

오버워치 리그 승률 예측 및 게임 전략 지표 개발

1. 오버워치 게임설명
2. 데이터 소개
3. 문제정의 및 기대효과
4. 데이터 전처리 계획
5. 프로젝트 목표

오버워치 게임 소개

☀ 게임 목표:

- 각 전장에서 목표 달성하기



《오버워치》(Overwatch)는 블리자드 엔터테인먼트가 개발 및 배급한 팀 기반의 1인칭 슈팅 게임 (FPS게임)

오버워치 게임 소개_팀 구성



6:6으로 싸우는 게임

- 6명이 한 팀
- 돌격포지션: 2명
- 공격포지션: 2명
- 지원포지션: 2명

목차 [순기기]	
1	골키퍼 (GK)
2	수비수 (DF)
2.1	센터백 (CB)
2.2	스위퍼/리베로 (SW)
2.3	풀백 (FB/LB/RB)
2.4	윙백 (WB/LWB/RWB)
3	미드필더 (MF)
3.1	중앙 미드필더 (CM)
3.2	수비형 미드필더 (DM)
3.3	공격형 미드필더 (AM)
3.4	미드라이커 (AM)
3.5	윙어 (LW/LM/RW/RM)
4	공격수 (FW)
4.1	중앙 공격수 (CF)
4.2	세컨드 스트라이커 (SS)
4.3	윙어 (LW/RW)
4.4	폴스나인 (False9)
5	같이 보기
6	각주

(축구에서 공격수, 수비수, 골키퍼가 있는 것처럼 크게 맡은 역할이 있다.

각 캐릭터마다 스킬이 달라 세부 역할이 나뉘고 그걸 다루는 플레이어의 역량에 따라 다양한 전략을 펼칠 수 있다.)

돌격								
	D.Va	라만하르트	레깅볼	로드호그	시그마	오리사	윈스턴	자리아
공격								
	겐지	돔피스트	리퍼	맥크리	메이	바스티온	솔저: 76	솜브라
								
	시메트라	애쉬	에코	위도우메이커	정크렛	토르비온	트레이서	파라
지원								
	한조							
+								
	루시우	메르시	모이라	바티스트	브리기테	아나	젠야타	

오버워치 게임 소개_전장 종류

• 쟁탈(Control)

- 1.1. 네팔(Nepal) - 네팔
- 1.2. 리장 타워(Lijiang Tower) - 중국(구이린)
- 1.3. 부산(Busan) - 대한민국(부산)
- 1.4. 오아시스(Oasis) - 이라크
- 1.5. 일리오스(Ilios) - 그리스(산토리니)

• 점령(Assault)

- 2.1. 볼스카야 인더스트리(Volskaya Industries) - 러시아(모스크바)
- 2.2. 아누비스 신전(Temple of Anubis) - 이집트(기자)
- 2.3. 파리(Paris) - 프랑스(파리)
- 2.4. 한amura(Hanamura) - 일본(도쿄)
- 2.5. 호라이즌 달 기지(Horizon Lunar Colony) - 달

• 화물호위(payload)

- 3.1. 66번 국도(Route 66) - 미국
- 3.2. 감시 기지: 지브롤터(Watchpoint: Gibraltar) - 지브롤터
- 3.3. 도라도(Dorado) - 멕시코(베라크루스)
- 3.4. 리알토(Rialto) - 이탈리아(베네치아)
- 3.5. 쓰레기촌(Junkertown) - 호주
- 3.6. 하바나(Havana) - 쿠바(아바나)

• 혼합(Hybrid)

- 4.1. 눔바니(Numbani) - 서아프리카
- 4.2. 블리자드 월드(Blizzard World) - 미국(어바인)
- 4.3. 아이헨발데(Eichenwalde) - 독일(슈투트가르트)
- 4.4. 왕의 길(King's Row) - 영국(런던)
- 4.5. 할리우드(Hollywood) - 미국(로스앤젤레스)



