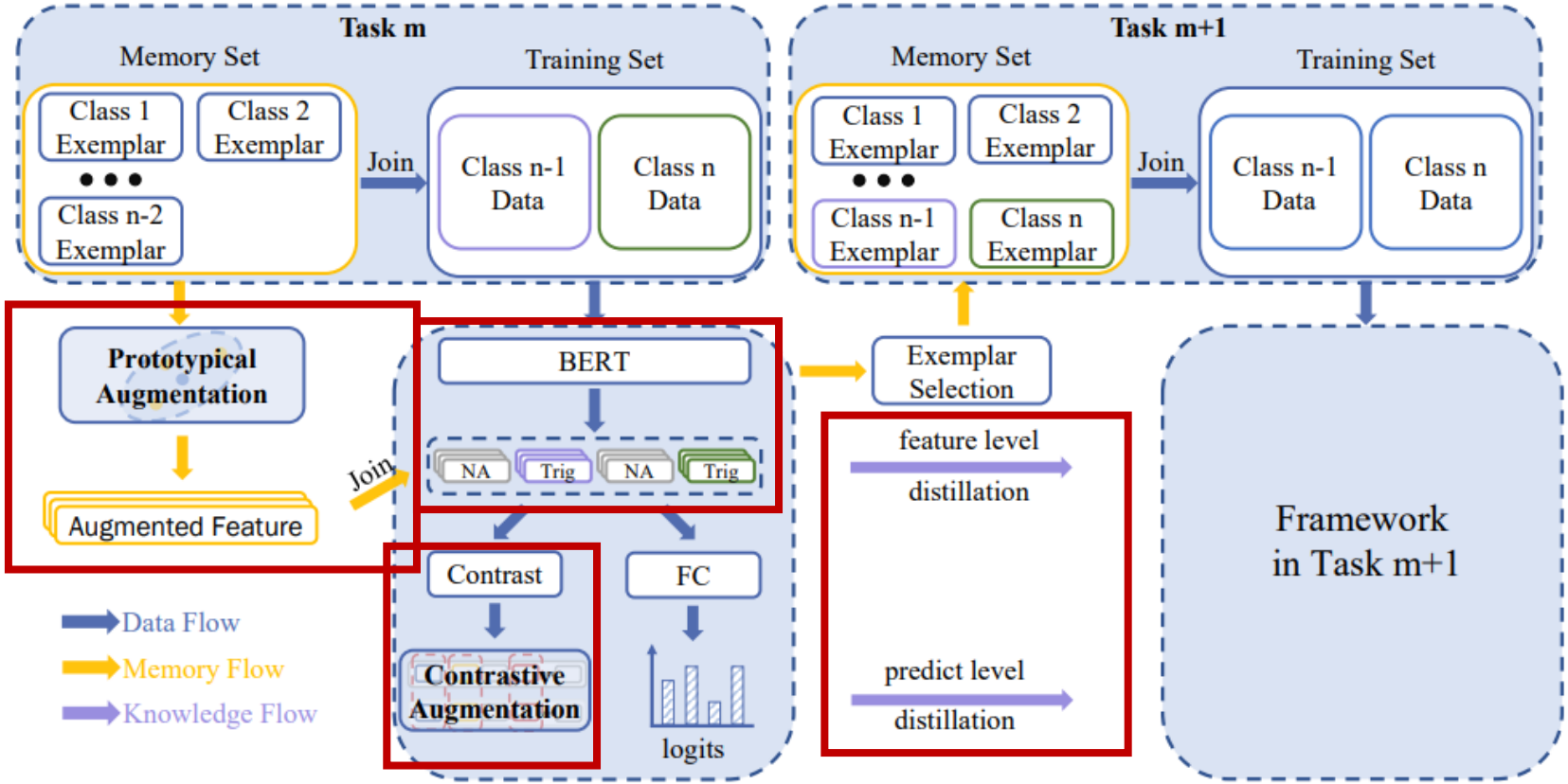


Challenges

- **Challenge 1: exemplar collapse in memorizing previous tasks**
  - Existing CED methods mitigate catastrophic forgetting by restoring prototypical exemplars.
  - In CFED task, the prototypical exemplars are no longer representative as only one sample can be restored.
  - Use limited exemplar to effectively characterize the prototypical feature space.
- **Challenge 2: overfitting in learning new tasks**
  - In CED settings, model learn from abundant samples in new tasks.
  - However, when trained with limited samples in CFED tasks, they struggle to generalize well and suffer from overfitting.
  - Find a solution to fully utilize valuable knowledge from few-shot samples.

