



LEAGUE of LEGENDS

브론즈에게 희망을

LoL api를 이용한 유저 데이터 분석



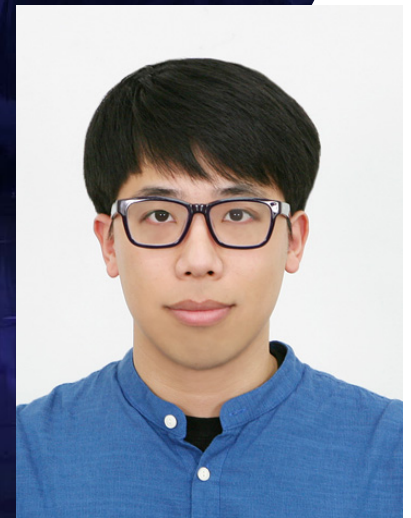
201714167 양현영



201714168 유호원



201714169 조영래



201714170 조해성

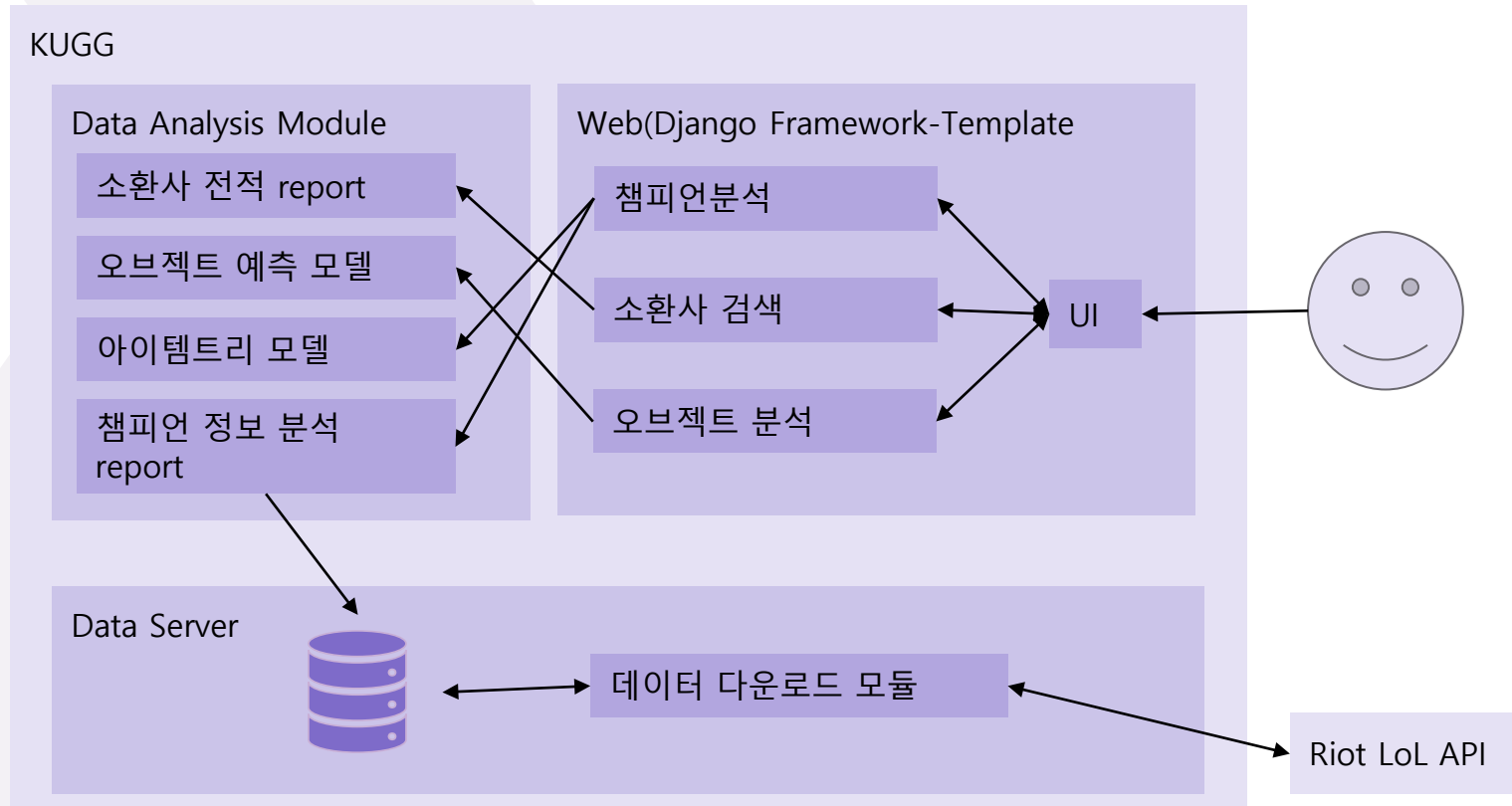
작품 개요

작품 개요

본 작품은 유명 게임인 League of Legends의 전적검색 및 통계, 그리고 승률 예측을 제공해주는 웹사이트를 만드는 것을 목표로 한다.

KU.GG의 아키텍처는 아래와 같으며 크게 데이터수집 모듈, 챔피언 분석 모듈, 오브젝트 분석 모듈로 구성된다.

KU.GG는 데이터 분석을 통해 만들어진 모델을 이용해 유저에게 최대한 정확도 높은 정보 제공을 목표로 한다.



데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix



데이터 수집 모듈

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix



데이터 수집 모듈은 서버에서 24시간 동작하며 데이터를 수집하는 것을 가정하고 만들었다. 데이터 수집 모듈은 Riot API를 통해 raw 데이터를 DB에 저장하는 역할을 한다.

해당 league에 속하는 유저들의 데이터 수집 예시

LEAGUE-V4

```
GET /lol/league/v4/challengerleagues/by-queue/{queue}
GET /lol/league/v4/entries/by-summoner/{encryptedSummonerId}
GET /lol/league/v4/entries/{queue}/{tier}/{division}
GET /lol/league/v4/grandmasterleagues/by-queue/{queue}
GET /lol/league/v4/leagues/{leagueId}
