Spear and Shield: Adversarial Attacks and Defense Methods for Model-based Link Prediction on Continuous-Time Dynamic Graphs

Dongjin Lee[1], Juho Lee[2], Kijung Shin[1,2]

[1]School of Electrical Engineering, [2]Kim Jaechul Graduate School of AI, KAIST, South Korea

{dongjin.lee, juholee, kijungs}@kaist.ac.kr

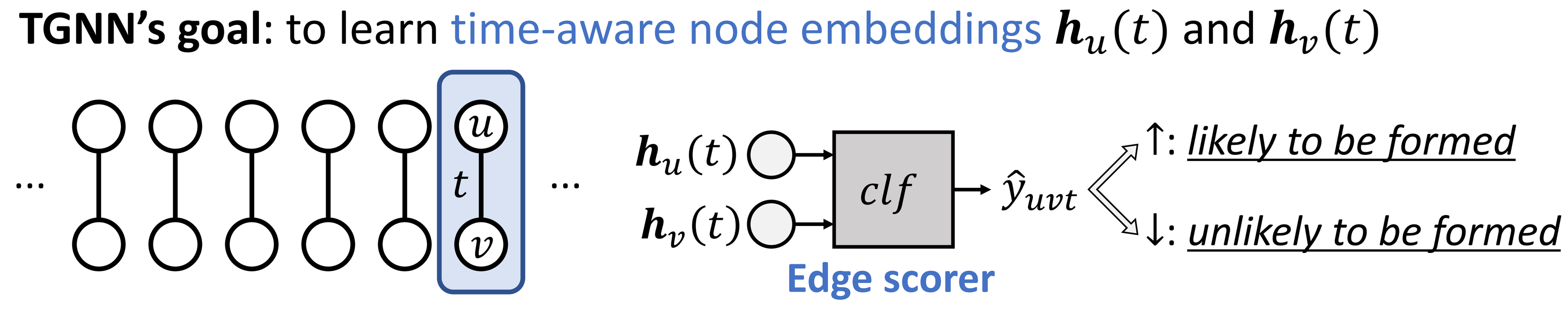
Code and Data: https://github.com/wooner49/T-spear-shield

Summary

- **Goal:**
  1. To explore the weakness of temporal graph neural networks (TGNNs) by proposing an adversarial attack for link prediction on continuous-time dynamic graphs (CTDGs)
  2. To enhance the robustness of TGNNs against perturbations and alleviate performance degradation
- **Proposed Algorithms:**
  - **T-SPEAR:** a gray-box and poisoning attack for link prediction on CTDGs
  - **T-SHIELD:** a robust training method for TGNNs
- **Contributions**:
  - We formulate adversarial attacks on CTDGs under 4 realistic constraints regarding unnoticeability.
  - Our attack can significantly degrade TGNNs' performances in link prediction.
  - Our defense can accurately classify the adversarial edges and mitigate performance degradation.

TGNNs for Link Prediction

- TGNNs are commonly trained using temporal interactions as supervision.
- **Given**: two nodes u and v at time t
- **TGNN's goal**: to learn time-aware node embeddings $\boldsymbol{h}_u(t)$ and $\boldsymbol{h}_v(t)$