Methodology

HANet: Hierarchical Augmentation Networks



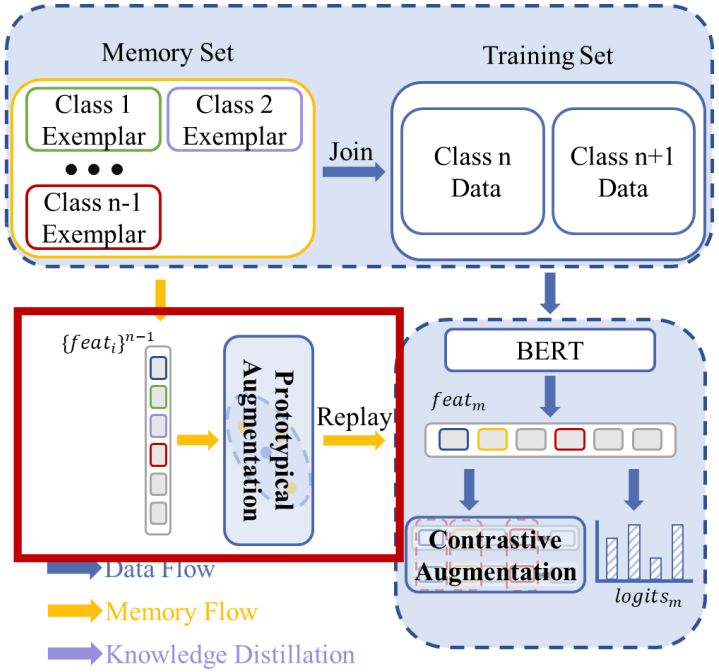
Hierarchical Augmentation Networks

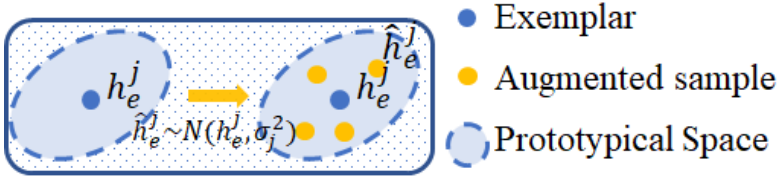
- **Prototypical Augmentation**
  - Solve exemplars collapse when memorizing previous tasks.
  - Reconstruct prototypical feature space via augmenting the exemplars from gaussian distribution estimated from previous tasks.

$$\hat{\mathbf{H}}_e^j = \{\hat{\mathbf{h}}_{e,1}^j, \ldots, \hat{\mathbf{h}}_{e,n}^j\} \sim \mathcal{N}(\mu_j, \sigma_j^2)$$

  - Replay augmented exemplars via cross-entropy.

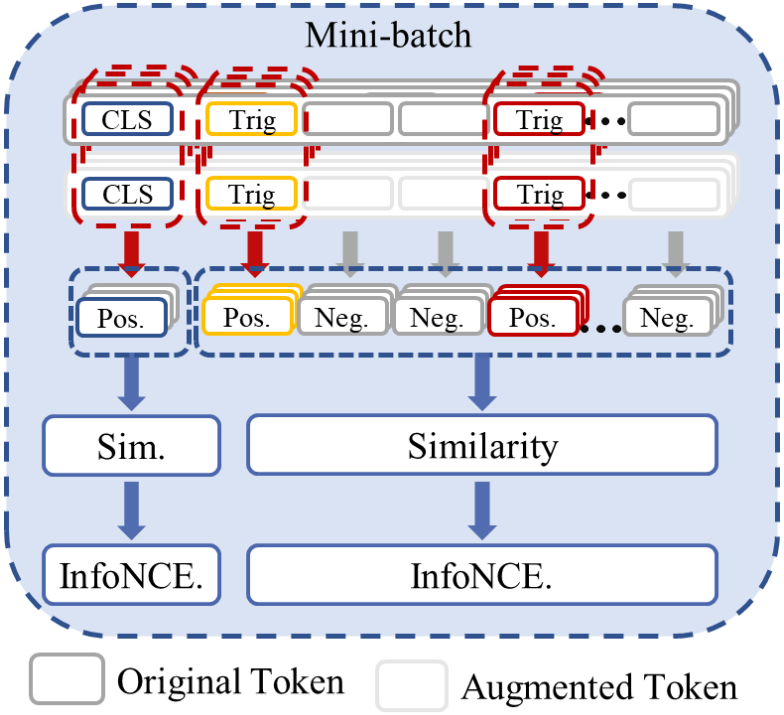
$$\mathcal{L}_{re} = -\sum^{\hat{\mathbf{H}}_e^j} \mathbf{y}_j \log \hat{\mathbf{p}}_j$$