GET /lol/league/v4/masterleagues/by-queue/{queue}
```

```
{
  "tier": "CHALLENGER",
  "leagueId": "XXXXXXXXXX",
  "queue": "RANKED_SOLO_5x5",
  "name": "Hecarim's Snipers",
  "entries": [
    {
      "summonerId": "XXXXXXXXXX",
      "summonerName": "Mango Fish",
      "leaguePoints": 1078,
      "rank": "I",
      "wins": 568,
      "losses": 450,
      "veteran": true,
      "inactive": false,
      "freshBlood": false,
      "hotStreak": true
    }
  ]
}
```

유저의 id로 수집할 수 있는 유저의 상세정보 예시

SUMMONER-V4

```
GET /lol/summoner/v4/summoners/by-account/{encryptedAccountId}
GET /lol/summoner/v4/summoners/by-name/{summonerName}
GET /lol/summoner/v4/summoners/by-puuid/{encryptedPUUID}
GET /lol/summoner/v4/summoners/{encryptedSummonerId}
```

```
{
  "id": "XXXXXXXXXX",
  "accountId": "XXXXXXXXXX",
  "puuid": "XXXXXXXXXX",
  "name": "Hide on bush",
  "profileIconId": 6,
  "revisionDate": 1603953347000,
  "summonerLevel": 357
}
```