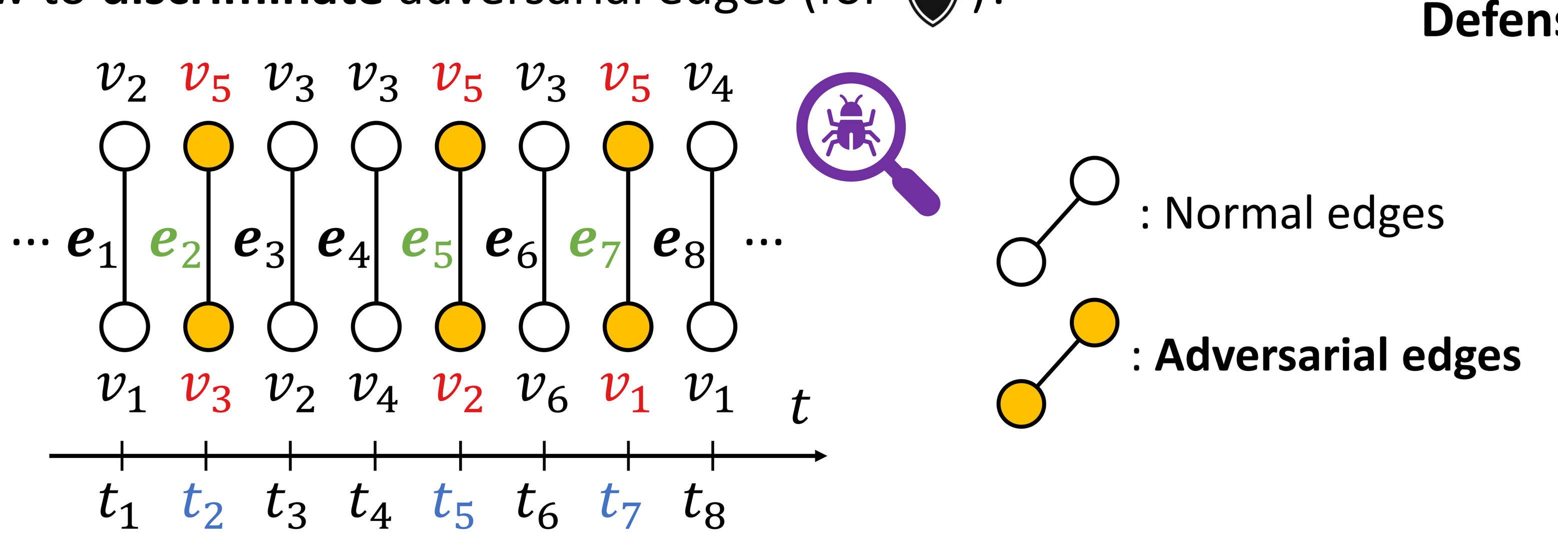


Challenges

C1) **What** to connect (for Attack)?
C2) **When** to connect (for Attack)?
C3) How to apply a **stealthy attack** (for Attack)?
C4) How to **discriminate** adversarial edges (for Defense)?