Hierarchical Augmentation Networks

- **Contrastive Augmentation**
  - Avoid overfitting when learning new tasks with few-shot samples.
  - Uncover the implicit inter-information in the token scale.
  - Construct Contrastive pairs according to the triggers' labels.
  - Apply InfoNCE loss for both [CLS] token and [Trigger] token to fully exploit the information.



$$\mathcal{L}_{cls} = \frac{1}{n-1}\sum_{i}^{|\mathcal{B}|} -\frac{1}{m}\sum_{i\neq k}^{|\mathcal{O}|} \log \frac{\exp(S(\mathbf{h}_{cls_i}^{j}, \mathbf{h}_{cls_i}^{k})/\tau)}{\sum_{p\neq i}^{|\mathcal{B}|}\sum_{q}^{|O|}\exp(S(\mathbf{h}_{cls_i}^{j}\mathbf{h}_{cls_p}^{q})/\tau)}$$

$$\mathcal{L}_{trig} = \frac{1}{n-1}\sum_{i\neq l}^{|\mathcal{B}|} -\frac{1}{m}\sum_{j\neq k}^{|\mathcal{O}|}[\mathbf{y}_i^j = \mathbf{y}_l^k] \log \frac{\exp(S(\mathbf{h}_{e_i}^{j}, \mathbf{h}_{e_l}^{\kappa})/\tau)}{\sum_{p\neq i}^{|\mathcal{B}|}\sum_{q}^{|\mathcal{O}|}[\mathbf{y}_i^j \neq \mathbf{y}_p^q]\exp(S(\mathbf{h}_{e_i}^{j}, \mathbf{h}_{e_p}^{q})/\tau)}$$

Experiments

- **4-way MAVEN Dataset and 2-way ACE Dataset**
  - Our method outperforms previous methods by a large margin.

Method	4-way 5-shot						4-way 10-shot					
	1	2	3	4	5	$\bar{F1}_{micro}$	1	2	3	4	5	$\bar{F1}_{micro}$
Fine-tune	40.43±2.34	33.17±3.55	17.5±2.07	19.72±0.92	21.01±0.87	26.36±1.3	40.43±2.34	38.18±2.83	20.46±1.11	20.35±2.19	23.57±1.01	28.6±0.92
Retrain	40.43±2.34	42.1±1.13	39.61±1.12	43.03±1.56	**47.43±0.67**	42.52±0.7	40.43±2.34	44.27±1.36	44.76±1.37	**48.28±1.43**	**53.66±0.97**	46.28±0.95
EWC	40.43±2.34	34.29±1.41	17.4±1.5	18.61±2.52	20.43±1.67	26.23±1.39	40.43±2.34	36.42±3.34	19.69±0.93	20.02±1.14	23.72±1.19	28.06±1.01
LwF	40.43±2.34	37.27±4.9	26.69±4.07	24.7±1.47	30.54±1.43	31.93±2.05	40.43±2.34	41.09±2.8	31.89±0.57	30.57±1.09	34.43±2.08	35.68±0.69
ICaRL	35.82±4.76	37.16±4.85	33.74±2.85	35.54±2.37	35.98±2.48	35.65±2.93	35.82±4.76	42.43±4.48	37.45±1.58	40.11±0.9	41.04±1.17	39.37±2.05
KCN	40.43±2.35	48.38±1.66	41.99±2.01	41.32±1.53	40.29±1.51	42.48±1.49	40.43±2.35	51.15±1.19	45.22±1.22	44.31±0.69	44.47±1.51	45.12±1.09
KT	41.04±1.59	40.19±2.17	35.21±1.34	32.69±0.78	33.77±0.58	36.58±1.06	41.04±1.59	44.39±0.91	40±1.3	39.42±0.33	37.87±0.95	40.54±0.58
EMP	40.17±1.34	30.95±0.75	31.21±1.32	22.9±2.09	22.25±1.43	29.5±0.76	40.17±1.34	32.33±0.69	32.95±1.11	26.68±1.5	28.16±1.89	32.06±0.8
HANet(Ours)	**41.91±3.76**	**51.39±1.55**	**43.21±3.19**	**43.53±4.21**	43.89±5.65	**44.79±2.33**	**41.91±3.76**	**53.17±1.27**	**46.71±2.51**	46.36±3.64	48.12±5.49	**47.25±2.23**

