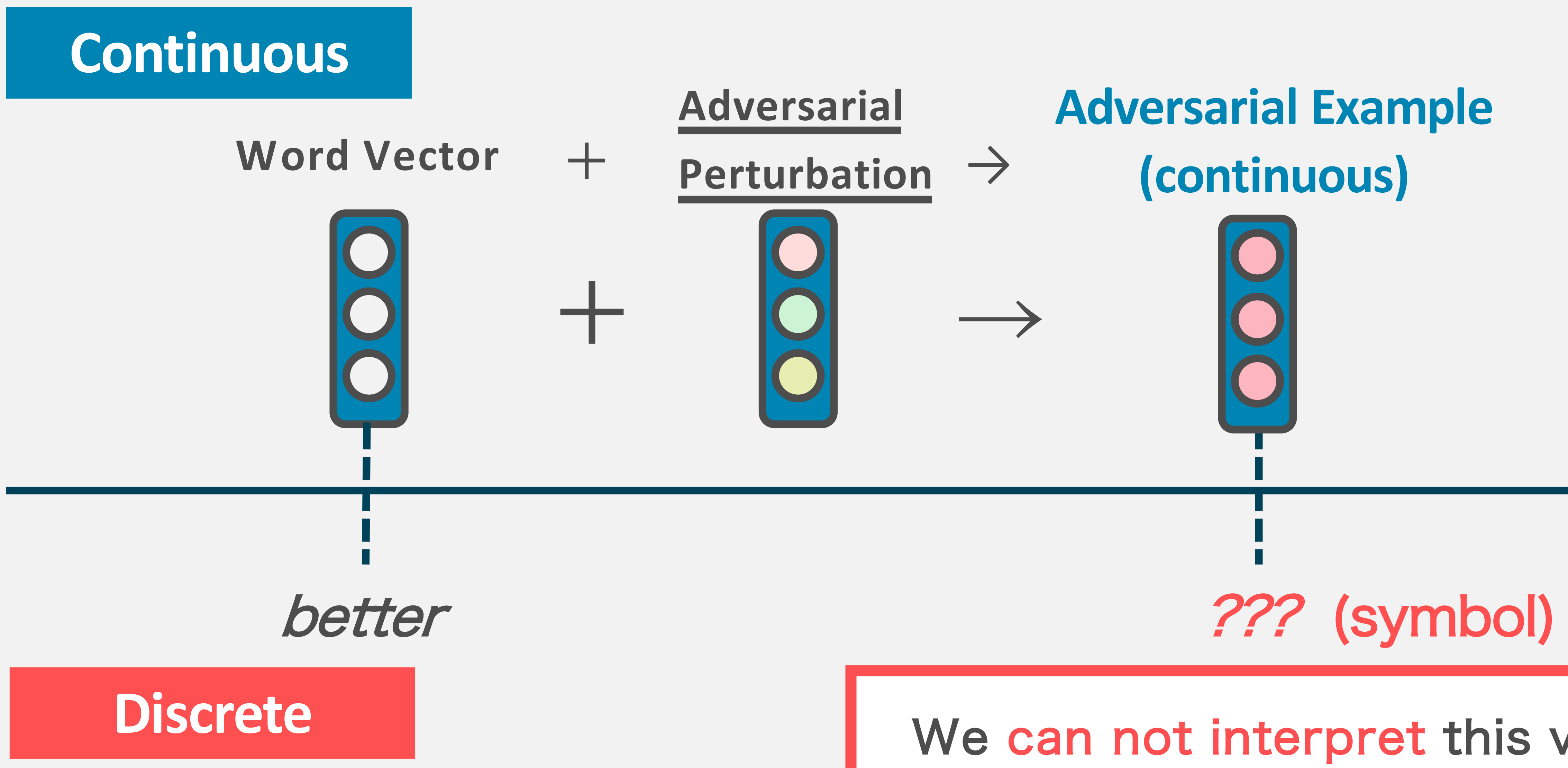
Interpretable Adversarial Perturbation in Input Embedding Space for Text

Motoki Sato [1], Jun Suzuki [2,4], Hiroyuki Shindo[3,4], Yuji Matsumoto[3,4]