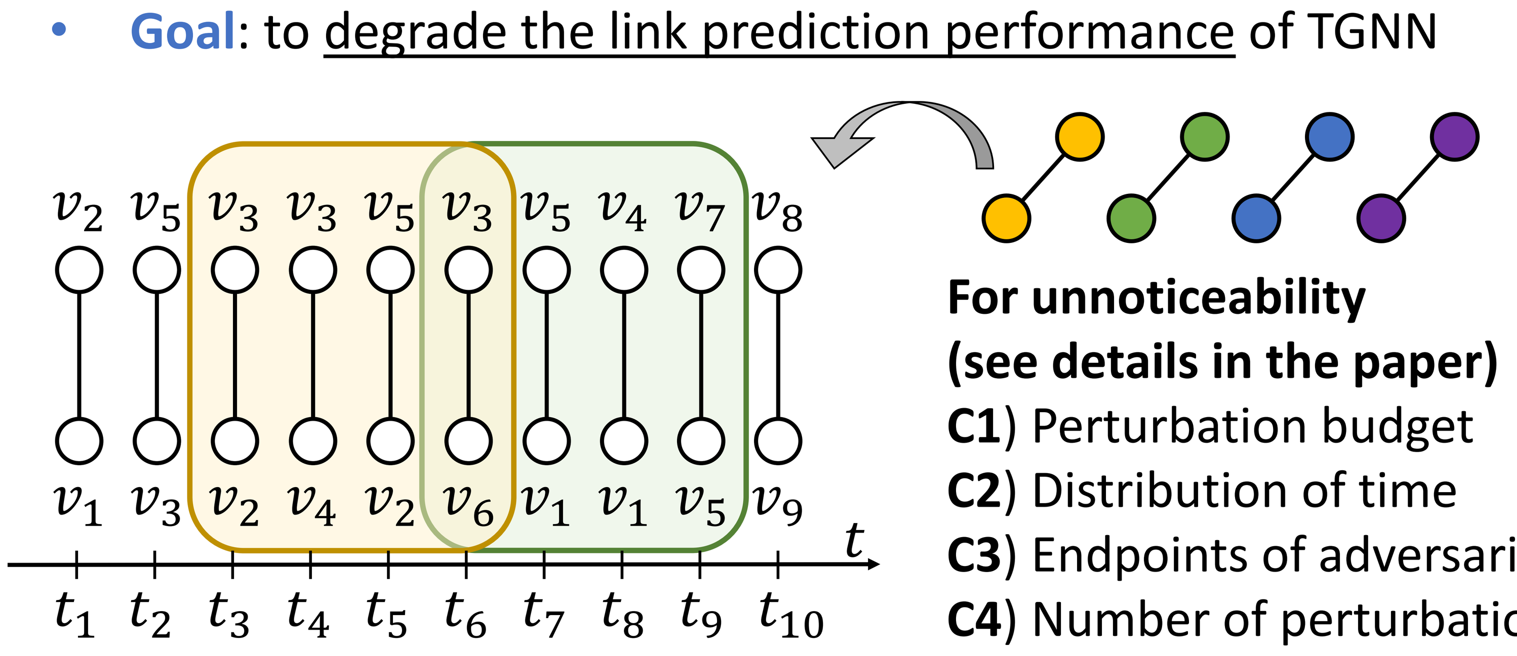
Attack

Defense



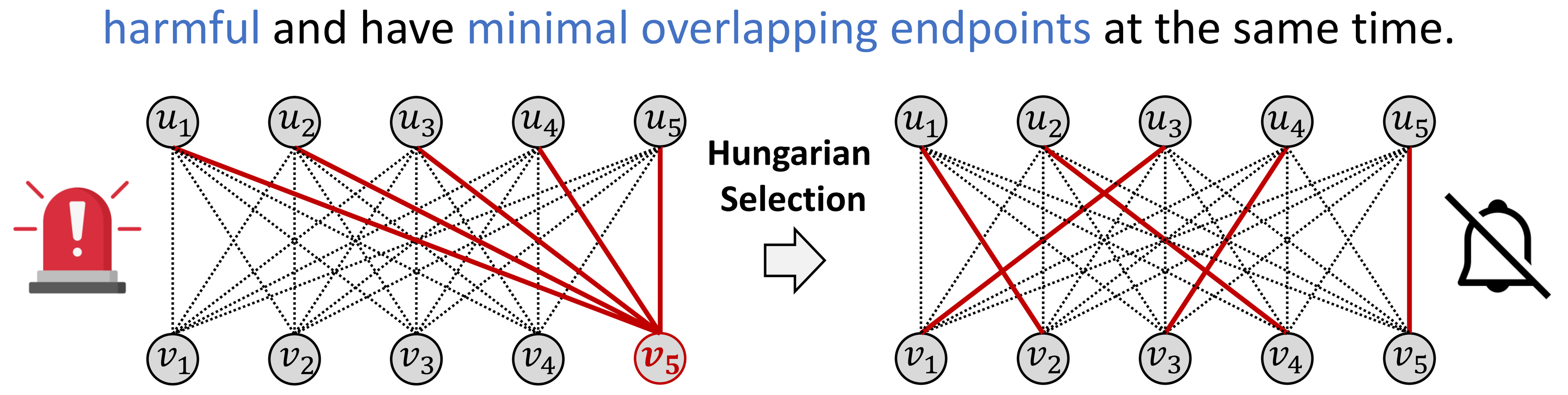
Proposed Attack: T-SPEAR

- We proposed an effective adversarial attack method, called *T-SPEAR*.
  - **Constraints**: adhering to the four constraints (C1-C4) for unnoticeability
  - **How**: by injecting adversarial edges into the original edge sequence
  - **Goal**: to degrade the link prediction performance of TGNN



For unnoticeability
(see details in the paper)
C1) Perturbation budget
C2) Distribution of time
C3) Endpoints of adversarial edges
C4) Number of perturbations per node

- **Step 1**: Train the surrogate model (TGN)
- **Step 2**: Sample the timestamp where the adversarial edges are inserted
- **Step 3**: Choose the endpoints which compose the adversarial edges
  - We propose **Hungarian selection** to select adversarial edges that are harmful and have minimal overlapping endpoints at the same time.