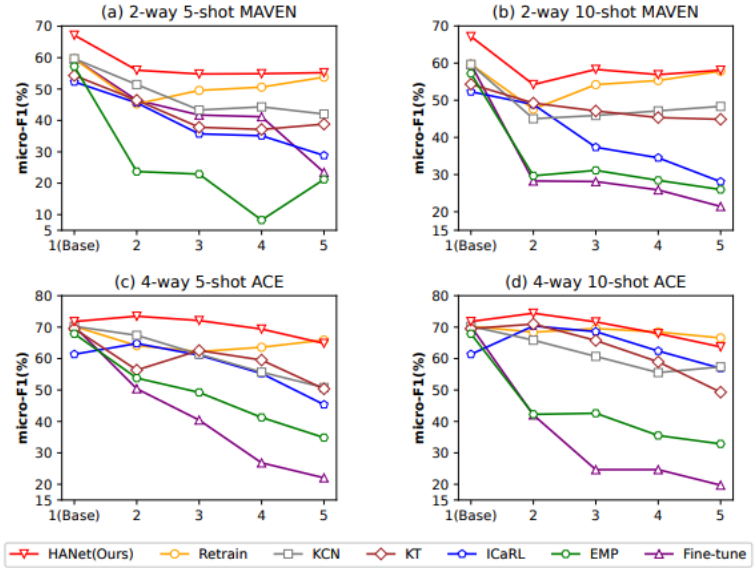
Table 1: $F1_{micro}$ of every sub-task and $\bar{F1}_{micro}$ across all sub-tasks on 4-way MAVEN benchmark.

Method	2-way 5-shot						2-way 10-shot					
	1	2	3	4	5	$\bar{F1}_{micro}$	1	2	3	4	5	$\bar{F1}_{micro}$
Fine-tune	60.86±2.96	52.09±9.59	46.37±10	26.64±6.98	23.15±4.66	41.82±3.56	60.86±2.96	48.17±9.8	49.55±2.91	23.29±8.2	24.66±3.23	41.31±3.31
Retrain	60.86±2.96	62.45±4.27	52.21±7.83	52.2±4.68	**58.36±6.09**	57.22±4.48	60.86±2.96	63.39±2.87	63.75±2.67	**61.23±2.08**	**64.25±3.13**	**62.7±1.3**
EWC	60.86±2.96	49.3±8.93	45.41±10.43	27.14±11.24	22.36±3.9	41.02±4.85	60.86±2.96	47.58±10.11	51.15±3.05	23.82±7.67	21.79±3.1	41.04±2.78
LwF	60.86±2.96	47.31±10.4	38.91±12.89	23.31±13.46	28.4±2.83	39.76±6.85	60.86±2.96	46.98±8.32	50.77±3.35	33.48±2.7	29.69±2.91	44.36±2.2
ICaRL	50.85±6.51	52.21±2.72	37.39±6.78	31.33±6.31	28.85±5.04	40.13±4.1	50.85±6.51	52.06±2.66	42.45±6.48	32.89±4.96	34.7±3.93	42.59±2.8
KCN	60.86±2.96	56.38±5.03	47.56±10.41	38.62±9.47	37.05±7.11	48.09±6.41	60.86±2.96	59.41±6.74	57.39±6.19	46.48±6.1	44.3±5.43	53.69±4.42
KT	53.16±2.25	42.55±2.33	33.93±2.97	38.48±8.66	31.27±9.34	39.88±3.84	53.16±2.25	59.12±1.78	50.02±5.13	49.02±5.34	28.54±2.95	47.97±2.67
EMP	54.78±1.49	40.49±1.9	24.32±3.37	27.15±8.46	22.53±6.02	33.85±2.96	54.78±1.49	37.28±7.37	19.6±4.96	34.69±4.76	24.19±6.62	34.11±3.48
HANet(Ours)	**61.16±2.29**	**63.07±3.09**	**57.5±5.98**	**53.21±4.64**	54.31±3.21	**57.85±2.91**	**61.16±2.29**	**66.84±2.88**	**64.68±3.77**	58.02±6.58	54.37±5.94	61.02±3.46