[1] Preferred Networks, Inc.
[2] NTT Communication Science Laboratories
[3] Nara Institute of Science and Technology
[4] RIKEN Center for Advanced Intelligence Project

Overview of this paper

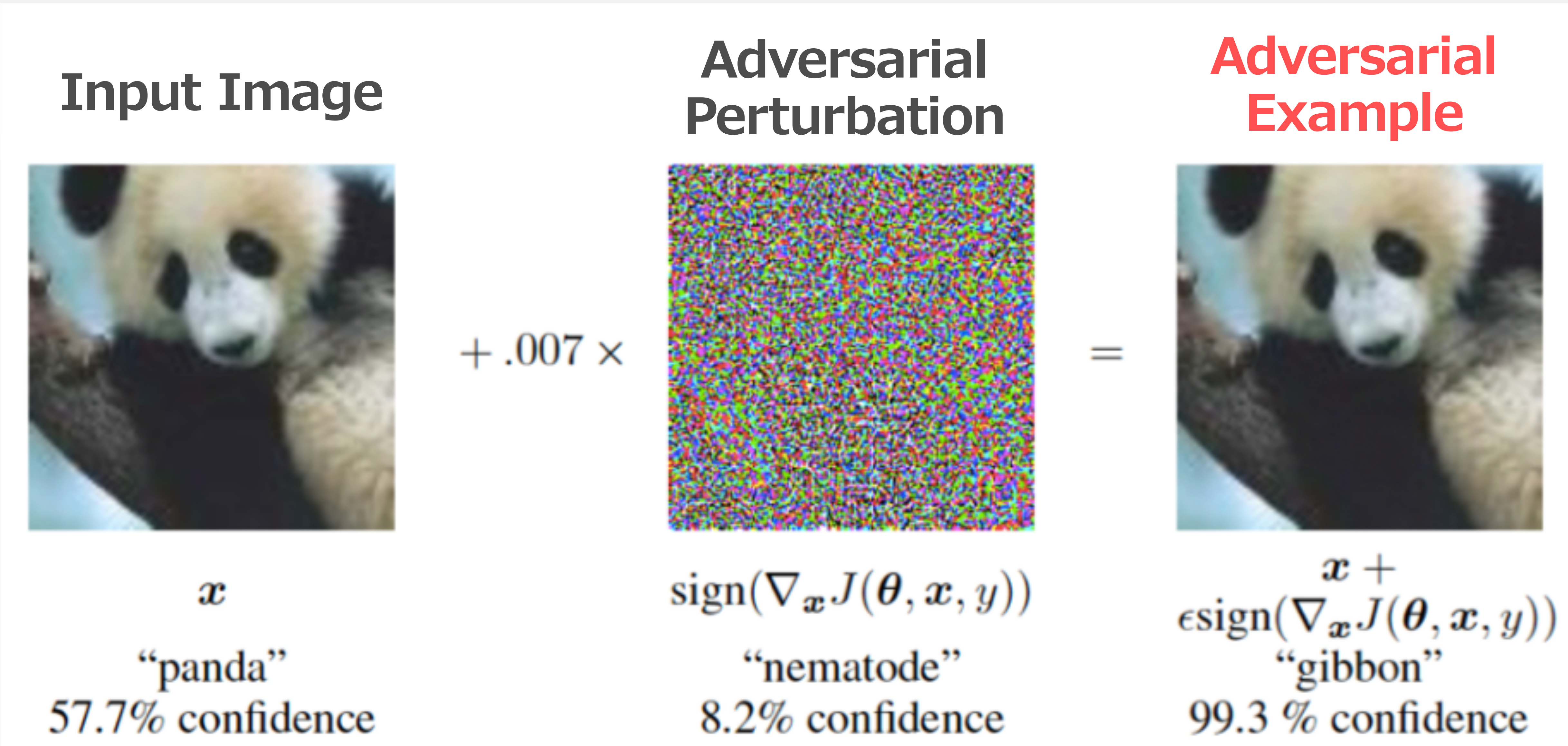
- Adversarial Training for Text



We can not interpret this vector in discrete space

Adversarial Perturbation

- **Adversarial perturbations** induce **prediction error**
  [Szegedy *et al.*, 2014, Goodfellow *et al* .,2015]