<쟁탈 목표> -> 공격, 수비가 따로 X

- 라운드 승리 목표: 거점 점령 게이지 100% 도달
- 승리 목표: 라운드 2번 승리

<점령 목표>

- 공격팀 목표: 제한 시간 안에 거점 2개 모두 점령
- 수비팀 목표: 제한 시간 동안 공격팀의 거점 점령 저지

<화물호위 목표>

- 공격팀 목표: 제한 시간 안에 화물을 목적지까지 운송
- 수비팀 목표: 제한 시간 동안 공격팀의 화물 운송 저지

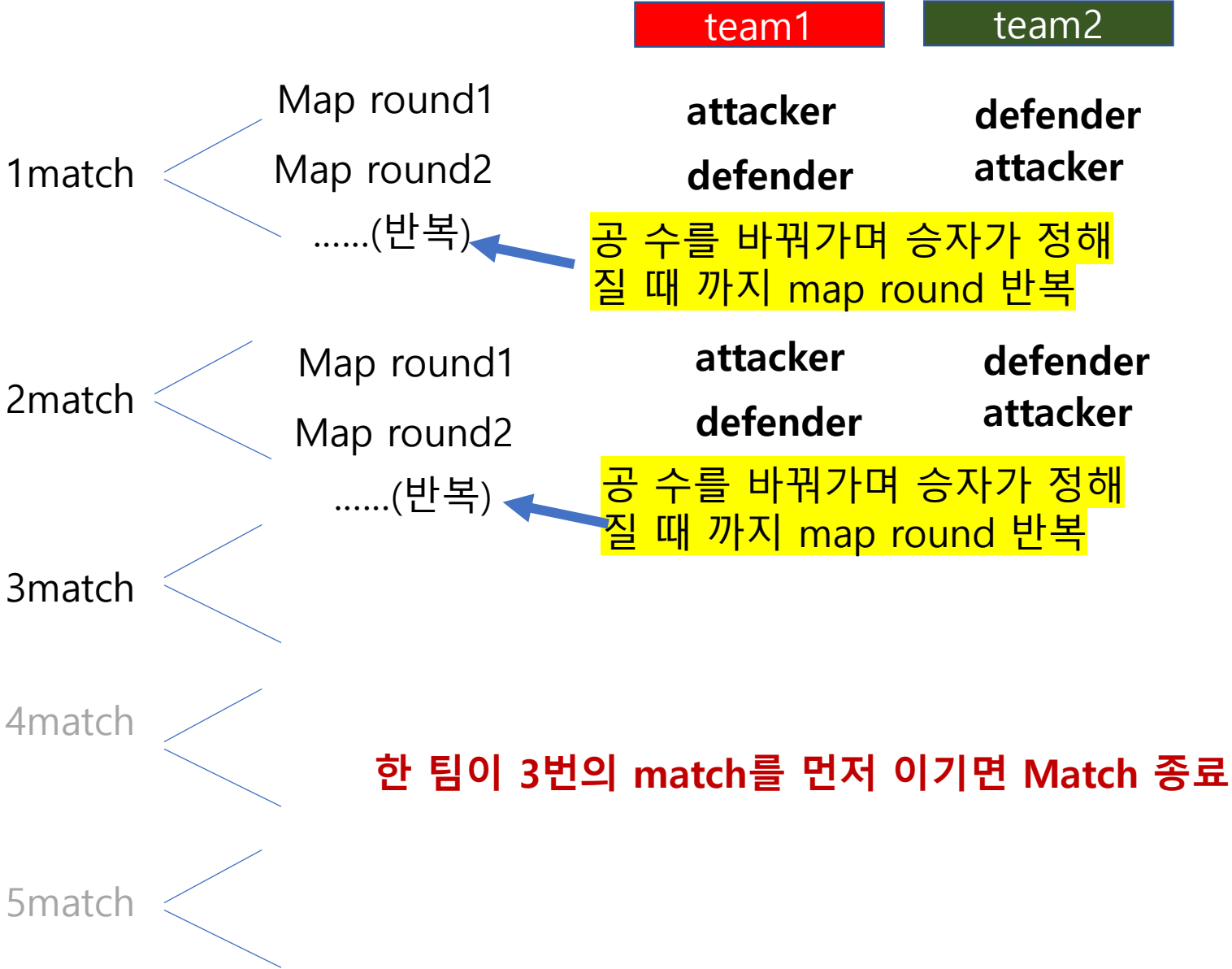
<혼합 목표> = 점령 + 화물호위

- 공격팀 목표: 제한 시간 안에 거점을 점령후 화물을 목적지까지 운송
- 수비팀 목표: 제한 시간동안 공격팀의 거점 점령 & 화물 운송 저지

오버워치 게임 소개

게임 규칙

5판 3선승: 5개의 match
중
3개의 match에서
먼저 이긴 팀이 승리한다.