Table 2: $F1_{micro}$ of every sub-task and $\bar{F1}_{micro}$ across all sub-tasks on 2-way ACE benchmark.

Experiments

- **2-way MAVEN Dataset and 4-way ACE Dataset**
  - Compared with previous methods, our approach significantly outperforms them across all sub-tasks.



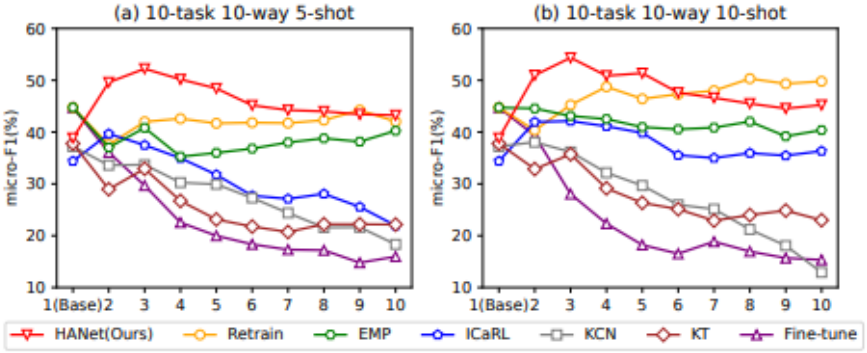
Experiments

- **Ablations**
  - Effectiveness of Prototypical Augmentation.
  - Effectiveness of Contrastive Augmentation.
  - Synergetic effect of Hierarchical Augmentation.

Method	2-way 5-shot						2-way 10-shot					
	1	2	3	4	5	$\bar{F}1_{micro}$	1	2	3	4	5	$\bar{F}1_{micro}$
HANet(Ours)	**67.16**	**56.01**	**54.80**	**54.89**	**55.22**	**57.62**	**67.16**	**54.22**	**58.31**	**56.90**	**58.09**	**58.94**
w/o Replay*	**67.16**	51.02	44.15	38.76	36.78	47.57	**67.16**	48.13	48.14	41.07	40.01	48.90
w/o Distill	**67.16**	46.83	42.77	37.17	42.90	47.37	**67.16**	45.45	44.07	44.90	47.77	49.87
w/o PA	**67.16**	54.28	53.01	50.98	52.21	55.53	**67.16**	52.94	57.47	53.91	55.38	57.37
w/o CA	59.67	54.45	49.14	50.08	49.57	52.58	59.67	53.31	53.75	53.16	53.46	54.67
w/o PA and CA	59.67	51.43	43.32	44.32	42.04	48.16	59.67	45.03	45.90	47.14	48.35	49.22

Experiments

- **Evaluation in Extreme Scenarios**
  - Effectiveness of HANet with more Incremental tasks.
  - Effectiveness of HANet with 1-shot and 2- shot tasks.