유저의 accountId로 수집할 수 있는 유저의 전적과 각 게임별 상세정보 예시

MATCH-V4

```
GET /lol/match/v4/matches/{matchId}
GET /lol/match/v4/matchlists/by-account/{encryptedAccountId}
GET /lol/match/v4/timelines/by-match/{matchId}
GET /lol/match/v4/matches/by-tournament-code/{tournamentCode}/ids
GET /lol/match/v4/matches/{matchId}/by-tournament-code/{tournamentCode}
```

```
{
  "matches": [
    {
      "platformId": "KR",
      "gameId": "XXXXXXXXXX",
      "champion": 64,
      "queue": 420,
      "season": 13,
      "timestamp": 1603971983169,
      "role": "NONE",
      "lane": "JUNGLE"
    }
  ],
  "teams": [
    {
      "teamId": 100,
      "win": "Win",
      "firstBlood": true,
      "firstTower": true,
      "firstInhibitor": true,
      "firstBaron": true,
      "firstDragon": false,
      "firstRiftHerald": false,
      "towerKills": 7,
      "inhibitorKills": 1,
      "baronKills": 1,
      "dragonKills": 1,
      "vilemawKills": 0,
      "riftHeraldKills": 1,
      "dominionVictoryScore": 0,
      "bans": [
        {
          "championId": "XXXXXXXXXX"
        }
      ]
    }
  ],
  "frames": [
    {
      "participantFrames": {
        "1": {
          "participantId": 1,
          "position": {
            "x": 560,
            "y": 581
          },
          "currentGold": 500,
          "totalGold": 500,
          "level": 1,
          "xp": 0,
          "minionsKilled": 0,
          "jungleMinionsKilled": 0,
          "dominionScore": 0,
          "teamScore": 0
        }
      }
    }
  ]
}
```

데이터 수집 모듈

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix

usersLeague Data
(LEAGUE-V4)

Challenger, grandmaster, master에 속한 유저들의
summonerId 및 정보 수집

usersInfo Data
(SUMMONER-V4)

이 전에 수집한 summonerId를 이용해 전적검색을
위한 유저의 상세정보 및 accountId를 수집

usersMatchlist Data
(MATCH-V4)

이 전에 수집한 accountId를 이용해 유저가
플레이한 각 게임들의 matchId 및 전적들의
정보를 수집

matchTimeline Data
(MATCH-V4)

