



# Bidirectional\_LSTM을 사용한 악플 탐지 및 제거 프로그램

발표자 노태현

발표일 2019-12-13

# 목차

## 1. 개요 및 구성

- 1-1 만들게 된 배경
- 1-2 전체적인 구성도

## 2. Word2Vec 간략 설명

- 2-1 모델 설명
- 2-2 시각화

## 3. Bi\_LSTM 설명

- 3-1 LSTM 모델 설명
- 3-2 Bi\_LSTM 모델 설명
- 3-3 epoch와 accuracy

## 4. 시연 동영상

- 4-1 시연영상

## 5. 문제점 및 해결방법

- 5-1 문제점
- 5-2 내가 생각한 해결방법

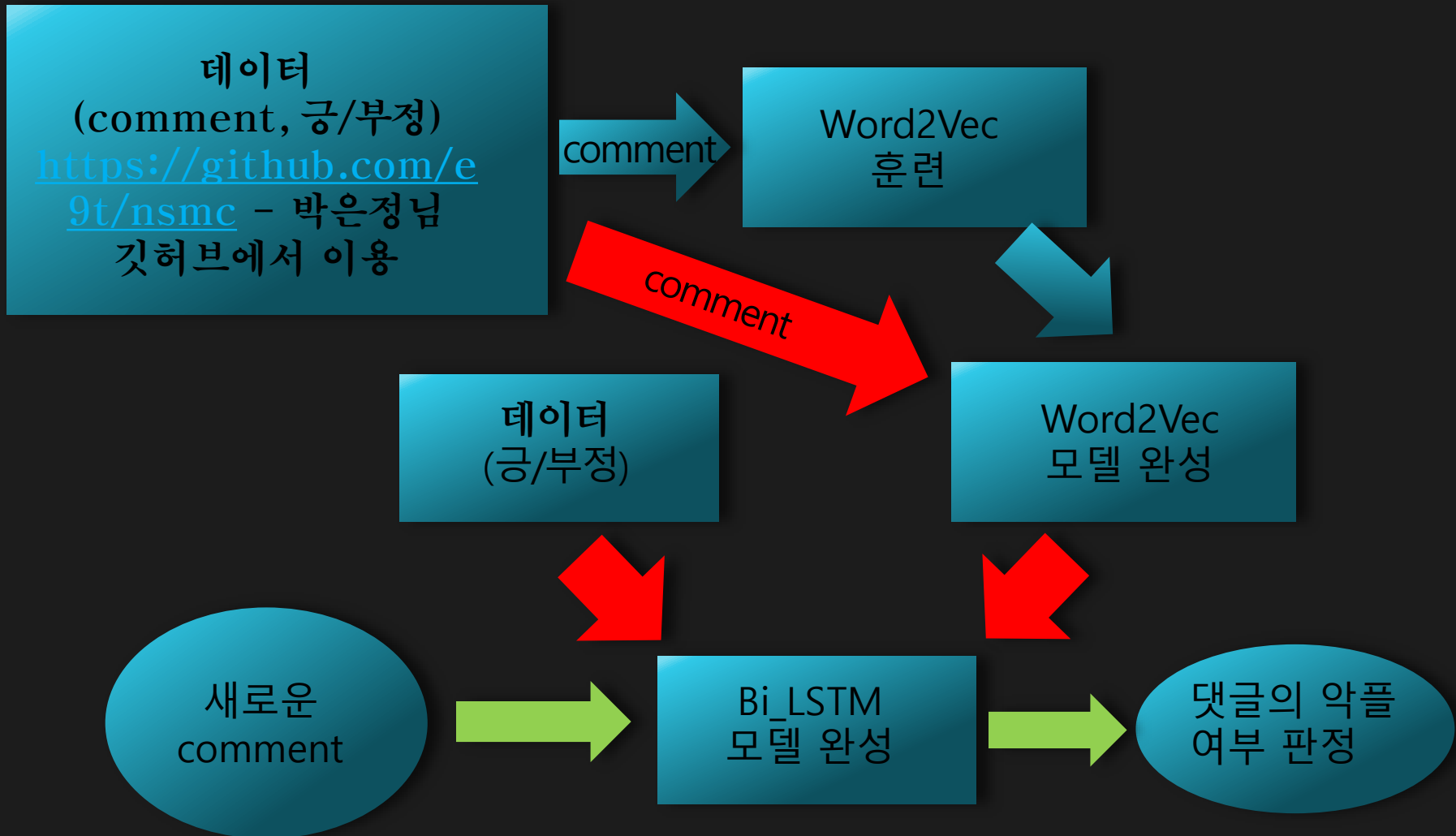