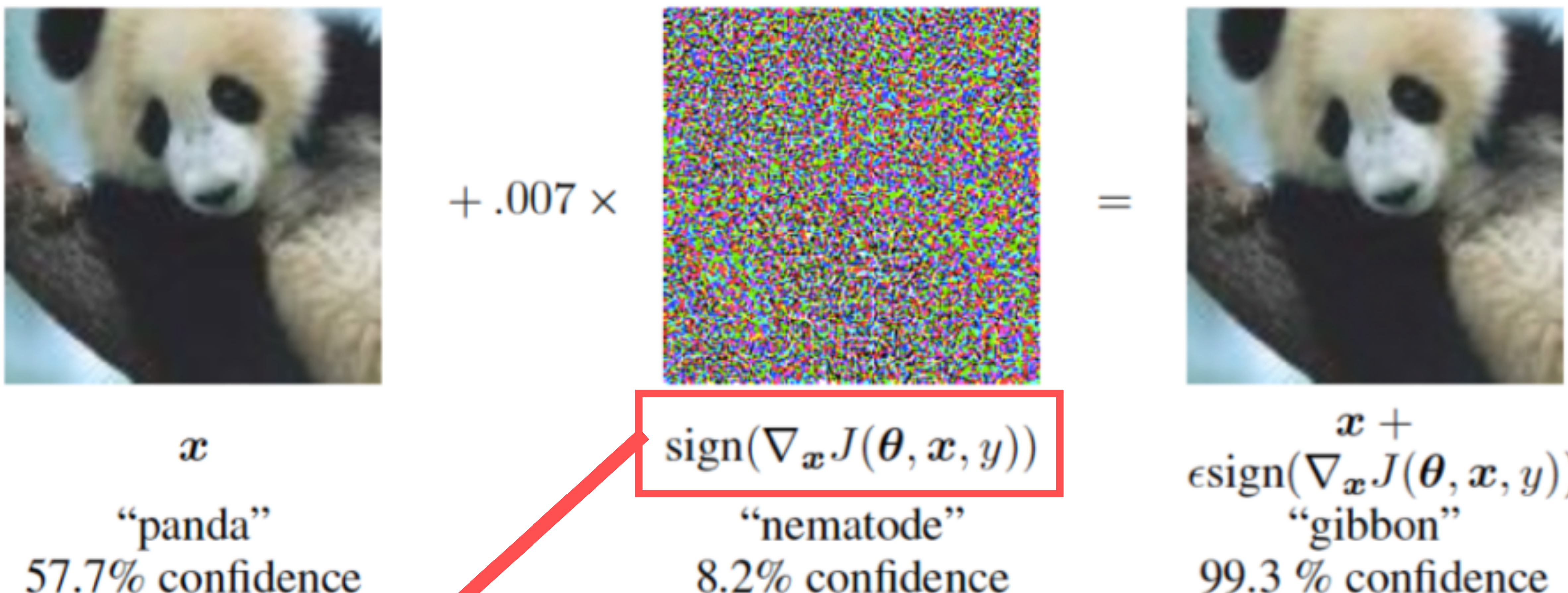
[Goodfellow *et al.*, 2015]

- [Szegedy *et al.*, 2014] : "Intriguing properties of neural networks.", ICLR 2014.
- [Goodfellow *et al.*,2015]:"Explaining and Harnessing Adversarial Examples", ICLR 2015.

Adversarial Perturbation

- **Adversarial perturbations** induce **prediction error**
  [Szegedy *et al.*, 2014, Goodfellow *et al* .,2015]

Input Image | **Adversarial Perturbation** | **Adversarial Example**



[Goodfellow *et al.*, 2015]

