

실험제목	데이터셋 공부
실험목적	데이터 셋에 대한 깊은 이해

Continued from page :

train과 test를 분리하는 이유?

- 정확히 말하자면 train / validation으로 볼수 있음.
- 머신러닝 모델에 train 데이터를 100% 학습 시킨 후 test 데이터/ 모델을 적용했을 때. 성능이 생각보다 안나오는 경우가 많음

(이를 overfitting 이라 함)

즉, 모델이 생각보다 내가 가진 학습 데이터에 너무 과적합 되도록 학습한 나머지. 이를 조금이라도 벗어난 케이스에 대해서는 예측률이 현저히 떨어짐

train | validation | test
점도.

위와 같이 train / test 로 구분되어있는 데이터셋을 중간중간

validation 셋으로 내가 학습한 모델을 평가 해주는 것.

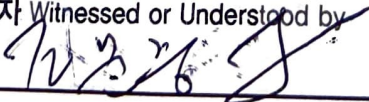
안정 모델이 과적합 되었다면 validation 셋으로 검증가능.

* 사이킷런의 model_selection 패키지 안에 train-test split

모델을 활용하여 손쉽게 train set test set 을

분리할 수 있음.

Continued to page :

기록자 Written by	점검자 Witnessed or Understood by 	점검자 Witnessed or Understood by
일자 Date 2022. 01. 3	일자 Date	일자 Date

실험제목	LSTM 설명	목차참조
실험목적	더 자세하게 LSTM을 이해하고자하는 목적.	본문참조

Continued from page :

*LSTM 설명

Long - Short Term Memory.

RNN은 과거의 데이터 값을 기억함. 시계열데이터로
이전 과거의 값이 현재의 값에 영향을
미치는 것.

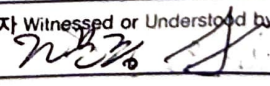
RNN의 layer가 deep해질 수록 점점 과거의 값들이 희석됨.
즉, 일반적인 Neural Network에서 발생했던
'Vanishing Gradient'와 같은 문제가 발생.

예를들어, $t-1$ 의 값들은 t 상태에서 잘 가지고 있지만,
 $t-100$

$t-1000 \sim$ 의 값은 거의 영향력이 X.

따라서 과거의 데이터들이 희석되는 문제점을 고쳐준 것이
바로 LSTM 이다.

Continued to page :

기록자 Written by	점검자 Witnessed or Understood by 	점검자 Witnessed or Understood by
일자 Date 2022.01.10	일자 Date	일자 Date

실험제목

원 핫 인코딩 공부.

실험목적

원 핫 인코딩 하는 방법 이해.

Continued from page :

* one-hot-encoding.

- dummy 변수 개념을 활용하여 새롭게 코딩 하는 방법.

예를 들어 items = ['TV', '냉장고', '전자레인지', '컴퓨터', '선풍기', '선풍기', '믹서', '믹서'] 3 외이었다 가정

① 일단 레이블 인코딩 과정을 거쳐 문자열이 숫자 하나씩 대칭.

encoder = LabelEncoder() 즉

② 2차원 배열로 변환

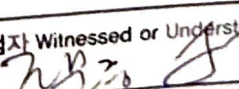
labels = labels.reshape(-1,1)

③ One Hot Encoder 함수를 변수에 할당시켜 디아터 fit.

array [[1,0,0,0,0],
[0,1,0,0,0] ... 등등

위의 과정을 거치면 2차원 배열의 인코딩 값을 생성할 수 있음.
변환을 매겨 주는 느낌임!

Continued to page :

기록자 Written by	점검자 Witnessed or Understood by 	점검자 Witnessed or Understood by
일자 Date 2022. 11. 17	일자 Date	일자 Date

실험제목	레이블 인코딩 공부
실험목적	레이블 인코딩 하는 방법 이해.

Continued from page :

* label encoding

- 문자열로 구성되어 있는 items 데이터를
'카테고리화' (코딩화) 하는 것임.

EX) items = [TV, 냉장고, 전자레인지, 컴퓨터, 선풍기, 선풍기,
믹서, 믹서] 로 가정

- label encoder 함수에서 지원하는 transform 기능을 활용하여
items 데이터를 인코딩 시키고, 이를 label에 저장
: labels = encoder.transform(items)

⇒ 결과

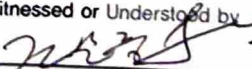
array([0, 1, 4, 5, 3, 3, 2, 2], dtype=int64)

↓ 인코더가 랜덤으로 배치한 인코딩 순서임

TV, 냉장고, 믹서 ~

- encoder.inverse_transform 함수를 사용하면
인코딩된 데이터를 역으로 되돌릴 수 있다.

Continued to page :

기록자 Written by	점검자 Witnessed or Understood by 	점검자 Witnessed or Understood by
일자 Date 2022.01.24	일자 Date	일자 Date