홍여원

정적 GNN – lightweight

저번보다 input-output 명확

F1\_score는 저번보다 높아 보이지만, 무조건 1로 예측했다고 봐도 무방

Height, weight, reach, stance 4가지 feature만 사용

8:1:1 -> 0.7188

6:2:2 -> 0.7397

여기까진 계속 1로 예측한다…

Threshold 다시 구해서 학습하니 무조건 1로 예측하진 않음.

하지만 0.49로 성능 하락

Baseline 평가 다시해야할 듯.. [민규]

경기 중에 나오는 4가지 feature만 다시 사용

모든 정보로 평가

정확도 뽑아보기

Epoch 50 (val 비율 가장 낮게 했음)

8:1:1, 6:2:2 (가능하면 6:3:1도?)

모든 체급에 대해! (경기 수 10 미만 제외)

시각화

Red에 대해서만 해도 되나?

Red/blue 모두 해 보고, 합쳐서도 해 보겠다. [준성]

데이터셋의 red/blue가 실제 red/blue와 다른 경우 있는데 반영된 건가?

-> event\_dataset\_2에 반영됨

Model2.py

경기를 체급별로 분류

GNN 모델 생성 – 한 체급 내의 모든 선수들을 노드로 생성

각 경기를 edge로 생성

Label에 red 승리 – 1, blue 승리 – 0 저장. label을 가지고 학습

Train/val/test – 6:2:2. 가장 결과 잘 나오는 비율로 가야할 듯

10 epoch마다 저장 - Validation loss 가장 작은 epoch를 뽑아줌

--evaluate [모델이름] 커맨드에 입력하면 평가 진행됨

=> 이걸로 cross validation 해야 한다!

=> hidden\_channels, dropout, learning rate, train test 비율 최적의 값 실험으로 찾기

[원렬]

TemporalGNN 구현해보겠다 [여원]

evolveGCN이나 다른 모델 구현해보기 [일규]

화요일에는 현 모델로 cross-validation 통해 발표

이후에는 다른 접근 완료되면 해당 결과로 발표

조민규

내가 준비한 내용…

Github 참고