오버워치 게임 데이터분석 현황

- IBM이 오버워치와 파트너십을 맺음.

- 오버워치 리그 순위 시스템 개선을 위해 IBM은 자연어 처리·머신러닝 전문가들을 모아 2021년부터 관련 기술 개발을 시작할 예정.

- 왓슨AI를 통해 실시간 방송 및 방송 내 예측 분석 솔루션을 제공 예정.

IBM에서 오버워치 리그와 파트너십을 맺고 데이터 분석을 할 정도로 오버워치 게임 내에서 데이터 분석에 대한 수요는 많은 것으로 보인다. 또한, 일반인들이 참여할 수 있는 E-sport 경기가 많아지면서 앞으로 게임 리그 데이터 분석에 대한 수요가 많아질 것으로 보인다.

이제 분석되기 시작해서 아직 오버워치 데이터분석에 관한 논문은 없는 것으로 보임 (대부분 롤(게임)에 관한 논문)

IBM, 오버워치 리그에 AI 기술 지원...e스포츠 첫 진출

박성은 기자 | 입력 2020.10.12 17:55 | 수정 2020.10.13 13:58 | 댓글 0 | 좋아요 0



액티비전 블리자드에 AI·ML·클라우드 기술 지원 협약
금년부터 2022년까지 오버워치 리그 진행 개선
리그 랭킹 시스템·방송 예측 분석 개선하는 솔루션 개발



(사진=셔터스톡)2019년 러시아 e스포츠 카운터 스트라이크 현장

IBM의 AI·머신러닝·클라우드 기술이 인기 온라인게임 오버워치 리그에 사용된다. 벤처비트는 8일 IBM이 액티비전 블리자드와 오버워치 리그 진행을 위한 기술 협약을 맺었다고 보도했다.

문제 정의

내용

오버워치 리그 데이터를 분석하여 승률에 영향을 끼치는 요소에 대한 insight를 발굴하여 실시간 또는 팀별 승률을 예측하고, MVP(키 플레이어)를 찾아 팀 전략에 도움을 줄 수 있는 지표 만들기

기대효과

- E-sport 산업이 발달함에 따라 일반인들이 출전하는 대회가 많아짐
- 리그데이터는 프로게이머들의 데이터이므로 전략을 세울 때 표준이 될 수 있음.
- 따라서 팀전략에 도움이 되는 지표를 개발하면 데이터분석가가 없는 대학 간 오버워치 대회 출전팀이나 프로지향팀, 게임 내에 랭크전에 나가는 일반인들로 구성된 팀들에게 솔루션을 제공할 수 있음

match_map_stats.csv

phs연도.csv

1. Map round게임 결과 데이터

2. Match에서 player들의 행동을 측정한
(stat) 데이터

데이터 수집

데이터 수집

<https://overwatchleague.com/ko-kr/statslab-players>

DOWNLOAD THE DATA

Fans are welcome to play with the data. We are also providing a list of appropriate data. These files and dashboards will be updated regularly throughout the 2020 season.

Download: 2018 Player Stats Updated 2/18/2020

Download: 2019 Player Stats Updated 2/18/2020

Download: 2020 Player Stats Updated 10/12/2020

Download: Map Stats Updated 10/12/2020

match_map_stats.csv

2018년부터 2020년까지 오버워치 리그의 게임 자체 데이터.
Map round(한 행에 대한 key) 별 **Map round(게임 결과에 대한 데이터).**

phs연도.csv

Player에 대한 데이터

overwatch match 에서 각각 match id에 대한 데이터에 경기에 참가한 player의 세부적인 데이터(어떤 영웅을 사용했는지, 어떤 기여를 했는지)가 들어있는 데이터.

