

파이토치로 배우는 딥러닝 기초

RNN 퀴즈 정답



Quiz 1 해설

Quiz

정답 Q 1 (단일선택, 10점)

Recurrent Neural Networks(이하 RNN) 구조를 적용하여 처리하기에 적합하지 않은 것은 무엇인가?

- ☐ 음성인식 (sound wave → 텍스트)
- ☐ 기계번역 (영어 → 한국어)
- ☒ 이미지 분류 (이미지 → Label)
- ☐ 목소리로 성별 분류 (sound wave → 성별)

1

2

3

4

5

6

7

8

9

10

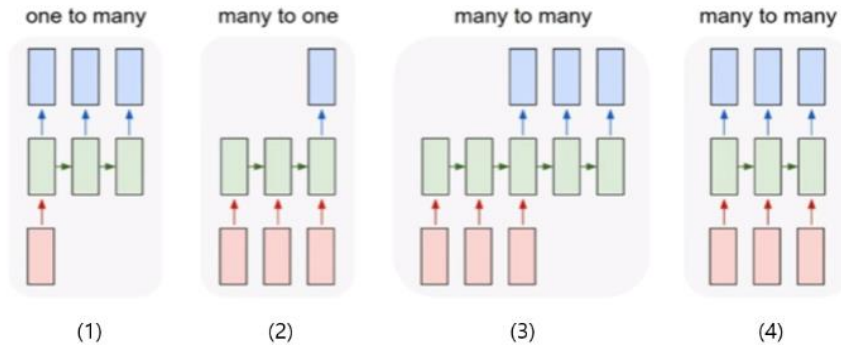
"음성인식, 기계번역, 목소리로 성별 분류" 세가지 모두 RNN 구조를 사용합니다. 이미지 분류는 보통 CNN 을 사용합니다.

Quiz 2 해설

Quiz

정답 Q 2 (단일선택, 10점)

RNN의 구조와 이것이 적용된 Task로 잘못 짝지어진 것은?



1. Image Captioning (이미지 → 문장) : 이미지를 가져와 이를 설명하는 문장을 출력
2. Sentiment Classification (문장 → 감정) : 문장을 분석하여 감정을 출력
3. Video Classification on frame level (영상 → 문장) : 비디오 영상의 변화에 대한 분석 결과
4. Machine Translation (문장 → 문장) : 한국어 문장을 영어 문장으로 번역

- ☐ 1
☐ 2
☐ 3
☒ 4

1 2 3 4 5 6 7 8 9 10

기계번역(Machine Translation)은 3번 의 구조를 가지고 있습니다.

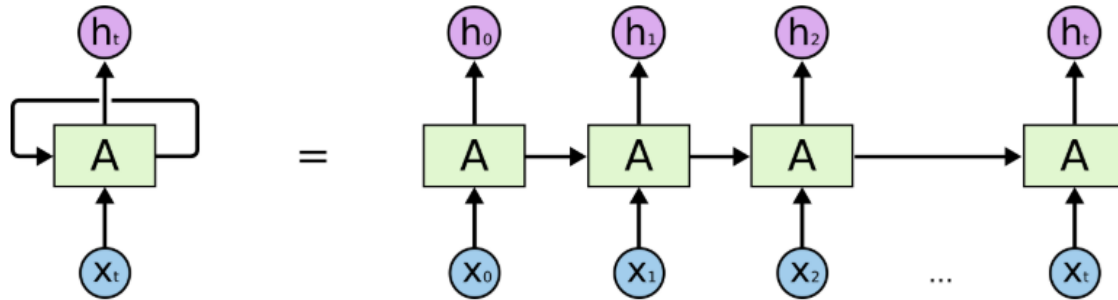
Quiz 3 해설

Quiz

정답 Q 3 (단일선택, 10점)

다음 문구의 진위 여부를 판별하세요.

"RNN에서 전방향 전파(feed-forward) 시, 매 step(0, 1, 2, ..., t)를 진행할 때마다 같은 값의 Hidden Layer(A) parameters(Weight, Bias 등)를 적용하여 Hidden state(h_0, h_1, \dots, h_t)를 생성해준다."



- ☒ True
☐ False

1 2 3 4 5 6 7 8 9 10

같은 순환 내에서 각 Hidden Layer의 함수 및 parameter들은 같은 값을 공유하며, 이들과의 연산으로 나온 각 step 별 Hidden state를 계산해줍니다.

Quiz 4 해설

Quiz

정답 Q 4 (단일선택, 10점)

RNN을 구축하는 과정에서 다음과 같은 코드를 작성하였다.

```
1 rnn = torch.nn.RNN(input_size, hidden_size)
2 outputs, _status = rnn(input_data)
```

이 때, input_data의 크기에 들어갈 정보로 알맞은 것은?

- Batch size: 미니배치 크기
- Input Dimension: 입력 차원
- Sequence Length: 시퀀스 길이

- ☐ (Batch Size, Input Dimension, Sequence Length)
- ☒ (Batch Size, Sequence Length, Input Dimension)
- ☐ (Input Dimension, Batch Size, Sequence Length)
- ☐ (Input Dimension, Sequence Length, Batch Size)

1 2 3 4 5 6 7 8 9 10

RNN 모델을 정의할 시 batch_first=True를 설정해 놓으면 입력 데이터의 형태는 (Batch Size, Sequence Length, Input Dimension) 가 되어 합니다. 만약에 False 로 설정하면 입력 데이터의 형태는 (Sequence Length, Batch Size, Input Dimension) 가 됩니다.