more inconspicuous and balanced perturbations

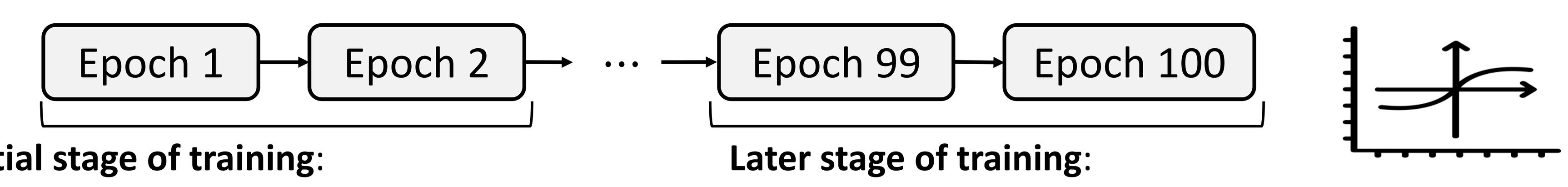
Proposed Defense: T-SHIELD

- We proposed a robust training method for TGNNs, called *T-SHIELD*.
  - **Knowledge**:
    - No information about the attacker
    - Not even know that the dynamic graph has been corrupted.
  - **Goal**: to alleviate performance degradation caused by adversarial attacks

1. **Edge filtering**: To identify and eliminate potential adversarial edges

$\boldsymbol{h}_u(t)$, $\boldsymbol{h}_v(t)$ → clf → $\hat{y}_{uvt}$
- if $\hat{y}_{uvt} \geq \tau$, then (u, v, t) is a *legitimate edge*
- if $\hat{y}_{uvt} < \tau$, then (u, v, t) is a *potential adversarial edge* ($\hat{e}$)

❖ Two issues arise when filtering with fixed threshold τ.
❖ We use cosine annealing scheduler to gradually increase the threshold from τ_s to τ_e.



Initial stage of training:
High τ → remove too many legitimate edges

Later stage of training:
Low τ → fail to filter out adversarial edges

2. **Temporal smoothness**: To prevent sudden changes of node embeddings

$$\mathcal{L}_{tmp} = -\sum_{(u,v,t)\in\mathcal{E}_c} \sum_{i\in\{u,v\}} w(t - t_i^-)\mathrm{sim}(\boldsymbol{h}_i(t), \boldsymbol{h}_i(t_i^-)),$$

❖ $\mathcal{E}_c$: filtered dynamic graph
❖ $\mathrm{sim}(\cdot,\cdot)$: cosine similarity
❖ t_i^-: the last timestamp before t of node i
❖ $w(\Delta t) = \exp(-\theta \Delta t)$

