

# Gradient Clipping

---

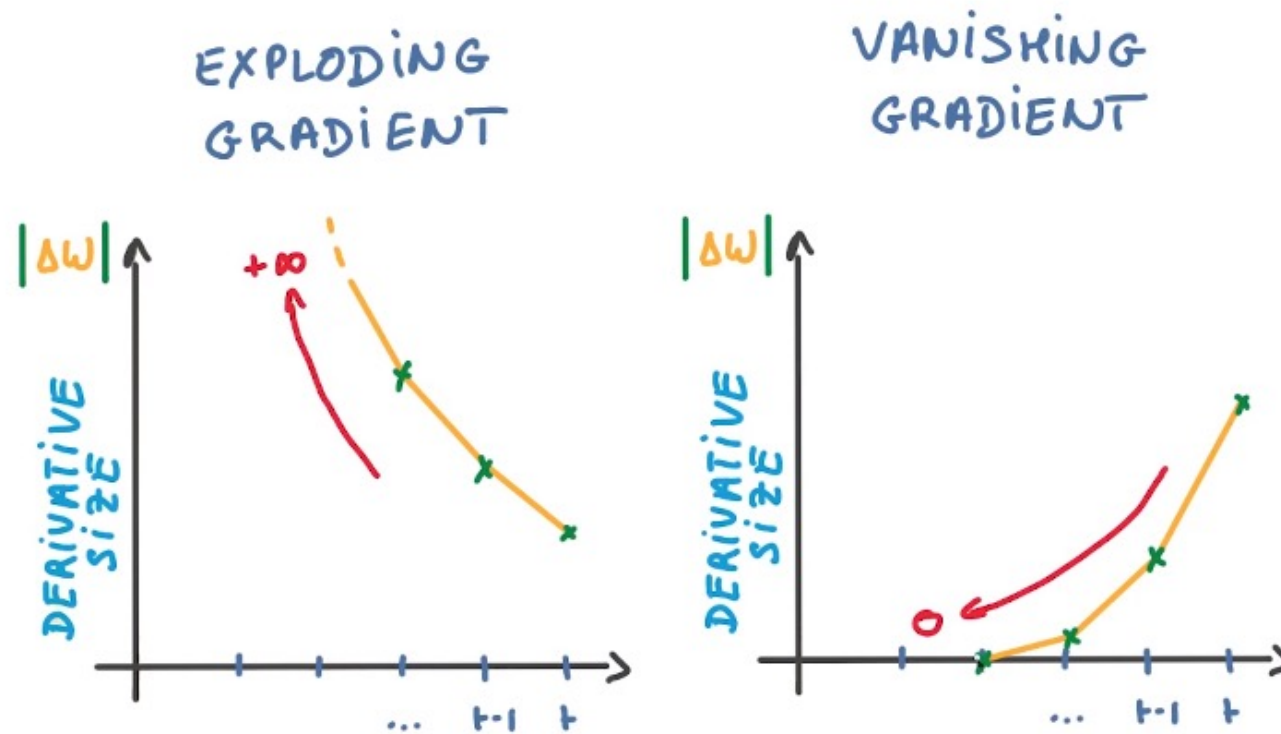
에이아이스쿨(AISchool) 대표  
양진호 (솔라리스)

<http://aischool.ai>

<http://solarisailab.com>

# Exploding gradient & Vanishing gradient

- 계산과정에서 Vanishing gradient problem과 반대로 Gradient가 무한히 커지는 **Exploding Gradient Problem**이 발생할 수 있습니다.
- RNN은 순환 연산 구조를 갖기 때문에 Exploding Gradient Problem이 발생하기 더 쉽습니다.



## Exploding gradient Problem 해결책 – Gradient Clipping

- 다행히 Exploding Gradient Problem은 Gradient Clipping이라는 방법론을 통해서 손쉽게 해결할 수 있습니다.

---

Algorithm 1 Pseudo-code for norm clipping

---

$$\hat{\mathbf{g}} \leftarrow \frac{\partial \mathcal{E}}{\partial \theta}$$
$$\text{if } \|\hat{\mathbf{g}}\| \geq \text{threshold} \text{ then}$$
$$\quad \hat{\mathbf{g}} \leftarrow \frac{\text{threshold}}{\|\hat{\mathbf{g}}\|} \hat{\mathbf{g}}$$
$$\text{end if}$$

---

## TensorFlow에서 Gradient Clipping을 적용하는 법

- [https://github.com/solaris33/deep-learning-tensorflow-book-code/blob/master/Ch08-RNN/8.5-linear\\_regression\\_with\\_gradient\\_clipping\\_v2\\_keras.py](https://github.com/solaris33/deep-learning-tensorflow-book-code/blob/master/Ch08-RNN/8.5-linear_regression_with_gradient_clipping_v2_keras.py)

```
26 # 최적화를 위한 function을 정의합니다.
27 @tf.function
28 def train_step(model, x, y):
29     with tf.GradientTape() as tape:
30         y_pred = model(x)
31         loss = mse_loss(y_pred, y)
32     gradients = tape.gradient(loss, model.trainable_variables)
33     # Gradient Clipping을 적용
34     clipped_grads = []
35     for grad in gradients:
36         clipped_grads.append(tf.clip_by_norm(grad, grad_clip))
37     optimizer.apply_gradients(zip(clipped_grads, model.trainable_variables))
```

# Thank you!

---