match_map_stats.csv

```
[ ] df.iloc[rindex,:]
```

	round_start_time	round_end_time	stage	match_id	game_number	match_winner	map_winner	map_loser	map_name	map_round	...	team_one_name	team_two_name	attacker_payload_distance	defender_payload_distance	att
661	2018-02-22 05:13:02	2018-02-22 05:17:13	Overwatch League - Stage 2	10280	3	Los Angeles Gladiators	Los Angeles Gladiators	San Francisco Shock	King's Row	1	—	San Francisco Shock	Los Angeles Gladiators	70.293495	0.000000	
663	2018-02-22 05:33:20	2018-02-22 05:41:32	Overwatch League - Stage 2	10280	3	Los Angeles Gladiators	Los Angeles Gladiators	San Francisco Shock	Watchpoint: Gibraltar	1	—	San Francisco Shock	Los Angeles Gladiators	83.651459	0.000000	
662	2018-02-22 05:19:10	2018-02-22 05:26:04	Overwatch League - Stage 2	10280	3	Los Angeles Gladiators	Los Angeles Gladiators	San Francisco Shock	King's Row	2	—	San Francisco Shock	Los Angeles Gladiators	104.270859	70.293495	
664	2018-02-22 05:43:30	2018-02-22 05:50:00	Overwatch League - Stage 2	10280	3	Los Angeles Gladiators	Los Angeles Gladiators	San Francisco Shock	Watchpoint: Gibraltar	2	—	San Francisco Shock	Los Angeles Gladiators	75.560165	83.651459	
4065	2019-04-27 16:55:38	2019-04-27 16:58:57	Overwatch League Stage 2	21352	1	London Spitfire	London Spitfire	Paris Eternal	Busan	2	—	Paris Eternal	London Spitfire	0.000000	0.000000	
4066	2019-04-27 16:37:19	2019-04-27 16:58:57	Overwatch League Stage 2	21352	1	London Spitfire	London Spitfire	Paris Eternal	Busan	2	—	Paris Eternal	London Spitfire	0.000000	0.000000	
5361	2019-07-28 19:50:17	2019-07-28 19:54:40	Overwatch League Stage 4	21433	3	Shanghai Dragons	Seoul Dynasty	Shanghai Dragons	Hollywood	1	—	Seoul Dynasty	Shanghai Dragons	0.000000	0.000000	
5363	2019-07-28 20:06:32	2019-07-28 20:14:46	Overwatch League Stage 4	21433	3	Shanghai Dragons	Shanghai Dragons	Seoul Dynasty	Junkertown	1	—	Seoul Dynasty	Shanghai Dragons	101.897583	0.000000	

