소설 작가 분류 AI 경진대회 <월간 데이콘 9>

NLP 분반 : AI야 도와 조

김도윤, 조민제, 허채은

목차

1. 과제 선정 배경
2. 실험 모델 종류 및 결과
3. 결론 및 보완할 점

1. 과제 선정 배경

1. 과제 선정 배경

- 적정 수준의 공모전 과제 수행
  - 학습 내용 실적용
  - 결과물 제출 - > 객관적 평가
- 모델 직접 설계 및 구현 , 코드 작성 연습
  - Layer 추가, Hyper Parameter Tuning 패키지 활용 (ex: ray)

2. 실험 모델 종류 및 결과

2. 실험 모델 종류 및 결과

Keras : 민제, 채은

- LSTM : Double + Bi Directional
- Transformer : Encoder
- Simple Neural Network

* Loss Function

: Categorical Cross Entropy

PyTorch : 도윤

- CNN
- LSTM : Double, Bi Directional, Double + Bi Directional
- GRU : Double + Bi Directional
- Transformer : Encoder

* Loss Function : NLL Loss

** Cross Entropy = Log Softmax + Negative Log Likelihood*

2. 실험 모델 종류 및 결과

< Keras >

** Optimizer : Adam*

Type	Structure	Valid Loss	Test Loss
Double Bi LSTM	Embedding Layer + (Bi LSTM + Bi LSTM) + Dense Layer + Softmax	0.7011	0.6074
Transformer	Token/Position Embedding +TransformerBlock + AvgPool + Dropout + Dense Layer + ReLU + Dropout + Softmax	0.15806	0.6400
Neural Network	Embedding Layer + AvgPool + Softmax	0.6893	**0.4179**

2. 실험 모델 종류 및 결과

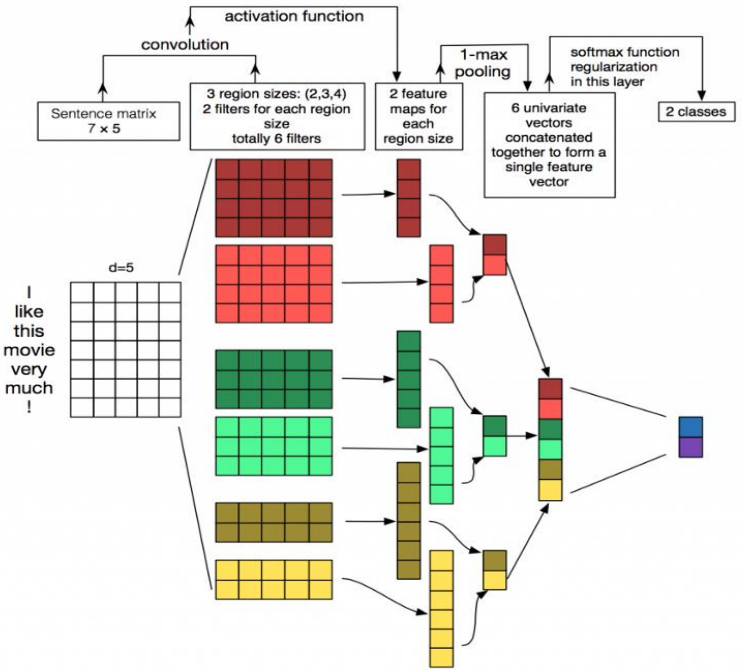
< PyTorch >

** Optimizer : SGD + Momentum*

Type	Structure	Valid Loss	Test Loss
Double LSTM	Embedding Layer + (LSTM + LSTM) + Dense Layer + Log Softmax	0.9287	-
Bi LSTM	Embedding Layer + Bi LSTM + Dense Layer + Log Softmax	0.8838	-
Double Bi LSTM	Embedding Layer + (Bi LSTM + Bi LSTM) + Dense Layer + Log Softmax	0.8120	8.0782
Double Bi GRU	Embedding Layer + (Bi GRU +Bi GRU) + Dense Layer + Dense Layer+ Log Softmax	0.8015	7.5524