The gradient of loss function

Adversarial Training

Adversarial Training

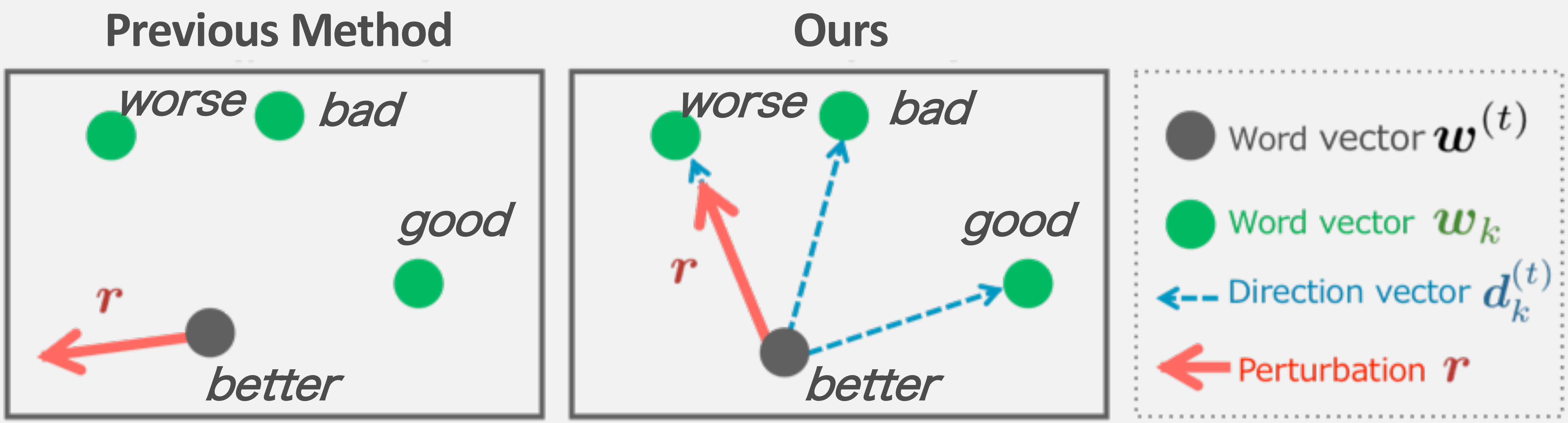
- **Improve generalization performance** [Goodfellow *et al* .,2015]

$$\tilde{J}(\boldsymbol{\theta}, \boldsymbol{x}, y) = \underbrace{\alpha J(\boldsymbol{\theta}, \boldsymbol{x}, y)}_{\text{Original Training Data}} + \underbrace{(1-\alpha) J(\boldsymbol{\theta}, \boldsymbol{x} + \epsilon \operatorname{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))}_{\text{Adversarial Example}}$$

Adversarial Training for Text

- **Adversarial Perturbation** to **Word Vector** **[Miyato *et al.*, 2017]**
  - Although they achieved state-of-the-art in text classification, interpretability of perturbation is not discussed

Our main idea



[Miyato *et al.*, 2017]

- **Restrict** the **directions** of the perturbations
- **Restrict** toward the **locations** of **existing words.**

Related Work

Related Work

How to create Adversarial Example for Text?

- **Human Knowledge** [Jia and Liang, 2017]

  Fooling **Reading Comprehension System** using **crowdsourcing**

- **Random search** [Belinkov and Bisk, 2018]

  Random character-level swaps can break output of Neural Machine Translation

- **Synonym dictionary** [Samanta and Mehta, 2017]