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9107 entries, 0 to 9106
Data columns (total 25 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   round_start_time                      9106 non-null  object
1   round_end_time                        9107 non-null  object
2   stage                                9107 non-null  object
3   match_id                              9107 non-null  int64
4   game_number                           9107 non-null  int64
5   match_winner                          9107 non-null  object
6   map_winner                            9107 non-null  object
7   map_loser                             9107 non-null  object
8   map_name                              9107 non-null  object
9   map_round                             9107 non-null  int64
10  winning_team_final_map_score          9107 non-null  int64
11  losing_team_final_map_score           9107 non-null  int64
12  control_round_name                    2648 non-null  object
13  attacker                              9107 non-null  object
14  defender                              9107 non-null  object
15  team_one_name                         9107 non-null  object
16  team_two_name                         9107 non-null  object
17  attacker_payload_distance              9107 non-null  float64
18  defender_payload_distance              9107 non-null  float64
19  attacker_time_banked                  9107 non-null  float64
20  defender_time_banked                  9107 non-null  float64
21  attacker_control_perecent              2647 non-null  float64
22  defender_control_perecent              2647 non-null  float64
23  attacker_round_end_score              9107 non-null  int64
24  defender_round_end_score               9107 non-null  int64
dtypes: float64(6), int64(7), object(12)
memory usage: 1.7+ MB
```

Map round(게임 결과에 대한 데이터)

Round가 시작한시간,
Round가 끝난 시간,
어떤 매치에서 이루어진 map round인지,
누가이겼는지,
참가한 팀 이름,
mapround에서 누가 attacker였는지,
누가 defender였는지.
Attacker팀이 얼마나 공격을 했는지.
Defender팀이 얼마나 방어했는지.
각 팀의 점수

Phs_연도_1.csv

-기간:
2020년 2/8~7/6

요약 :
플레이어가 한
match에서 어떤 행
동을 했는지에 관한
데이터

	start_time	esports_match_id	tournament_title	map_type	map_name	player_name	team_name	stat_name	hero_name	stat_amount
0	2/8/2020 18:13	30991	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	All Damage Done	All Heroes	6674.164055
1	2/8/2020 18:13	30991	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Assists	All Heroes	11.000000
2	2/8/2020 18:13	30991	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Average Time Alive	All Heroes	60.199502
3	2/8/2020 18:13	30991	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Barrier Damage Done	All Heroes	1875.000000
4	2/8/2020 18:13	30991	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Damage Blocked	All Heroes	14640.911780
5	2/8/2020 18:13	30991	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Damage Done	All Heroes	4798.164055
6	2/8/2020 18:13	30991	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Damage Taken	All Heroes	8267.621440
7	2/8/2020 18:13	30991	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Deaths	All Heroes	5.000000
8	2/8/2020 18:13	30991	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Eliminations	All Heroes	13.000000
9	2/8/2020 18:13	30991	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Final Blows	All Heroes	2.000000

Player에 대한 데이터

start_time: 맵경기 시작 시간

esports_match_id: 매치 id

tournament_title: 어떤년도의 어떤 시즌인지

map_type: 총 4가지 타입이 있

다. (ASSAULT, CONTROL, HYBRID, PAYLOAD) -> 맵 타입에 따
라 경기시간과 round 평균 횟수가 다르다.

map_name: 맵 이름

player_name: 선수닉네임

team_name: 팀이름

stat_name: 스탯이름 (처치, 어시스트, 죽음, 등등 공통적인 스탯
이 있고 각 HERO별로 조금씩 스탯이름이 다르다.)

hero_name: 영웅 이름 (영웅은 총 3타입(탱커,공격,힐러) 이 있
고 모두HERO별로 조금씩 스탯이름이 다르다.)

stat_amount: stat_name에 해당되는 숫자 (stat_name의 값이
damage 데미지를 준 량을 의미하고 stat_name의 값이 death면 죽
은 횟수를 의미)

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 760111 entries, 0 to 760110
Data columns (total 10 columns):
#   Column          Non-Null Count  Dtype
---  -
0   start_time      760111 non-null object
1   esports_match_id 760111 non-null int64
2   tournament_title 760111 non-null object
3   map_type        760111 non-null object
4   map_name        760111 non-null object
5   player_name     760111 non-null object
6   team_name       760111 non-null object
7   stat_name       760111 non-null object
8   hero_name       760111 non-null object
9   stat_amount     760111 non-null float64
dtypes: float64(1), int64(1), object(8)
memory usage: 58.0+ MB
```

Phs_연도_2.csv

-기간:
2020년
7/17~10/10

Feature 및 data 의
특징은 phs_연도_1과
같다.

	start_time	esports_match_id	tournament_title	map_type	map_name	player_name	team_name	stat_name	hero_name	stat_amount
0	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	All Damage Done	All Heroes	6437.383286
1	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Assists	All Heroes	12.000000
2	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Average Time Alive	All Heroes	63.518002
3	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Barrier Damage Done	All Heroes	2300.000000
4	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Damage Blocked	All Heroes	10770.851446
5	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Damage Done	All Heroes	4037.383286
6	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Damage Taken	All Heroes	6481.058250
7	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Deaths	All Heroes	3.000000
8	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Eliminations	All Heroes	17.000000
9	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Environmental Deaths	All Heroes	2.000000