## Chapter 1

# 개요 및 구성

## 만들게 된 배경



## 전체적인 구성도

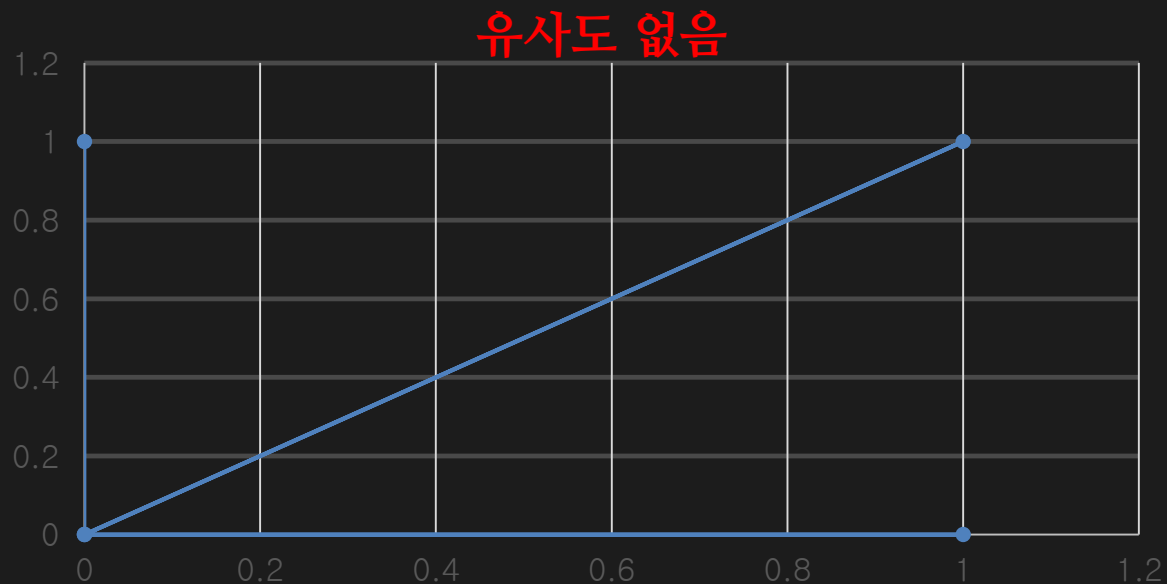


## Chapter 2

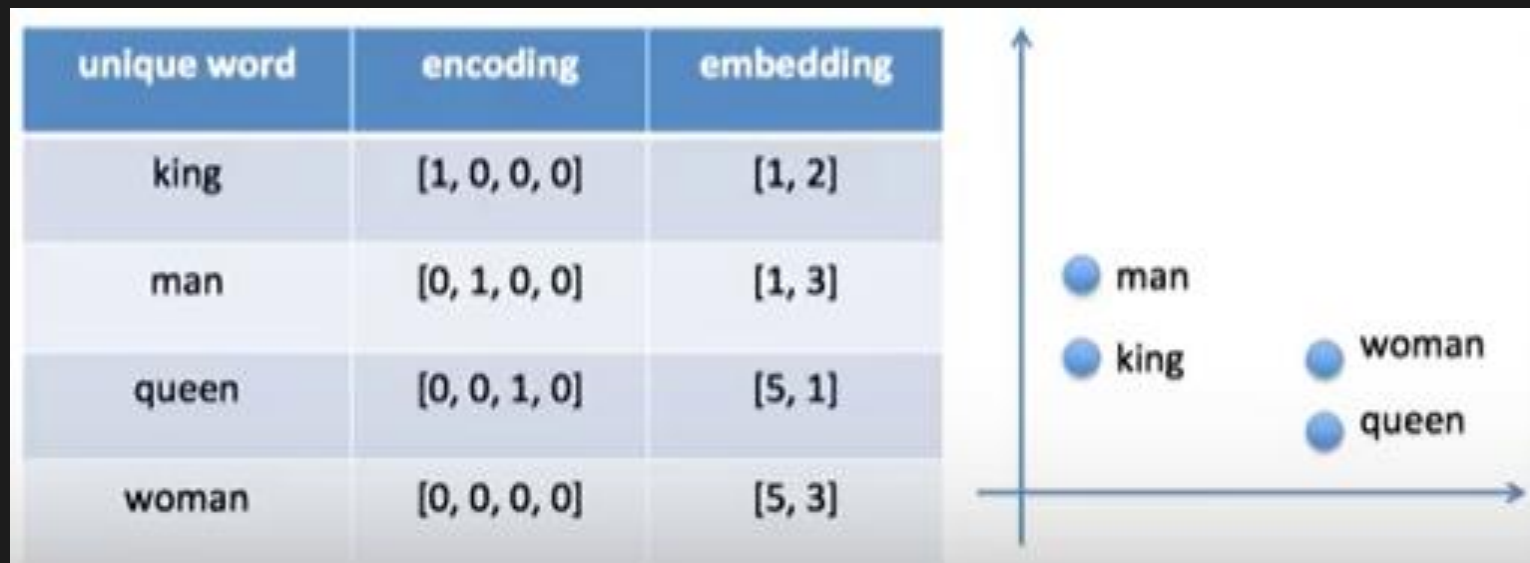
# Word2Vec

## Word2Vec

	Thank	You	love
Thank	1	0	0
You	0	1	0