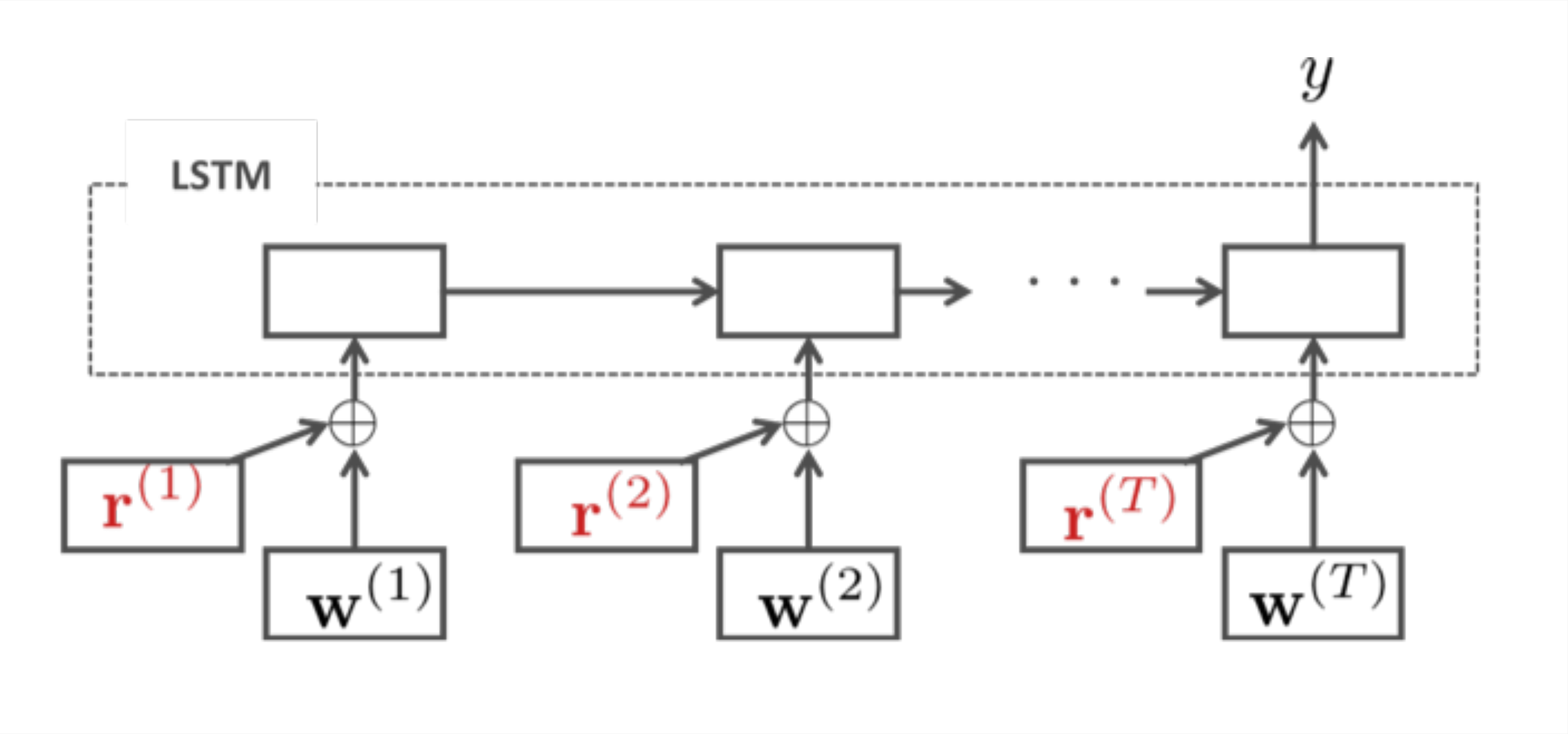
  Replacing a word with its **synonym**

- Existing methods require **human knowledge** or **heuristic**

Previous Method [Miyato *et al.*, 2017]

- **Takeru Miyato**, Andrew M Dai, and Ian Goodfellow, **ICLR 2017**

  "Adversarial training methods for semi-supervised text classification."
- **Uni-directional LSTM** + **Pre-Training** (Language Model) + Adversarial Training



Adversarial Perturbation : $\boldsymbol{r}^{(t)}$

Word Vector : $\boldsymbol{w}^{(t)}$

Previous Method [Miyato *et al.*, 2017]

Adversarial Perturbation : $\boldsymbol{r}$

Word Vector : $\boldsymbol{w}^{(t)}$

Definition

ϵ : hyper-parameter (e.g.: 1.0)

$$\tilde{X}_{+\boldsymbol{r}} = (\boldsymbol{w}^{(t)} + \boldsymbol{r}^{(t)})_{t=1}^{T}$$

: Word vector with perturbation

$$\boldsymbol{r}_{\mathtt{AdvT}} = \underset{\boldsymbol{r}, ||\boldsymbol{r}|| \le \epsilon}{\operatorname{argmax}} \left\{ \ell(\tilde{X}_{+\boldsymbol{r}}, \tilde{Y}, \mathcal{W}) \right\}$$