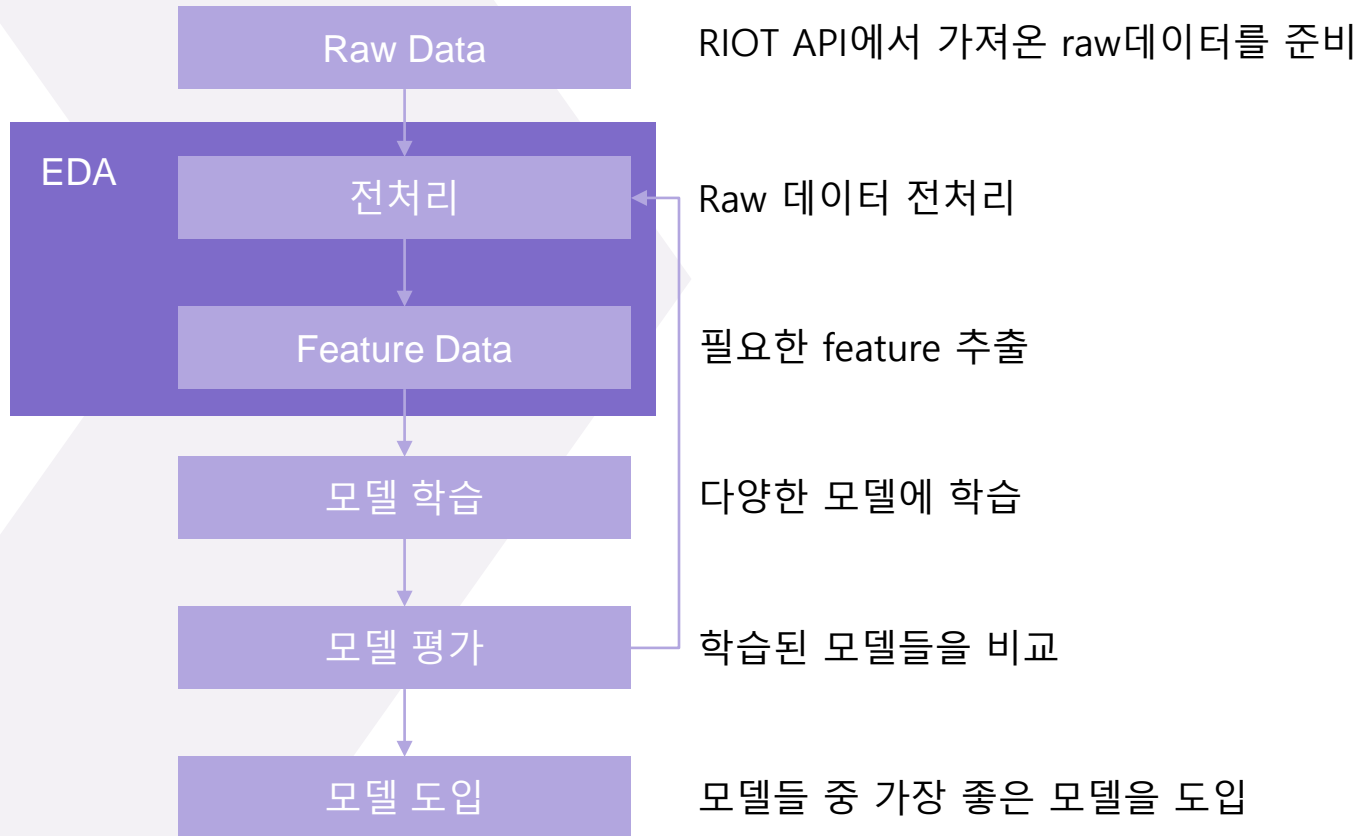
matchInfo Data
(MATCH-V4)

이전에 수집한 matchId를
이용해 각 게임들의 게임
내부 상세 정보들을 수집



데이터 분석과정

챔피언 분석 및 오브젝트 분석에 사용될 머신러닝 모델을 개발하고 웹 사이트에 도입하는 과정은 다음과 같다.



작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

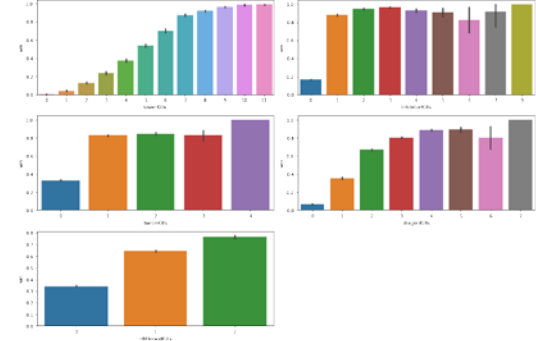
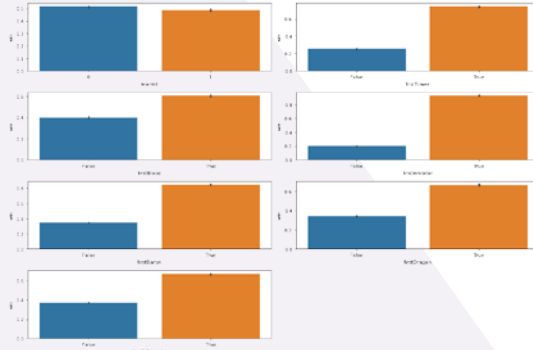
System
Test

Traceability
Matrix

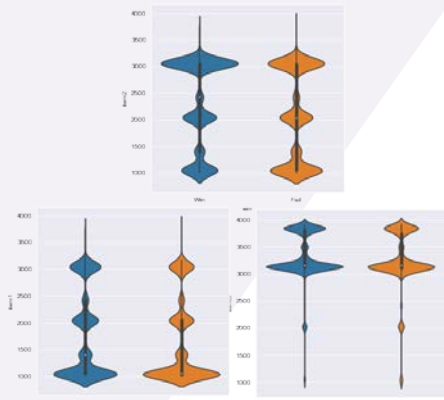


데이터 분석과정(EDA)