love	0	0	1

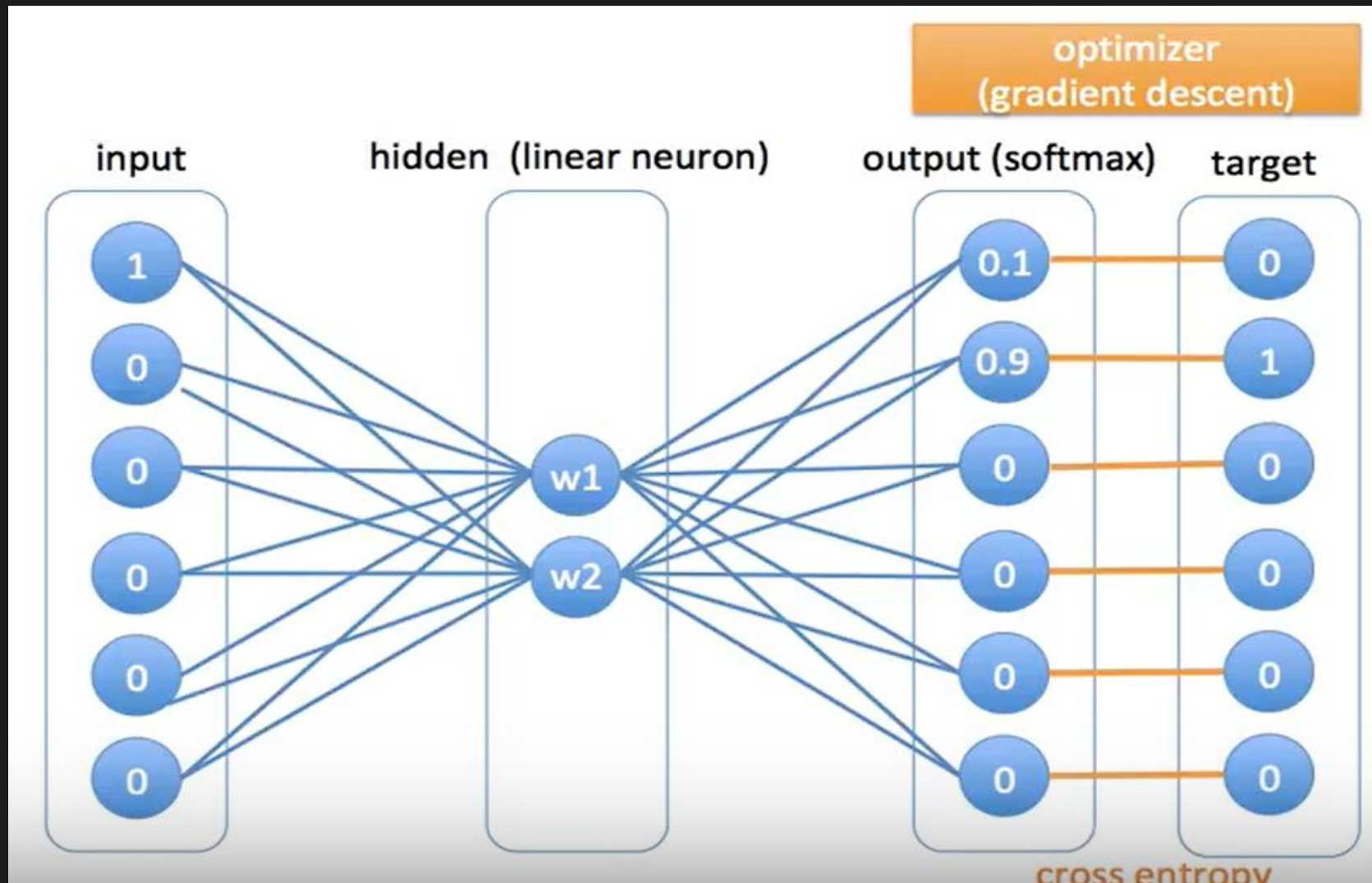


## Word2Vec





## Word2Vec



## Word2Vec

King =  
[1,0,0,0,0,0]

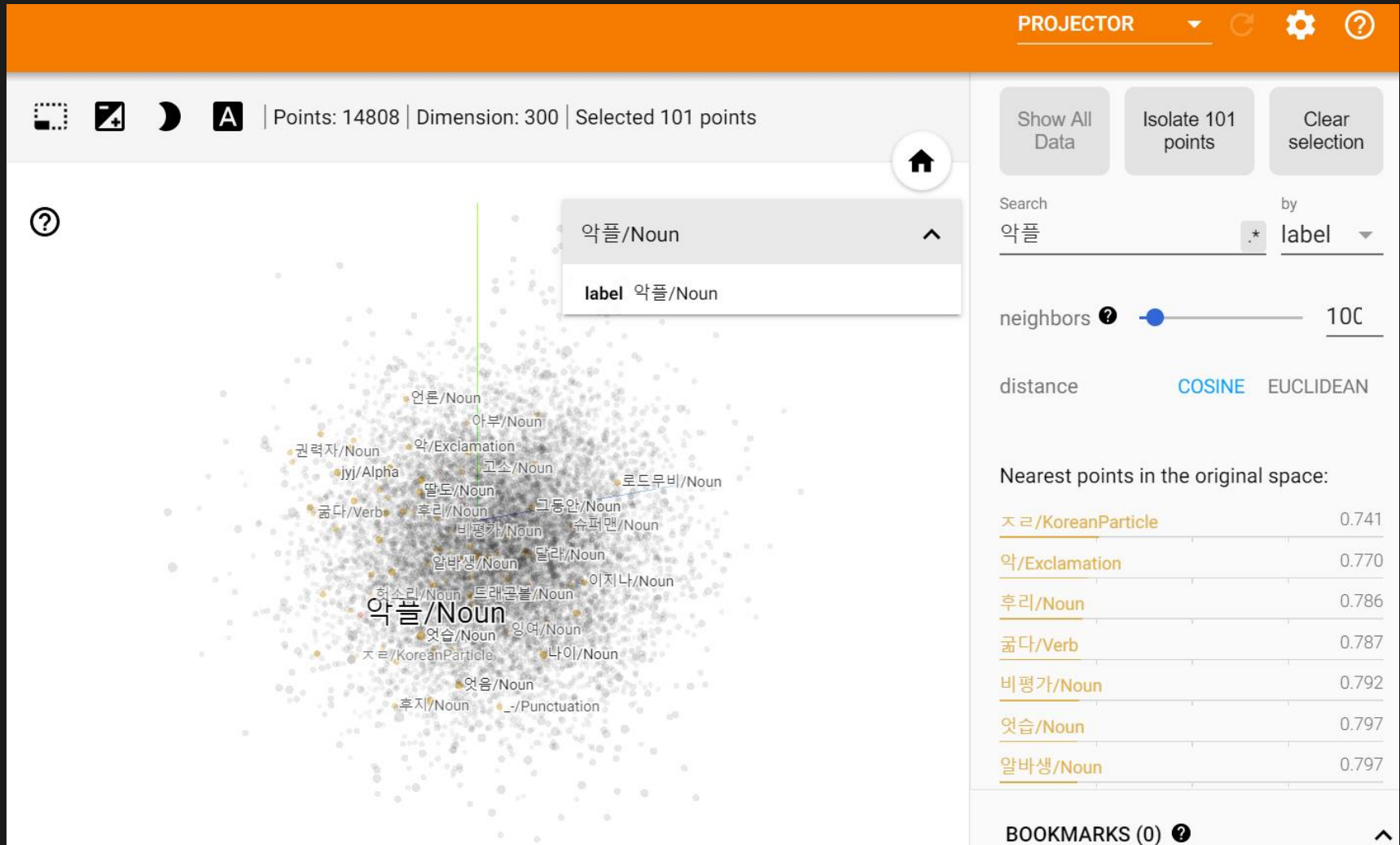
Brave=  
[0,1,0,0,0,0]