Experiments

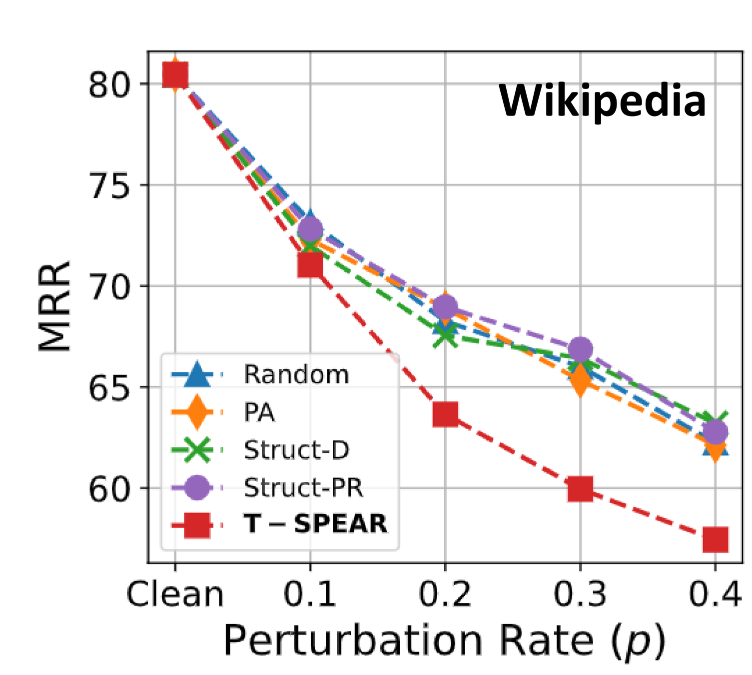
- **Attack Performance**

Metric: Mean Reciprocal Rank (MRR)
 # negatives: 100
Perturbation rate (p): 0.3

Dataset	Victim	Clean	Random	PA	Jaccard*	Struct-D	Struct-PR	T-Spear
Wikipedia	TGN	80.5 ± 0.5	66.0 ± 0.6	65.3 ± 1.1	-	66.4 ± 0.5	66.9 ± 0.9	**60.0 ± 1.6**
	JODIE	63.2 ± 1.2	36.3 ± 3.4	35.0 ± 2.3	-	**33.0 ± 2.1**	35.0 ± 2.5	33.8 ± 2.4
	TGAT	57.0 ± 0.6	41.4 ± 0.6	**39.2 ± 1.0**	-	41.1 ± 0.6	40.8 ± 0.5	40.7 ± 0.5
	DySAT	66.6 ± 0.4	63.0 ± 0.5	62.4 ± 0.5	-	62.1 ± 0.2	62.8 ± 0.4	**62.0 ± 0.4**
MOOC	TGN	61.8 ± 2.1	55.4 ± 1.7	56.0 ± 1.6	-	50.3 ± 1.4	53.2 ± 1.2	**48.2 ± 1.3**
	JODIE	37.4 ± 2.0	23.8 ± 0.6	29.3 ± 3.9	-	26.2 ± 1.8	26.6 ± 1.6	**22.8 ± 1.9**
	TGAT	11.8 ± 0.1	10.2 ± 0.2	10.0 ± 0.1	-	10.7 ± 0.2	9.6 ± 0.1	**9.3 ± 0.2**
	DySAT	18.4 ± 0.1	14.5 ± 0.1	14.5 ± 0.3	-	14.5 ± 0.2	14.4 ± 0.2	**14.3 ± 0.2**
UCI	TGN	44.2 ± 0.4	42.5 ± 0.3	44.2 ± 0.9	43.0 ± 0.5	42.5 ± 0.5	42.9 ± 0.8	**41.6 ± 0.3**
	JODIE	24.9 ± 0.4	21.9 ± 0.5	22.3 ± 0.4	21.9 ± 0.2	21.8 ± 0.6	**21.4 ± 0.5**	21.7 ± 0.2
	TGAT	16.3 ± 0.3	15.5 ± 0.3	14.7 ± 0.1	14.9 ± 0.2	14.7 ± 0.2	15.7 ± 0.4	**14.6 ± 0.2**
	DySAT	71.3 ± 0.9	65.3 ± 0.7	65.3 ± 0.3	65.3 ± 0.5	65.4 ± 0.3	65.3 ± 0.6	**64.7 ± 0.4**
Bitcoin	TGN	48.3 ± 0.7	48.0 ± 1.3	48.0 ± 1.0	48.4 ± 0.9	46.8 ± 0.5	46.2 ± 0.7	**45.2 ± 1.4**
	JODIE	34.6 ± 0.7	31.4 ± 0.5	30.3 ± 0.6	32.1 ± 0.6	29.2 ± 0.8	**27.9 ± 0.9**	28.9 ± 0.7
	TGAT	21.8 ± 0.4	18.6 ± 0.3	17.3 ± 0.3	18.5 ± 0.4	17.1 ± 0.4	**16.6 ± 0.6**	17.1 ± 0.2
	DySAT	84.7 ± 0.7	79.0 ± 0.8	78.9 ± 0.2	79.4 ± 0.6	79.0 ± 0.4	79.0 ± 0.2	**78.8 ± 0.2**
A.P.D.↓		-	-14.7%	-14.9%	-7.5%	-16.3%	-16.4%	**-18.9%**

*Jaccard cannot be applied to bipartite graphs.