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 463992 entries, 0 to 463991
Data columns (total 10 columns):
#   Column          Non-Null Count  Dtype
---  -
0   start_time      463992 non-null object
1   esports_match_id 463992 non-null int64
2   tournament_title 463992 non-null object
3   map_type        463992 non-null object
4   map_name        463992 non-null object
5   player_name     463992 non-null object
6   team_name       463992 non-null object
7   stat_name       457087 non-null object
8   hero_name       457087 non-null object
9   stat_amount     463992 non-null float64
dtypes: float64(1), int64(1), object(8)
memory usage: 35.4+ MB
```

Player에 대한 데이터

데이터 전처리 계획

match_map_stats.csv

match_map_stats.csv

phs연도.csv

전처리 <https://drive.google.com/file/d/1kVA7EU1XuQNSH-7GJN-PspafmD4bIZSP/view?usp=sharing>

X_train

- diff_att_pdistance : team_one's attacker_payload_distance - team_two's attacker_payload_distance
- diff_def_pdistance : team_one's defender_payload_distance - team_two's defender_payload_distance
- diff_att_time_b : team_one's attacker_time_banked - team_two's attacker_time_banked
- diff_def_time_b : team_one's defender_time_banked - team_two's defender_time_banked
- diff_att_control_p : team_one's attacker_control_perecent - team_two's attacker_control_perecent
- diff_def_control_p : team_one's defender_control_perecent - team_two's defender_control_perecent
- y
- result : team one이 우승하면 1 team two가 우승하면 0

데이터구축 <https://drive.google.com/file/d/1GH0AXSCJ3MjPZQwvPK0FXVSVGEIj45MH/view?usp=sharing>

- Feature : 팀 별 경기에서의 성과 차이
- Target : 승패 예측

4개의 Map type별로 train data 4개 만들기

Phs의 stat_name에서 Map별로 유효한 feature로 사용할 것들을 선택 (value값은 stat_amount에 해당)

- Feature : 한 팀의 6명의 **player들의 행동 stat**(map 별로 유효한 힐러의 stat, 딜러의 stat, 탱커의 stat 6~10개 정도)
- Target : 승패 예측

예측코드 <https://drive.google.com/file/d/1GH0AXSCJ3MjPZQwvPK0FXVSVGEIj45MH/view?usp=sharing>

프로젝트 과제 (중간고사 까지)

challenge 1 - 도메인 지식으로 승패에 영향을 주는 플레이어 행동
찾아서 feature로 만들기