[1,1]  
[1,2]  
[1,3]  
[5,5]  
[5,6]  
[5,7]

unique word	embedding
king	[1, 1]
brave	[1, 2]
man	[1, 3]
queen	[5, 5]
beautiful	[5, 6]
woman	[5, 7]

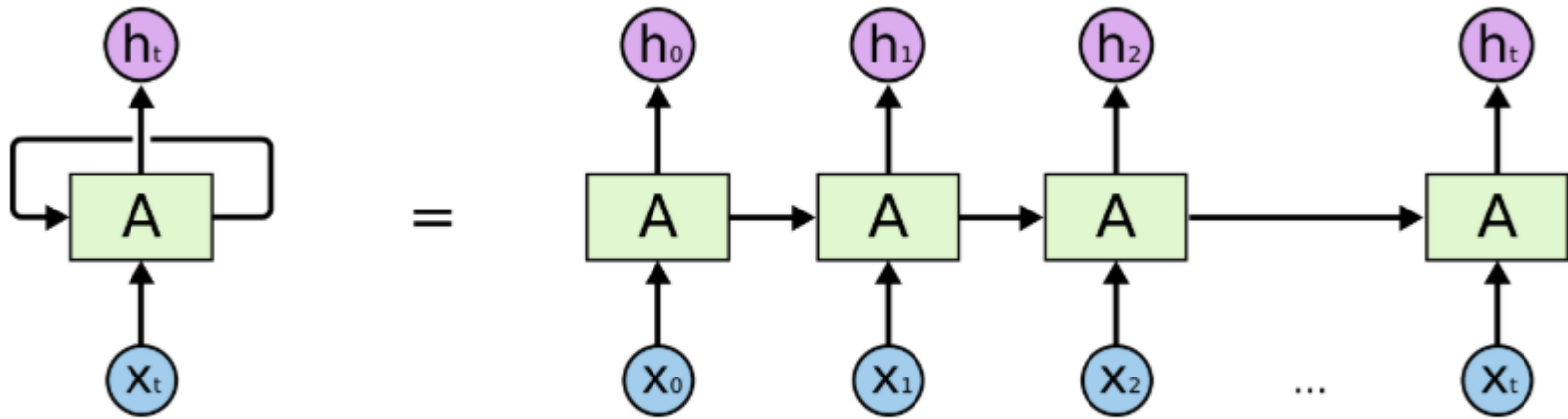
## Word2Vec



## Chapter 3

# Bidirectional\_LSTM

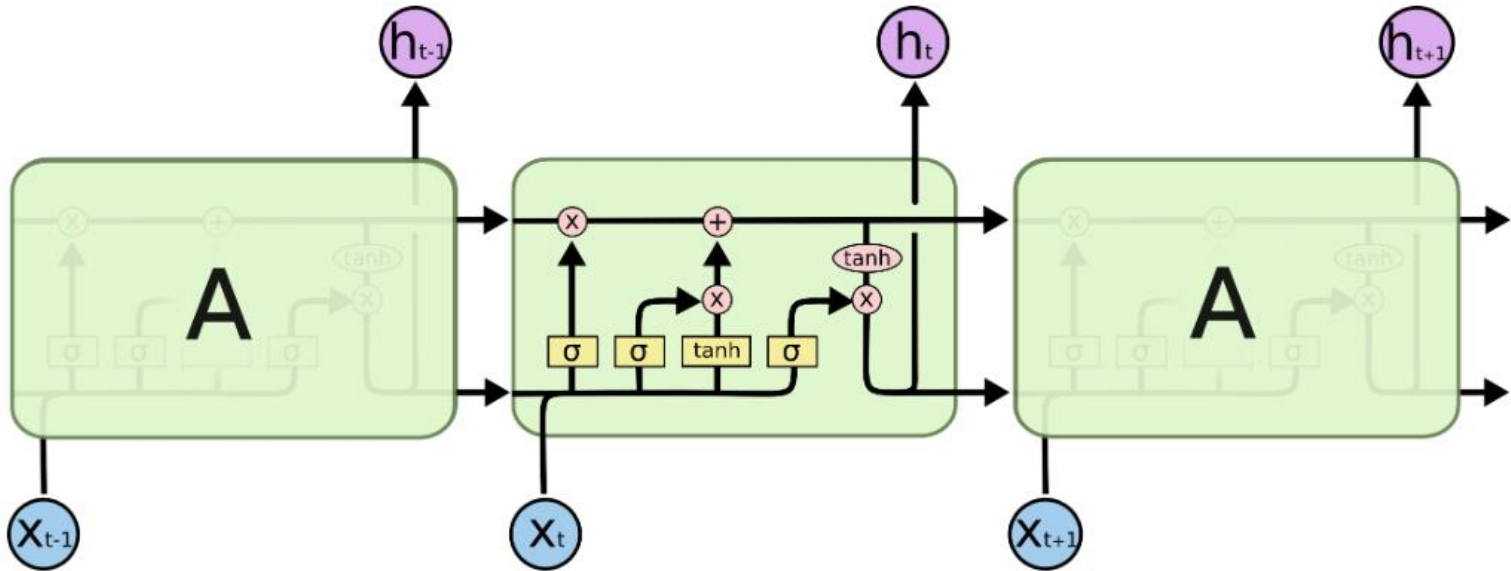
## LSTM



An unrolled recurrent neural network.

기존 RNN의 경우, Vanishing Gradient Problem / Exploding Problem이 발생. 즉, 차원 수가 많아지게 되면서 앞 단어들을 기억하지 못하게 됨

