

Deep Learning Term Project 2

Report - 2017013436 남예준

Problem



처음 코드를 받고 pack_padded_sequence를 활용할 계획을 가졌습니다. 하지만 그 과정에서 train의 accuracy는 늘었지만 test의 accuracy가 늘어나지 않음을 확인했습니다. 그 과정에서 과적합이 발생한다고 생각하게 되었습니다.

Approach

model.py에 드롭아웃을 추가해주고 test의 accuracy를 측정해보았지만 큰 변화를 얻을 수 없었고 더이상의 해결책을 찾을 수 없던 저는 조교님께 자문을 구하여 vocab이 불일치한다는 솔루션을 얻을 수 있었습니다.

그 부분을 해결하게 됨으로써 test의 accuracy가 대략 75%정도로 늘어나게 되었습니다.

Improve

문제를 해결한 후 저는 test의 accuracy를 늘여줄 방법에 대해서 고민을 했습니다. 가장 먼저 실행한 일은 RNN을 LSTM으로 바꿔준 것이었습니다. 검색을 통해 LSTM에서는 pack_padded_sequence를 사용할 때 pad_packed_sequence와 같이 사용함을 여러 예제를 통해서 확인하였고 마찬가지로 모델에 추가해주었습니다. 간단히 말해 pack_padded_sequence의 역연산임을 확인했습니다.

https://pytorch.org/docs/stable/generated/torch.nn.utils.rnn.pad_packed_sequence.html

다음으로 저는 hyper parameter를 조정해가며 결과를 확인하기로 했습니다. 제가 조정하며 확인한 변수는 batch size와 vocab size였습니다.

첫 번째 시도(batch size = 64, vocab size = 30000)

```
(base) PS C:\Users\Wyejun\Deep_Learning\NN\project\NN_project> python train.py
device : cuda
[Epoch 1 / 5]
100% | 2346/2346 [00:19:00:00, 120.89it/s]
Epoch 1
train_loss : 1.142852920807028
train_accuracy : 46.398
train_loss decreased (inf -> 1.142853). Saving model ...
[Epoch 2 / 5]
100% | 2346/2346 [00:18:00:00, 127.87it/s]
Epoch 2
train_loss : 1.0122728974567146
train_accuracy : 63.416
train_loss decreased (1.142853 -> 1.012273). Saving model ...
[Epoch 3 / 5]
100% | 2346/2346 [00:18:00:00, 127.05it/s]
Epoch 3
train_loss : 0.9581719959314708
train_accuracy : 74.570
train_loss decreased (1.012273 -> 0.958172). Saving model ...
[Epoch 4 / 5]
100% | 2346/2346 [00:18:00:00, 128.36it/s]
Epoch 4
train_loss : 0.9253268132632548
train_accuracy : 80.208
train_loss decreased (0.958172 -> 0.925327). Saving model ...
[Epoch 5 / 5]
100% | 2346/2346 [00:18:00:00, 127.57it/s]
Epoch 5
train_loss : 0.907849830839414
train_accuracy : 82.191
train_loss decreased (0.925327 -> 0.907850). Saving model ...
```

```
(base) PS C:\Users\Wyejun\Deep_Learning\NN\project\NN_project> python test.py
{0: 'education', 1: 'ict', 2: 'mechanics', 3: 'economy'}
100% | 2/2 [00:00:00:00, 2.74it/s]
Test Accuracy : 0.86000
```

test acc = 86%

두 번째 시도(batch size = 32, vocab size = 30000)

```
(base) PS C:\Users\wejun\Deep_Learning\WNN_project\WNN_project> python test.py
{0: 'education', 1: 'ict', 2: 'mechanics', 3: 'economy'}
100% | 4/4 [00:00<00:00, 5.56it/s]
Test Accuracy : 0.89000
```

세 번째 시도(batch size = 64, vocab size = 50000)

```
(base) PS C:\Users\weijun\Deep_Learning\WRNN_project\WRNN_project> python test.py
{0: 'education', 1: 'ict', 2: 'mechanics', 3: 'economy'}
100% | 2/2 [00:00<00:00, 2.77it/s]
Test Accuracy : 0.90000
```

마지막 시도(batch size = 32, vocab size = 50000)

```

(base) PS C:\Users\#yejun#Deep_Learning#RNN_project#RNN_project> python test.py
{0: 'education', 1: 'ict', 2: 'mechanics', 3: 'economy'}
100% | 4/4 [00:00<00:00, 5.59it/s]
Test Accuracy : 0.91000

```

test acc = 91%

	batch size = 64	batch size = 32
vocab size = 30000	86%	89%
vocab size = 50000	90%	91%

다음과 같은 결과를 얻게 되었습니다.

Evaluation

batch size가 영향을 미치는 것은 강의자료를 통해 어느정도 알고 있었지만 vocab size를 키워주는 것도 영향을 미치는 것을 확인할 수 있었습니다. 결론적으로 batch size = 32, vocab size = 50000으로 설정하게 되었습니다. 추가로 embedding_dim, hidden_dim등을 추가로 조정해보려 했지만 개선된 결과를 얻지 못해 프로젝트를 종료하기로 했습니다.