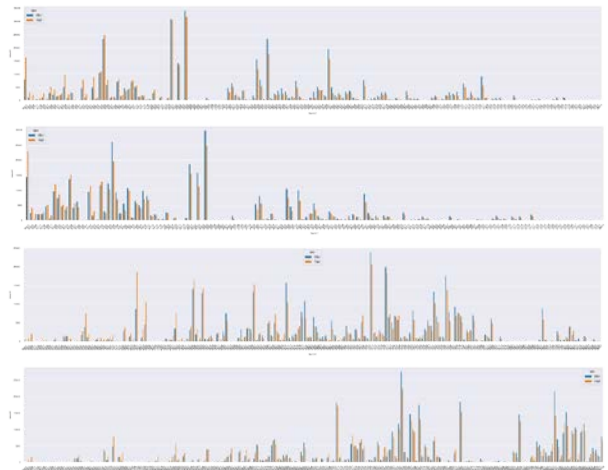
EDA를 통해 데이터의 분포 및 값을 확인해 데이터가 표현하는 다양한 패턴을 발견한다. 또한 raw 데이터에서 이상치 및 null 값들을 찾아내어 정정해준다.



오브젝트 feature들과 승리와 연관관계



승패별 아이템 분포도



아이템 별 승리 횟수

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix



데이터 분석과정(EDA)

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix

데이터 분석 과정을 통해 챔피언 승률, 아이템 트리 픽률등 유저에게 도움이 될수있는 여러 일반 통계 자료도 제공해준다

Teemo의 라인별 승률

position	total matches	win rate
DUO_SUPPORT	22	63.636364
MIDDLE	16	62.500000
TOP	128	48.437500
JUNGLE	15	46.666667

MIDDLE - Ahri의 카운터 챔피언:

	champion	total matches	win rate
141	Yasuo	10	80.000000
159	Katarina	13	76.923077
215	Vladimir	7	71.428571
261	LeBlanc	10	70.000000
350	Talon	9	66.666667

다리우스 아이템트리 픽률

	count	item0name	item1name	item2name	item3name	item4name	item5name
0	6	Ruby Crystal	Doran's Shield	Refillable Potion	Ninja Tabi	Jaurim's Fist	Trinity Force
1	5	Cloth Armor	Dagger	Dagger	Doran's Blade	Phage	Ninja Tabi
2	5	Ruby Crystal	Long Sword	Doran's Shield	Refillable Potion	Ninja Tabi	Trinity Force
3	5	Long Sword	Doran's Shield	Health Potion	Control Ward	Trinity Force	Mercury's Treads
4	5	Ruby Crystal	Long Sword	Doran's Blade	Ninja Tabi	Jaurim's Fist	Trinity Force
5	5	Doran's Shield	Refillable Potion	Control Ward	Jaurim's Fist	Trinity Force	Mercury's Treads
6	5	Boots of Speed	Ruby Crystal	Long Sword	Doran's Blade	Health Potion	Kindlegem
7	5	Ruby Crystal	Doran's Shield	Health Potion	Jaurim's Fist	Trinity Force	Mercury's Treads
8	5	Dagger	Doran's Shield	Doran's Blade	Phage	Ninja Tabi	Sheen
9	5	Sapphire Crystal	Doran's Shield	Health Potion	Phage	Stinger	Mercury's Treads