## LSTM



LSTM의 경우, 이 문제를 해결.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

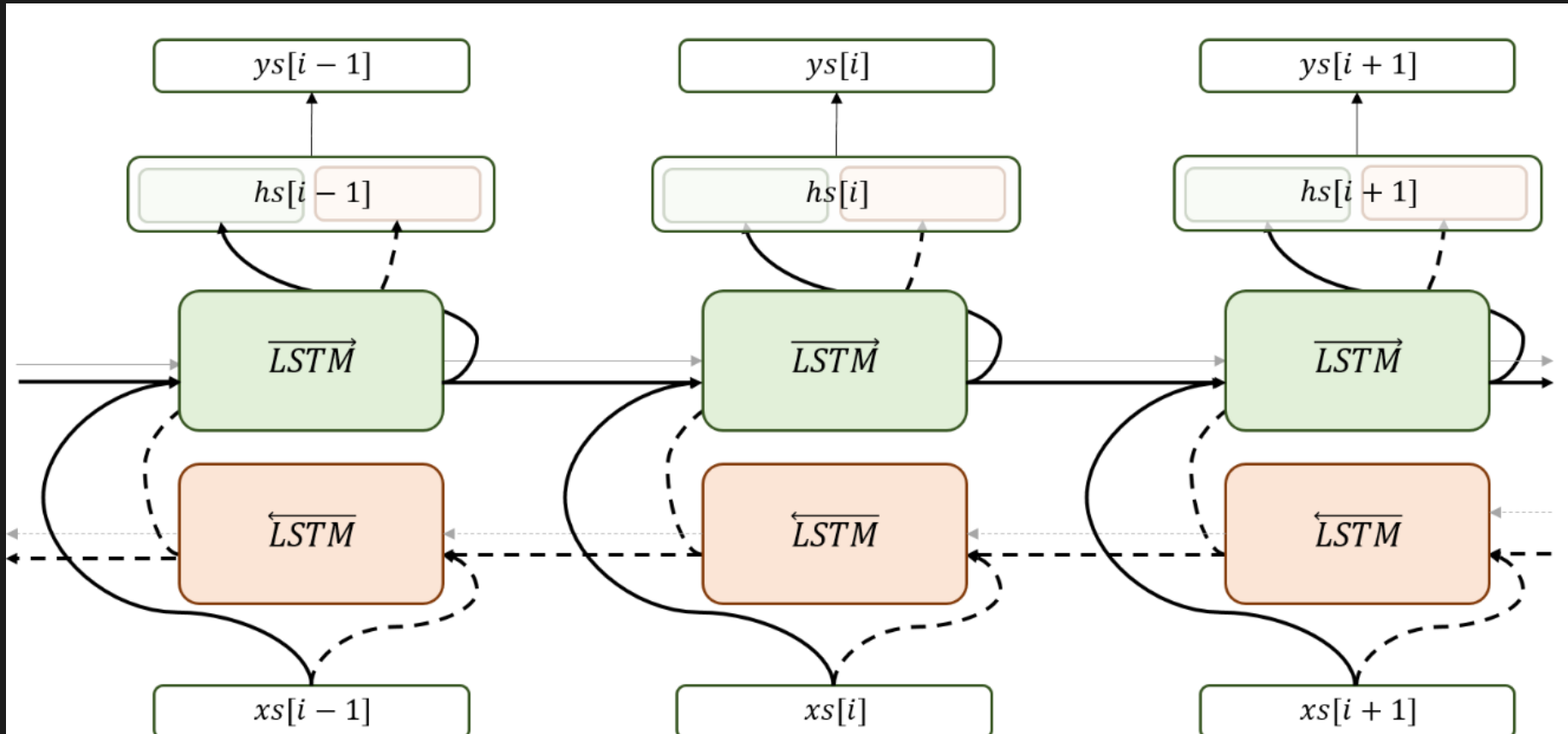
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