challenge 2 - 기준이 다른 두 데이터 합치기

challenge 3 - traindata 완성 후 기본 알고리즘 적용

프로젝트 목표

Challenge1

도메인 지식으로 승패에 영향을 주는 플레이어 행동 찾아서 featrue로 만들기

Map별로 Phs의 stat_amount에서 feature로 사용할 것들을 선택

	start_time	esports_match_id	tournament_title	map_type	map_name	player_name	team_name	stat_name	hero_name	stat_amount
0	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	All Damage Done	A Heroes	6437.383286
1	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Assists	A Heroes	12.000000
2	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Average Time Alive	A Heroes	63.518002
3	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Barrier Damage Done	A Heroes	2300.000000
4	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Damage Blocked	A Heroes	10770.851446
5	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Damage Done	A Heroes	4037.383286
6	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Damage Taken	A Heroes	6481.058250
7	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Deaths	A Heroes	3.000000
8	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Eliminations	A Heroes	17.000000
9	2020-07-17 19:17:13	34844	OWL 2020 Regular Season	CONTROL	Lijiang Tower	BenBest	Paris Eternal	Environmental Deaths	A Heroes	2.000000

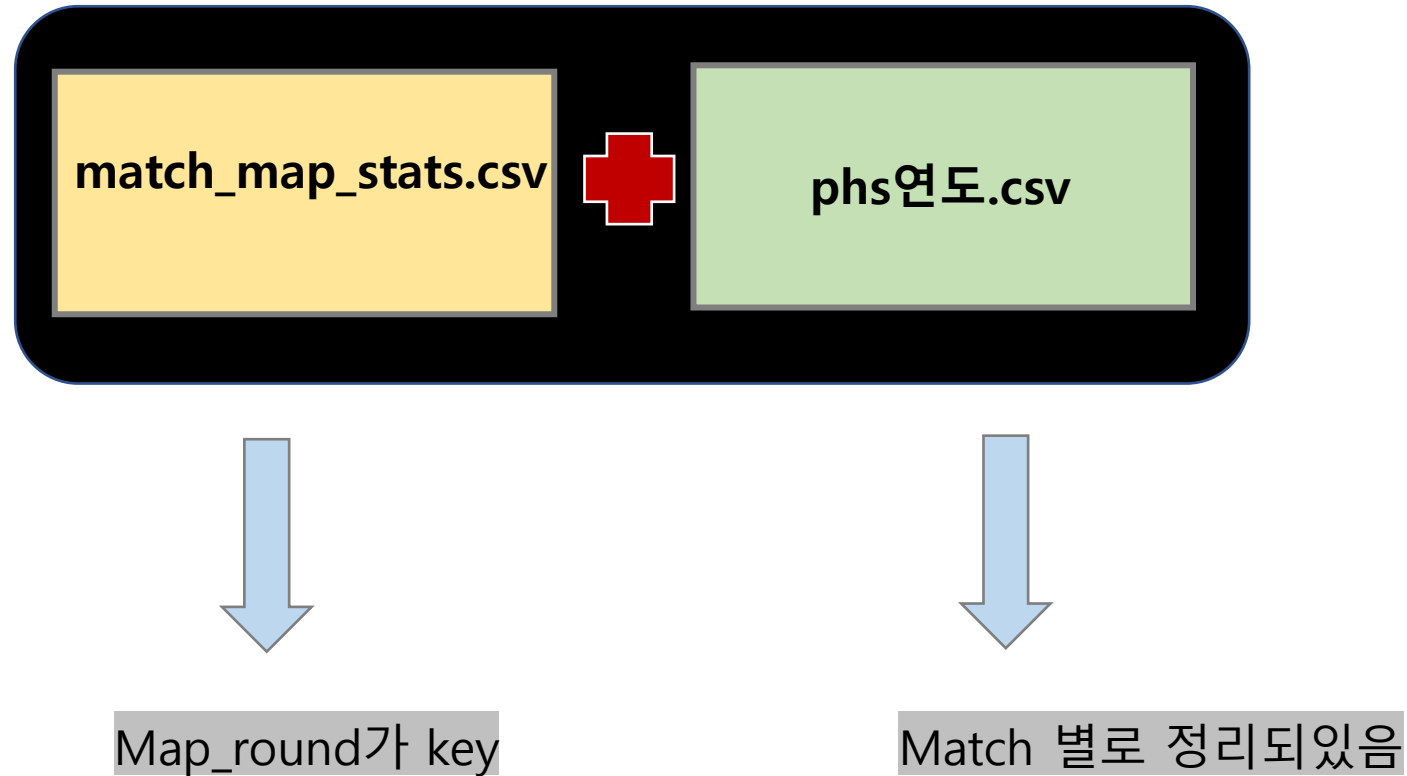
- ① Stat_name, stat_amount는 플레이어의 한 match에서의 행동에 대한 통계이름(도와준 정도, 적을 죽인 양, 살아있던 시간 등) 과 양을 의미함.
- ② 2020년 게임에서만 stat_name이 359개
- ③ 도메인 지식을 활용해 이 중에서 승패를 예측하고 전략을 세우는데 도움이 되는 유효한 변수를 뽑아 train data의 feature로 사용할 예정

```
1 data2020['stat_name'].value_counts().shape  
(359,)
```

Challenge2

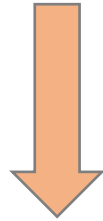
기준이 다른 두 데이터 합치기

두개의 데이터의 기준이 달라서 두데이터를 합치는 전처리에 시간이 소요될 것으로 예상



최종
프로젝트 목표

match_map_stats.csv



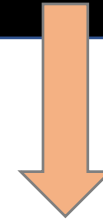
팀 별 경기에서의 성과 차이 -> 승패 예측

실시간 승률 예측 모형

match_map_stats.csv



phs연도.csv



Key player 찾는 알고리즘 개발
(할수있다면)

1. 각 팀원의 전적 데이터를 넣으면
2. 그 팀원이 게임에서 기여 할 것으로 예상되는 stat_amount 예측
3. 팀원들을 예측을 종합하여 전체 팀의 승률 예측

팀에게 게임 전략 및 솔루션 제공