: Find $\boldsymbol{r}$ to increase the loss function ℓ

How to obtain the perturbation

$$\boldsymbol{r}^{(t)}_{\mathtt{AdvT}} = \frac{\epsilon \boldsymbol{g}^{(t)}}{||\boldsymbol{g}||_2}, \quad \boldsymbol{g}^{(t)} = \nabla_{\boldsymbol{w}^{(t)}} \ell(\tilde{X}, \tilde{Y}, \mathcal{W})$$

: Compute the gradient with L2 normalization

Our Method

Adversarial Perturbation : $\boldsymbol{r}$

Word Vector : $\boldsymbol{w}^{(t)}$

Definition

$$\boldsymbol{d}_k^{(t)} = \frac{\tilde{\boldsymbol{d}}_k^{(t)}}{||\tilde{\boldsymbol{d}}_k^{(t)}||_2}, \quad \text{where} \quad \tilde{\boldsymbol{d}}_k^{(t)} = \boldsymbol{w}_k - \boldsymbol{w}^{(t)}$$

: Direction Vector

$$\boldsymbol{r}(\boldsymbol{\alpha}^{(t)}) = \sum_{k=1}^{|\mathcal{V}|} \alpha_k^{(t)} \boldsymbol{d}_k^{(t)}$$

: $\alpha_k^{(t)}$ is a weight for the direction

How to obtain the perturbation?

$$\boldsymbol{\alpha}_{\mathtt{iAdvT}}^{(t)} = \frac{\epsilon \boldsymbol{g}^{(t)}}{||\boldsymbol{g}||_2}, \quad \boldsymbol{g}^{(t)} = \nabla_{\boldsymbol{\alpha}^{(t)}} \ell(\tilde{X}_{+\boldsymbol{r}(\boldsymbol{\alpha})}, \tilde{Y}, \mathcal{W})$$

: Compute α with the gradient

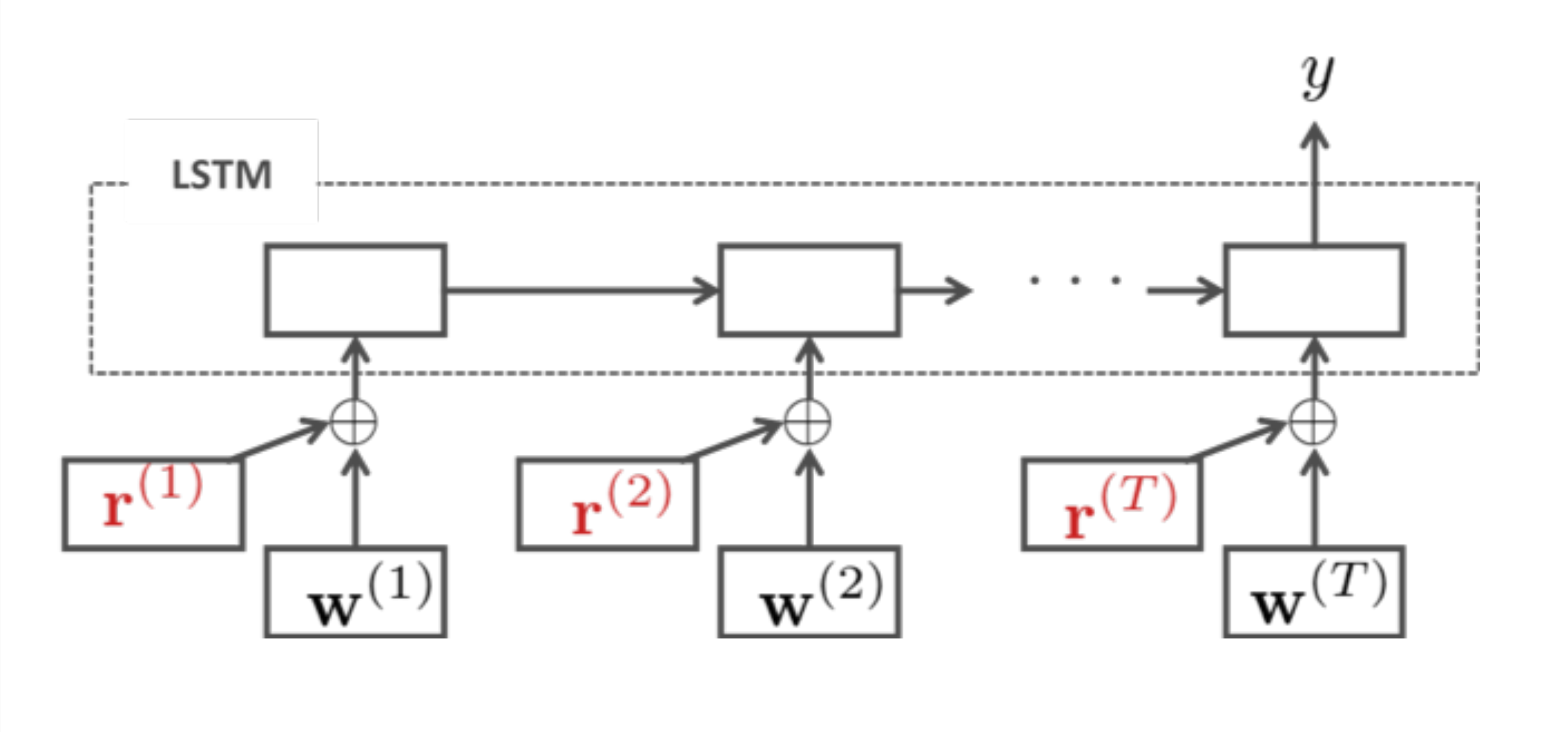
$$\boldsymbol{r}(\boldsymbol{\alpha}^{(t)}) = \sum_{k=1}^{|\mathcal{V}|} \alpha_k^{(t)} \boldsymbol{d}_k^{(t)}$$