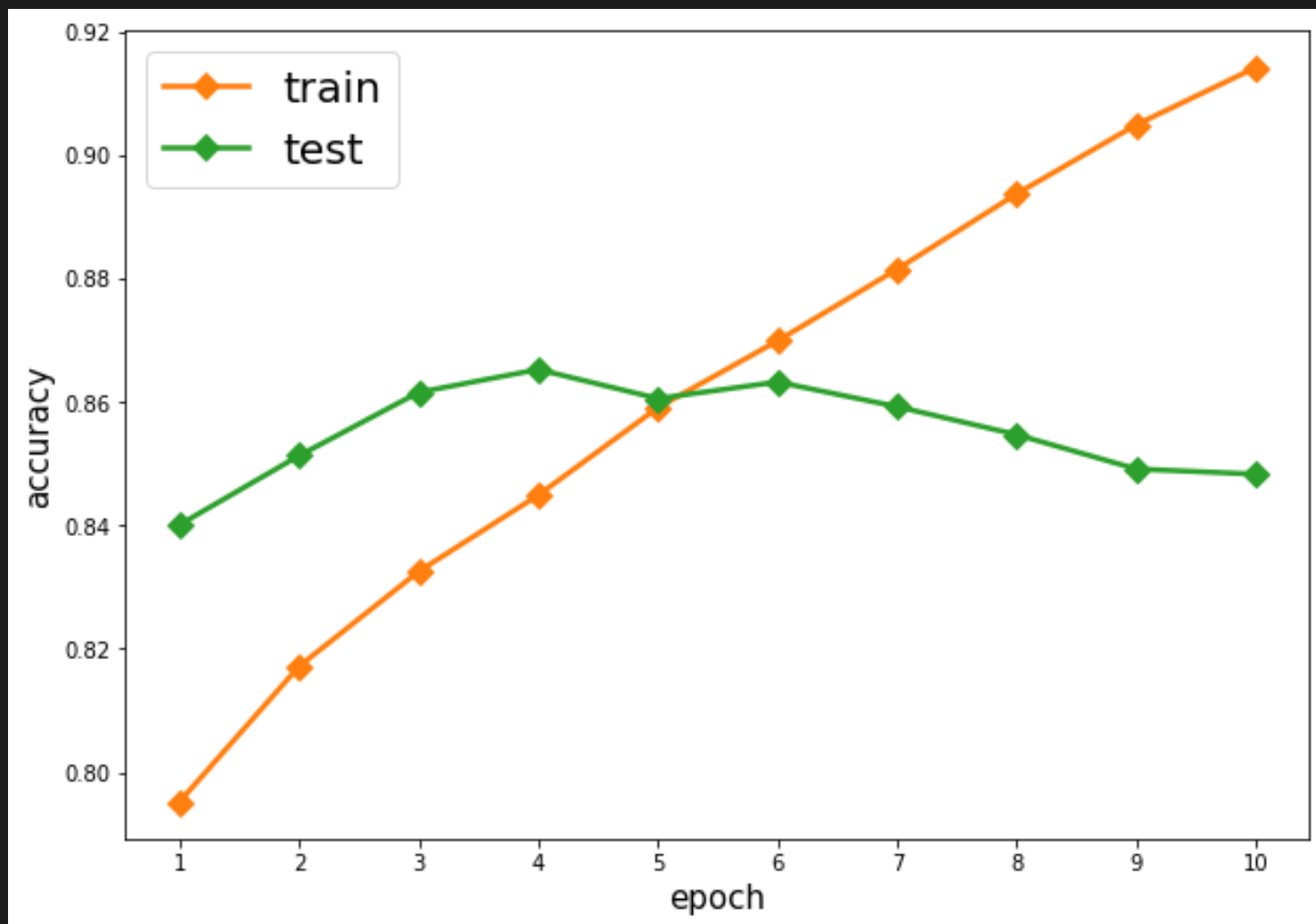
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

## Bi\_LSTM



## Epoch와 accuracy





## Chapter 4

# 시연영상

LSTM &gt; Bad\_Comment\_Detection\_main.py

Bad\_Comment\_Detection\_main

Get\_News\_articles\_and\_Comment.py x Identification\_Of\_Malicious\_Comments.py x Show\_News.py x Bad\_Comment\_Detection\_main.py x

```
8 import Identification_Of_Malicious_Comments
9 from selenium import webdriver
10 import time
11
12 driver = webdriver.Chrome('C:\\Users\\NohTaeHyun\\Desktop\\Bad_Comment_Detection_Project\\chromedriver_win32\\chromedriver.exe')
13 Unknown_Comment = []
14 ArticleTitle = []
15 Article = []
16 Known_Comment = []
17 _temp = []
18 while True:
19     cur_url = driver.current_url
20     # if driver.current_url != cur_url:
21     #     cur_url = driver.current_url
22     #     print(cur_url)
23     #     print(cur_url[28:32])
24     if cur_url[28:32] == "read":
25         Unknown_Comment = Get_News_articles_and_Comment.get_replies(cur_url, driver)
```

Bad\_Comment\_Detection\_main x

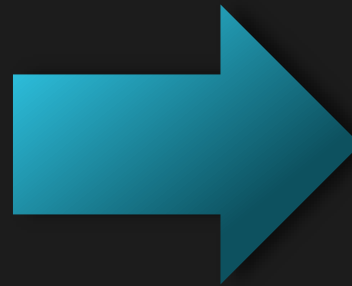
폴입니다.

Process finished with exit code -1

# 문제점 및 해결방법

배경상황에 따라 악플  
의 기준이 달라짐.

당사자에 따른 악플의  
기준이 다름



뉴스 카테고리 별로  
ML모델을 만들게  
된다면 다소 배경상  
황에 맞는 악플탐지  
를 할 수 있을 것으  
로 기대됨.

감사합니다.