오브젝트 분석

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

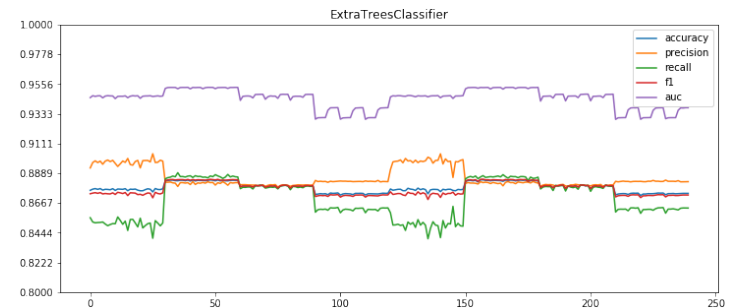
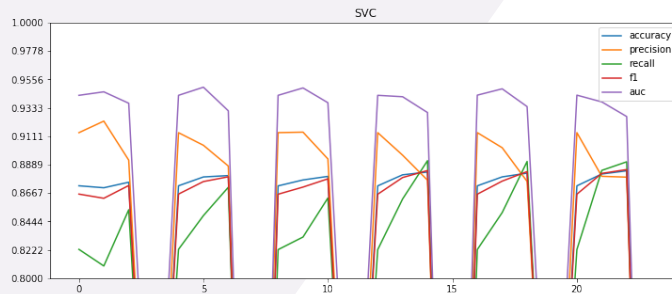
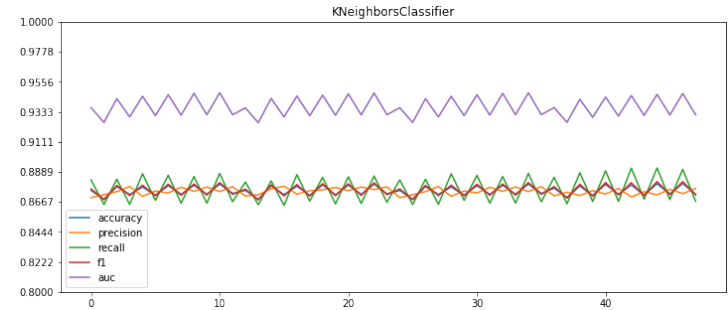
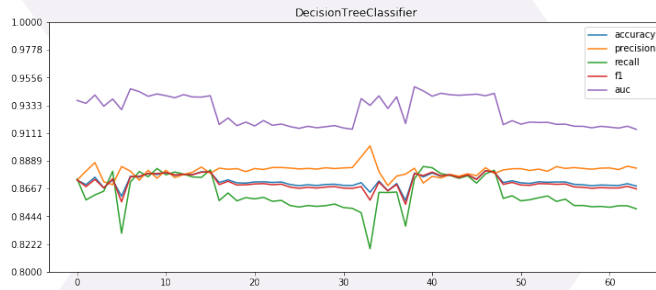
KU.GG
WEB

System
Test

Traceability
Matrix

오브젝트는 League of Legends 내의 다양한 타워, 몬스터 등을 의미함. 각 팀마다 취한 오브젝트들의 개수 또는 첫 오브젝트 여부에 따라 달라지는 승리 확률을 모델에 학습시키고 예측가능하게 하는 '오브젝트분석 시뮬레이션' 을 제작한다.

모델은 DecisionTree, KNN, Random Forestm Stochastic Gradient Descent, SVM 등의 모델을 사용해 학습시킨다.



모델은 모델에서 이용할 수 있는 다양한 하이퍼파라미터를 바꾸면서 학습시킨다. 그리고 accuracy, recall, precision, f1 등 다양한 모델 평가 요소를 활용하고 이 값들의 추이를 그래프로 표현해 관찰한다.