Quiz 5 해설

Quiz

정답 Q 5 (단일선택, 10점)

다음과 같이 주어진 sentence 문장을 RNN 모델에 적용시키기 위하여 One-hot Encoding을 하려고 한다. 빈칸에 들어갈 것으로 알맞은 것은?

```
1 sentence = "if you want you"
2 char_set = list(set(sentence))
3 char_dic = {c: i for i, c in enumerate(char_set)}
4 dic_size = len(char_dic)
5
6 sample_idx = [char_dic[c] for c in sentence]
7 x_data = [sample_idx[:-1]]
8 x_one_hot = [ " 빈칸 " for x in x_data ]
9 y_data = [sample_idx[1:]]
```

- ☐ numpy.flatten(dic_size)[x]
- ☒ numpy.eye(dic_size)[x]
- ☐ torch.Tensor.view(dic_size)[x]
- ☐ torch.Tensor.reshape(dic.size)[x]

1 2 3 4 5 6 7 8 9 10

PyTorch의 [eye 함수](#) 혹은 NumPy의 [eye 함수](#)를 이용해서 dic_size(char_dic의 크기)의 차원으로 one-hot encoding 해주는 코드입니다.

Quiz 6 해설

Quiz

정답 Q 6 (단일선택, 10점)

이전 문제와 같이 One-hot Encoding된 문장 데이터 X를 가지고 다음 문자를 예측하는 알고리즘을 짜기 위해 학습시키려고 한다. 다음 중 Loss Function으로 사용될 코드를 고르시오.

```
1 # declare RNN
2 rnn = torch.nn.RNN(input_size, hidden_size, batch_first=True)
3
4 # loss & optimizer setting
5
6 criterion = ### [Loss 코드] ###
7 optimizer = optim.Adam(rnn.parameters(), learning_rate)
8
9 # start training
10 for i in range(100):
11     optimizer.zero_grad()
12     outputs, _status = rnn(X)
13     loss = criterion(outputs.view(-1, input_size), Y.view(-1))
14     loss.backward()
15     optimizer.step()
16
17 result = outputs.data.numpy().argmax(axis=2)
18 result_str = ''.join([char_set[c] for c in np.squeeze(result)])
19 print(i, "loss: ", loss.item(), "prediction: ", result,
20       "true Y: ", y_data, "prediction str: ", result_str)
```

- ☒ torch.nn.CrossEntropyLoss()
- ☐ torch.nn.MSELoss()

1 2 3 4 5 6 7 8 9 10

분류 문제에서 사용되는 알맞은 손실 함수는 Cross-Entropy Loss 입니다.

Quiz 7 해설

Quiz

정답 Q 7 (단일선택, 10점)

다음 문구의 진위 여부를 판별하세요.

"RNN모델은 sequence가 길어질수록 parameter들이 업데이트되며 Gradient가 소실되거나 폭발하는 Vanishing/Exploding Gradient 문제를 가지고 있다."

☒ True

☐ False

1 2 3 4 5 6 7 8 9 10

Layer가 깊어질수록 작은 숫자가 계속 곱해지면 Vanishing Gradient문제를, 큰 숫자가 계속 곱해지면 Exploding Gradient 문제가 생깁니다.

Quiz 8 해설

Quiz

정답 Q 8 (단일선택, 10점)

LSTM(Long Short-Term Memory) 모델에 대한 설명으로 옳지 않은 것은?

- ☐ RNN 모델의 Vanishing Gradient 문제를 해결하기 위하여 제안된 모델이다.
- ☒ 긴 Sequence 데이터에 이용하기에 적합하지 않은 모델이다.
- ☐ 기존 Hidden State에 Cell State가 추가된 구조이다.
- ☐ 3개의 Gate가 추가되어 정보를 선택적으로 통과시킬 수 있다.

1 2 3 4 5 6 7 8 9 10

LSTM은 Cell state를 추가함으로써 Sequence가 길더라도 Gradient가 잘 전달되어 장기 의존성 (Long-term Dependency) 문제가 해결되어 긴 Sequence 데이터에 적합합니다.

Quiz 9 해설

Quiz

정답 Q 9 (단일선택, 10점)

다음 문구의 진위 여부를 판별하세요.

"Seq2Seq 모델은 sequence의 마지막까지 보고 처리하여야 하는 챗봇에 사용하기 적합한 모델로, Encoder와 Decoder의 구조로 되어있다."

☒ True

☐ False

1 2 3 4 5 6 7 8 9 10

Seq2Seq 에 대한 설명입니다. 챗봇 뿐만 아니라 번역기 등 다양한 분야에서 사용되고 있습니다.

Quiz 10 해설

Quiz

정답 Q 10 (단일선택, 10점)

RNN의 변형 중 LSTM보다 조금 단순한 형태이면서 널리 사용되는 모델의 이름은 무엇인가요?

- ☐ CNN(Convolutional Neural Network)
- ☒ GRU(Gated Recurrent Unit)
- ☐ DDPG(Deep Deterministic Policy Gradient)
- ☐ SVM(Support Vector Machine)

1 2 3 4 5 6 7 8 9 10

보기 중 LSTM을 보다 단순화한 RNN의 변형모델은 GRU(Gated Recurrent Unit) 입니다.

이제 프로젝트를 진행하세요!