- **MRR (Mean Reciprocal Rank) as Perturbation Rate Increases**



- **Adversarial Edge Classification Accuracy**

Metric: AUROC
Perturbation rate (p): 0.3

Attack	Model	Wikipedia	MOOC	UCI
Random	TGN-Cosine	0.605	0.633	0.495
	T-Shield-F	0.848	0.697	0.531
	T-Shield	**0.850**	**0.709**	**0.534**
T-Spear	TGN-Cosine	0.474	0.556	0.518
	T-Shield-F	0.858	**0.679**	0.608
	T-Shield	**0.864**	0.674	**0.620**

➢ **T-SHIELD-F:** use only edge filtering

- **Defense Performance**

Under Random attack

Model	Wikipedia (Clean: 80.5 ± 0.5)			MOOC (Clean: 61.8 ± 2.1)			UCI (Clean: 44.2 ± 0.4)			A.P.G.↑
	$p=0.1$	$p=0.2$	$p=0.3$	$p=0.1$	$p=0.2$	$p=0.3$	$p=0.1$	$p=0.2$	$p=0.3$	
TGN	73.2 ± 1.1	68.3 ± 0.9	66.0 ± 0.6	58.7 ± 1.0	57.8 ± 0.9	55.4 ± 1.7	44.5 ± 0.3	43.5 ± 0.7	42.5 ± 0.3	-
TGN-SVD	65.3 ± 1.1	63.9 ± 1.2	60.6 ± 0.9	58.8 ± 0.3	58.0 ± 1.4	55.0 ± 1.1	**44.5 ± 0.7**	**44.3 ± 0.5**	**44.7 ± 0.7**	-2.1%
TGN-Cosine	75.8 ± 0.5	71.6 ± 0.7	68.8 ± 0.4	53.7 ± 2.7	52.8 ± 4.1	54.0 ± 3.2	43.2 ± 0.6	43.0 ± 0.6	43.8 ± 0.4	-0.9%
T-Shield-F	**77.4 ± 0.5**	**77.4 ± 0.2**	76.0 ± 0.3	59.2 ± 1.0	**58.8 ± 1.3**	58.0 ± 0.6	44.1 ± 0.2	44.2 ± 0.6	44.2 ± 0.7	+5.1%
T-Shield	77.3 ± 0.5	77.0 ± 0.3	**76.5 ± 0.5**	**59.4 ± 1.2**	58.4 ± 1.1	**58.4 ± 0.6**	43.9 ± 0.6	44.3 ± 0.7	44.6 ± 0.8	**+5.3%**

❖ **Clean**: MRR when attack is not applied.

Under T-SPEAR attack

Model	Wikipedia (Clean: 80.5 ± 0.5)			MOOC (Clean: 61.8 ± 2.1)			UCI (Clean: 44.2 ± 0.4)			A.P.G.↑
	$p=0.1$	$p=0.2$	$p=0.3$	$p=0.1$	$p=0.2$	$p=0.3$	$p=0.1$	$p=0.2$	$p=0.3$	
TGN	71.0 ± 2.0	63.7 ± 2.6	60.0 ± 1.6	55.2 ± 0.6	52.4 ± 0.4	48.2 ± 1.3	43.6 ± 1.0	41.7 ± 0.5	41.6 ± 0.3	-
TGN-SVD	67.6 ± 0.9	61.2 ± 1.4	58.1 ± 1.8	55.7 ± 1.3	50.6 ± 2.6	50.1 ± 2.1	**46.0 ± 0.5**	45.5 ± 0.4	**46.5 ± 0.4**	+1.8%
TGN-Cosine	72.3 ± 0.8	62.9 ± 2.0	59.6 ± 2.1	53.5 ± 3.1	47.6 ± 1.7	45.2 ± 2.1	43.7 ± 0.4	44.1 ± 0.3	43.2 ± 1.3	-1.0%
T-Shield-F	**77.3 ± 0.5**	76.8 ± 0.3	75.3 ± 0.4	58.5 ± 0.5	55.7 ± 0.9	**54.5 ± 0.9**	44.4 ± 0.8	**45.5 ± 0.3**	44.7 ± 0.4	+11.0%
T-Shield	77.1 ± 0.2	**76.9 ± 0.3**	**76.1 ± 0.3**	**58.8 ± 0.7**	**56.3 ± 1.2**	53.9 ± 1.1	44.9 ± 0.3	44.3 ± 0.5	45.4 ± 0.8	**+11.2%**