오브젝트 분석

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix

학습이 완료된 모델들은 모델의 평가요소들을 한눈에 볼 수 있게 데이터프레임으로 표현한다. 그리고 원하는 평가 요소를 오름차순 또는 내림차순으로 정렬해 표현한다.

	model	accuracy	precision	recall	f1	auc
3	RandomForestClassifier	0.884681	0.884434	0.885436	0.884918	0.953972
0	RandomForestClassifier	0.884681	0.884434	0.885436	0.884918	0.953972
4	RandomForestClassifier	0.884681	0.884434	0.885436	0.884918	0.953972
3	SVC	0.884077	0.879098	0.891096	0.885026	0.926520
0	SVC	0.884077	0.879098	0.891096	0.885026	0.926520
0	ExtraTreesClassifier	0.883972	0.881911	0.887113	0.884490	0.952944
3	ExtraTreesClassifier	0.883972	0.881911	0.887113	0.884490	0.952944
2	RandomForestClassifier	0.883894	0.882452	0.886222	0.884316	0.953741
4	ExtraTreesClassifier	0.883789	0.881867	0.886746	0.884285	0.953124
2	ExtraTreesClassifier	0.883290	0.879037	0.889314	0.884141	0.952086
2	SVC	0.883106	0.876733	0.891987	0.884286	0.929656
0	DecisionTreeClassifier	0.880980	0.883344	0.878361	0.880812	0.941001
3	DecisionTreeClassifier	0.880980	0.883344	0.878361	0.880812	0.941001
0	KNeighborsClassifier	0.880298	0.872826	0.890782	0.881703	0.946957
2	KNeighborsClassifier	0.880167	0.871869	0.891777	0.881704	0.946293
3	KNeighborsClassifier	0.880167	0.871869	0.891777	0.881704	0.946293
4	KNeighborsClassifier	0.879904	0.874351	0.887794	0.881009	0.947565
4	SVC	0.879248	0.904032	0.849012	0.875649	0.949472
4	DecisionTreeClassifier	0.879065	0.882938	0.874536	0.878639	0.948238

대체로 트리 기반 머신러닝 알고리즘들의 성능이 높음을 알 수 있다.

그래서 KU.GG에 사용될 오브젝트 분석 모델은 이러한 트리 기반의 알고리즘에 하이퍼 파라미터 튜닝, feature engineering, 앙상블 등을 적용해 최대한 성능을 올린 모델이 사용될 예정이다.



챔피언 분석(아이템 트리 시뮬레이션)

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix



챔피언 분석의 아이템트리 분석의 경우 유저의 아이템 사용 기록과 전적 데이터를 바탕으로 모델을 학습시킨다. 이를 바탕으로 내 아이템 트리와 상대방 아이템 트리에 따라 승리 확률을 예측할 수 있게 해주는 '아이템트리 시뮬레이션' 을 제작한다.

matchid	win	championID	position	item0	item1	item2	item3	item4	item5	
0	0	Fail	57	DUO_CARRY	1054	1029	3076	3065	3068	3047
1	0	Fail	235	DUO_SUPPORT	2055	3864	3009	3071	3179	3134
2	0	Fail	33	JUNGLE	1401	3075	3001	1029	1011	3009
3	0	Fail	142	MIDDLE	1082	3285	2033	3111	3165	3108
4	0	Fail	83	TOP	3140	2421	3078	3047	3812	3053
...
62395	6239	Win	103	MIDDLE	2033	3030	3905	0	1082	3020
62396	6239	Win	81	DUO_CARRY	3285	2422	3144	3025	3042	1055
62397	6239	Win	84	TOP	3146	2421	1026	3191	3047	1055
62398	6239	Win	59	JUNGLE	2055	3077	3047	1412	3071	0
62399	6239	Win	111	DUO_SUPPORT	1028	3401	2055	3117	3024	1029