Method	2-way 1-shot						2-way 2-shot					
	1	2	3	4	5	$\bar{F}1_{micro}$	1	2	3	4	5	$\bar{F}1_{micro}$
Fine-tune	59.67	26.81	28.34	22.96	18.79	31.31	59.67	56.17	41.67	33.13	22.81	42.69
Retrain	59.67	42.34	33.33	29.04	28.25	38.53	59.67	44.68	37.73	38.70	40.98	44.35
EWC	59.67	35.95	28.22	15.79	16.17	31.16	59.67	55.68	47.96	36.10	26.92	45.27
LwF	59.67	5.28	24.63	27.11	30.82	29.50	59.67	36.72	34.07	28.94	28.71	37.62
ICaRL	52.29	36.71	34.18	31.06	25.77	36.00	52.29	41.38	34.44	33.47	29.19	38.15
KCN	59.67	39.10	**43.19**	41.97	38.18	44.42	59.67	54.40	**50.67**	49.98	47.58	52.46
KT	54.32	5.94	5.78	3.70	3.61	14.67	54.32	35.22	32.71	27.47	28.23	35.59
EMP	57.21	4.95	5.53	5.42	5.29	15.68	57.21	18.28	6.84	7.06	8.43	19.56
HANet(Ours)	**67.16**	**45.54**	38.28	**42.39**	**40.40**	**46.75**	**67.16**	**55.87**	50.35	**51.63**	**51.39**	**55.28**

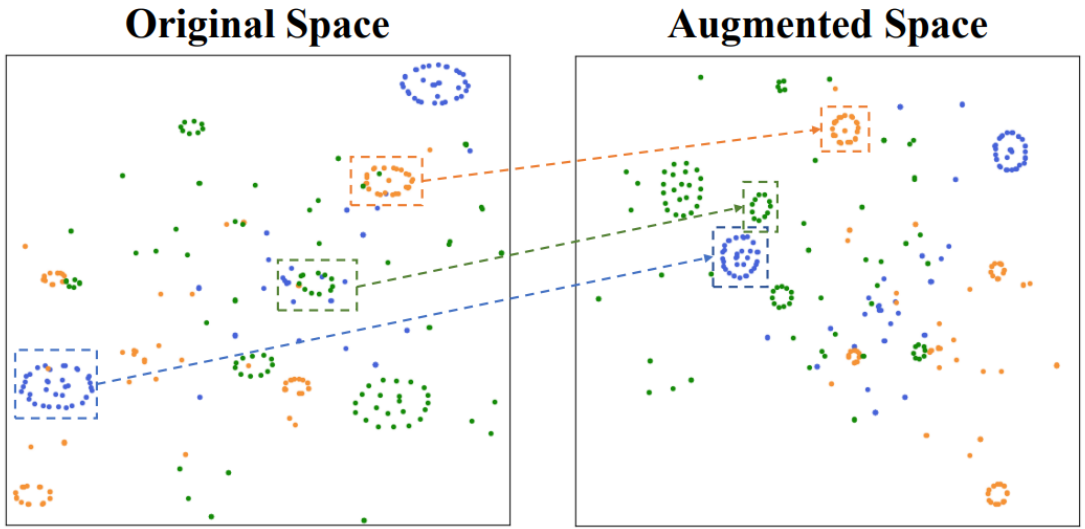
Experiments

- **Comparison with Large language models**
  - Experiments with gpt-3.5-turbo prove that our method outperforms ChatGPT significantly.

Benchmark	Method	2-way 1-shot						2-way 2-shot					
		1	2	3	4	5	$\bar{F}1_{micro}$	1	2	3	4	5	$\bar{F}1_{micro}$
MAVEN	HANet(Ours)	**67.16**	45.54	38.28	**42.39**	**40.40**	**46.75**	**67.16**	55.87	**50.35**	**51.63**	**51.39**	**55.28**
	gpt-3.5-turbo	54.22	**55.25**	**41.60**	37.88	33.31	44.45	57.00	**58.51**	43.64	40.39	36.56	47.22
ACE	HANet(Ours)	**60.99**	**51.93**	**41.67**	41.54	**35.84**	**46.40**	**60.99**	**58.38**	39.48	41.76	**44.60**	**49.04**
	gpt-3.5-turbo	42.20	50.29	40.51	**43.46**	35.21	42.33	56.36	49.72	**45.16**	**44.44**	42.96	47.73

Experiments

- **Visualization**
  - Effectiveness of Prototypical Augmentation.
  - the intra-class distances become closer for each type.



Conclusions

- We are the first to propose continual few-shot event detection, a more realistic yet challenging task.
- We devise Hierarchical Augmentation Networks (HANet)
- Experiments with existing methods prove the effectiveness of our methods.

Thanks for watching!