딥러닝 및 응용

RNN

2015004475 김태훈

1. 구동 환경

OS: Windows 10 64비트

Language: Python 3.7

환경설정은 conda를 이용하였으며 필요 패키지는 requirements.txt에 기재됨.

1. 코드 설명

sample: 샘플 문자열. ‘if you want you’

idx\_to\_char: 샘플 문자열에 있는 알파벳을 중복 없이 리스트화.

char\_to\_idx: 어떤 문자의 인덱스가 몇 번인지 딕셔너리 형태로 반환.

dict\_size: 문자 개수.

hidden\_size: RNN 셀에 들어갈 하이퍼 파라미터.

num\_classes: classify 결과 나올 클래스 개수.

sample\_idx: 샘플 문자열의 문자들을 인덱스 번호 리스트로 반환.

x\_data: ‘if you want yo’

y\_data: ‘f you want you’

X, Y: 학습할 X, Y 플레이스홀더.

x\_one\_hot: 원핫 인코딩

cell: RNN LSTM 셀.

initial\_state: 0으로 채워진 state tensor.

outputs, \_states: 위에서 정의한 RNN셀 동작 결과.

X\_for\_fc: fully connected layer에 들어갈 X, RNN셀 아웃풋을 flatten.

outputs: fc 레이어를 통과시켜 나온 결과물 reshape.

weights: 1로 채운 weight 매트릭스.

sequence\_loss: outputs와 ground truth를 비교한 loss

loss: sequence\_loss reduce mean.

train: adam 옵티마이저, loss 최소화.

prediction: output에 대한 argmax.

세션을 가동한 후…

1. 텐서 variable을 랜덤 초기화.
2. 50회의 epoch을 반복하며
   1. x\_data, y\_data를 모델에 전달하여 학습.
   2. 현재 학습 상태로 x\_data를 통해 예측한 prediction 계산.
   3. loss와 prediction 출력
3. 실험 결과