: Compute the perturbation

Experiments

Experiments

- **Setting**
  - Text Classification : IMDB (Sentiment Analysis)
  - Sequence Labeling : FCE-public (Grammatical Error Detection)



Adversarial Perturbation $\boldsymbol{r}^{(t)}$

Word Vector : $\boldsymbol{w}^{(t)}$

Evaluation by task performance

SEC: Sentiment Classification Task (IMDB)

GED: Grammatical Error Detection Task (FCE-public)

	Test Error rate (SEC)	$F_{0.5}$ (GED)
Baseline	7.05 (%)	39.21
Random Perturbation	6.74 (%)	39.90
AdvT-Text [Miyato et al., 2017]	6.12 (%)	**42.28**
iAdvT-Text (Ours)	6.08 (%)	42.26
VAT-Text [Miyato et al., 2017]	5.69 (%)	41.81
iVAT-Text (Ours)	**5.66 (%)**	41.88

- maintaining or even improving the task performance.

Model Analysis

Visualization of sentence-level perturbations

Test sentence (Positive)

Words reconstructed from perturbations

Ours

We visualized the perturbations for understanding its behavior.

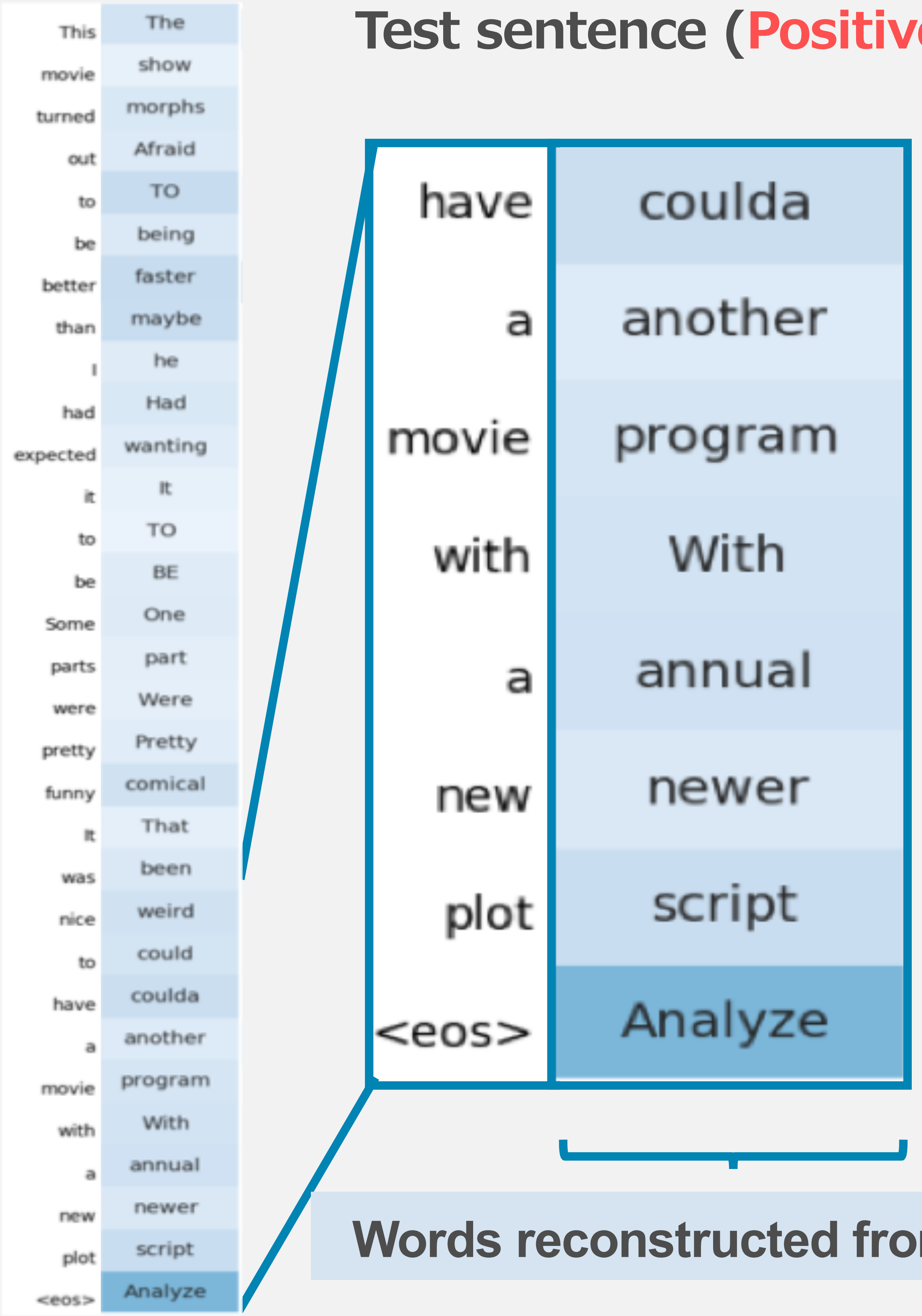
Left : Input sentence (test data) (**Positive class**)

Right : Words reconstructed from perturbations

Our method found that directions for replacing "**better**" → "**worse**" to increase the loss

Visualization of sentence-level perturbations

Test sentence (Positive)



Words reconstructed from perturbations

[Miyato *et al.*, 2017]

Cosine similarities between perturbation and words.

Previous method found that directions for replacing **“<eos>”** → **“Analyze”** (uninterpretable)

Left : Input sentence (test data) (**Positive class**)

Right : Most cosine similar words

Creating Adversarial Example

Test Text | **Predict: Negative**

There is really but one thing to say about **this** sorry movie It should never have been made The first one one of my favourites An American Werewolf in London is a great movie with a good plot good actors and good FX But this one It stinks to heaven with a cry of helplessness <eos>

Adversarial Example | **Predict: Positive**

There is really but one thing to say about **that** sorry movie It should never have been made The first one one of my favourites An American Werewolf in London is a great movie with a good plot good actors and good FX But this one It stinks to heaven with a cry of helplessness <eos>

Find the largest perturbation and replace the original word with one that matches the largest perturbation.

Conclusion

- We discussed the **interpretability** of **adversarial perturbation** in the **NLP (Text) field**.
- Our methods can generate reasonable **adversarial texts** and interpretable **visualizations**.

Code: https://github.com/aonotas/interpretable-adv

Thank you!

Adversarial Example

Original Text (Incorrect)

We all want to thank you for having choose such good places in London .

Prediction

0 0 0 0 0 0 0 0 1 0 0 0 0 0 0

Adversarial Example

We all want to thank you for having choosing such good places in London .

Prediction

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Incorrect → Correct