2. 실험 모델 종류 및 결과

Type	Private Score	Public Score
CNN	0.9967	-



2. 실험 모델 종류 및 결과

Name	Best Loss
20210309_DoubleBiGRU_seqlen_106_600_200.pt	0.8015
20210307_DoubleBiLSTM_seqlen_106_500_200_sum.pt	0.8120
DoubleBiLSTM_500_150_sum.pt	0.8253
20210302_DoubleBiLSTM_seqlen_106_200_256_sum.pt	0.8260
20210307_DoubleBiLSTM_seqlen_106_500_200_sum.pt	0.8286
20210302_DoubleBiLSTM_seqlen_106_250_128_sum.pt	0.8305
20210302_DoubleBiLSTM_seqlen_106_250_256_sum.pt	0.8317
20210302_DoubleBiLSTM_seqlen_106_200_512_sum.pt	0.8359
20210227_DoubleBiLSTM_600_100_sum.pt	0.8361
20210227_DoubleBiLSTM_500_150_mean.pt	0.8379
20210302_DoubleBiLSTM_seqlen_106_200_128_sum.pt	0.8411
20210301_DoubleBiLSTM_seqlen_106_400_300_sum.pt	0.8514

Name	Best Loss
DoubleBiLSTM_500_150_mean.pt	0.8543
DoubleBiLSTM_500_150_mean.pt	0.8543
20210302_DoubleBiLSTM_seqlen_106_200_64_sum.pt	0.8577
20210302_DoubleBiLSTM_seqlen_106_250_64_sum.pt	0.8623
20210307_DoubleBiLSTM_seqlen_154_450_200_sum.pt	0.8678
20210308_DoubleBiLSTM_seqlen_106_500_250_mean_...	0.8738
20210227_DoubleBiLSTM_600_100_mean_classweight.pt	0.8769
20210227_DoubleBiLSTM_600_100_sum_classweight.pt	0.8796
20210227_DoubleBiLSTM_500_150_mean_classweight.pt	0.8796
20210227_DoubleBiLSTM_450_200_sum.pt	0.8856
20210301_DoubleBiLSTM_seqlen_106_200_200_sum.pt	0.9080

2. 실험 모델 종류 및 결과

```
def main(num_samples=10, max_num_epochs=20, gpus_per_trial=1):
    !pip install tensorboardX
    import tensorboardX

    config = {
            'tok': tune.choice(['spacy','toktok']),
            'percentile':tune.choice([90,95,99]),
            'input_size':tune.sample_from(lambda _ : 50*np.random.randint(1,10)),
            'hidden_dim':tune.sample_from(lambda _ : 10*np.random.randint(1,20)),
            'dropout': tune.loguniform(1e-1, 7e-1),
            'weight' : tune.choice([True,False]),
            'lr': tune.sample_from(lambda _ : 10**(-np.random.randint(1,4))),
            'option' : tune.choice(['sum','mean'])
                    }
    scheduler = ASHAScheduler(
        metric="loss",
        mode="min",
        max_t=max_num_epochs,
        grace_period=1,
        reduction_factor=2)
    reporter = CLIReporter(
        #parameter_columns=["tok", "percentile", "input_size", "hidden_dim","dropout",'weight','lr','option'],
        metric_columns=["loss", "accuracy", "training_iteration"])
    result = tune.run(
        partial(Train_DoubleBiGRU),
        resources_per_trial={"cpu": 2, "gpu": gpus_per_trial},
        config=config,
        num_samples=num_samples,
        scheduler=scheduler,
        progress_reporter=reporter)
```

```
best_trial = result.get_best_trial("loss", "min", "last")
print("Best trial config: {}".format(best_trial.config))
print("Best trial final validation loss: {}".format(
    best_trial.last_result["loss"]))
print("Best trial final validation accuracy: {}".format(
    best_trial.last_result["accuracy"]))
```

3. 결론 및 보완할 점

3. 결론 및 보완할 점

- 딥러닝 모델이 무조건 짱? NO

: 코퍼스의 종류, 단어의 수, Sequence 길이 등 적절히 고려 필요

: 다양한 분류기 추가 사용 가능 : Boost, Ensemble 계열

- Hyper Parameter Tuning의 경향 분석

: 토크나이져 종류, 단어의 수, Max Sequence Len, Embedding Dim

- Validation Loss 와 Test Loss의 큰 차이 발생 원인 파악
- 구체적인 전처리 방향 설정

Thank you

소설 작가 분류 AI 경진대회
<월간 데이콘 9>

NLP 분반
김도윤, 조민제, 허채은