62400 rows x 10 columns

Raw데이터에서 필요한 데이터 추출

상대방의 아이템 트리와 결합

matchid	win	championID	position	item0	item1	item2	item3	item4	item5	win_opnt	championID_opnt	item0_opnt	item1_opnt	item2_opnt	item3_opnt	
0	0	Win	90	MIDDLE	3151	3116	3041	1052	3020	3285	Fail	142	1082	3285	2033	3111
1	0	Win	22	DUO_CARRY	3153	3085	3140	3031	1042	3006	Fail	57	1054	1029	3076	3065
2	0	Win	126	TOP	3142	3814	2420	3071	3123	3111	Fail	83	3140	2421	3078	3047
3	0	Win	117	DUO_SUPPORT	0	3107	3080	3158	3174	3504	Fail	235	2055	3064	3009	3071
4	0	Win	245	JUNGLE	3191	3152	2420	1402	3100	3020	Fail	33	1401	3075	3001	1029
...
97100	6239	Fail	89	DUO_SUPPORT	1001	3076	3401	1028	1004	0	Win	111	1028	3401	2055	3117
97101	6239	Fail	122	TOP	1054	3111	2031	3071	3742	0	Win	84	3146	2421	1026	3191
97102	6239	Fail	84	JUNGLE	2055	2031	3077	1400	3111	0	Win	59	2055	3077	3047	1412
97103	6239	Fail	145	DUO_CARRY	1042	1042	1055	3124	3004	3006	Win	81	3285	2422	3144	3025
97104	6239	Fail	183	MIDDLE	2065	3117	3133	0	0	0	Win	103	2033	3030	3905	0

Unnamed: 0	matchid	win	championID	position	item0	item1	item2	item3	item4	...	win_opnt	championID_opnt	item0_opnt	item1_opnt	item2_opnt	ite
0	0	0	Win	90	MIDDLE	1052	3020	3041	3116	3151	...	Fail	142	1082	2033	3108
1	1	0	Win	22	DUO_CARRY	1042	3006	3031	3085	3140	...	Fail	57	1029	1054	3047
2	2	0	Win	126	TOP	2420	3071	3111	3123	3142	...	Fail	83	2421	3047	3053
3	3	0	Win	117	DUO_SUPPORT	0	3107	3158	3174	3504	...	Fail	235	2055	3009	3071
4	4	0	Win	245	JUNGLE	1402	2420	3020	3100	3152	...	Fail	33	1011	1029	1401
...
92217	92217	6239	Fail	89	DUO_SUPPORT	0	1001	1004	1028	3076	...	Win	111	1028	1029	2055
92218	92218	6239	Fail	122	TOP	0	1054	2031	3071	3111	...	Win	84	1026	1055	2421
92219	92219	6239	Fail	84	JUNGLE	0	1400	2031	2055	3077	...	Win	59	0	1412	2055
92220	92220	6239	Fail	145	DUO_CARRY	1042	1042	1055	3004	3006	...	Win	81	1055	2422	3025
92221	92221	6239	Fail	183	MIDDLE	0	0	0	2065	3117	...	Win	103	0	1082	2033

아이템 트리 오름차순 정렬

챔피언 분석(아이템 트리 시뮬레이션)

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix

데이터가 숫자형이지만 아이템 id를 나타낸 만큼 범주형 데이터 이므로 원핫 인코딩을 통해 데이터간 관계가 독립적임을 표현후 아이템트리를 사용한 승률예측 모델을 학습하고 유저가 설정한 아이템트리를 토대로 승률 예측을 한다.

	win	championID_1	championID_2	championID_3	championID_4	championID_5	championID_6	championID_7	championID_8	championID_9	...	iti
0	1	0	0	0	0	0	0	0	0	0	...	0
1	1	0	0	0	0	0	0	0	0	0	...	0
2	1	0	0	0	0	0	0	0	0	0	...	0
3	1	0	0	0	0	0	0	0	0	0	...	0
4	1	0	0	0	0	0	0	0	0	0	...	0
...
92217	0	0	0	0	0	0	0	0	0	0	...	0
92218	0	0	0	0	0	0	0	0	0	0	...	0
92219	0	0	0	0	0	0	0	0	0	0	...	0
92220	0	0	0	0	0	0	0	0	0	0	...	0
92221	0	0	0	0	0	0	0	0	0	0	...	0

92222 rows × 2321 columns

```
from sklearn.model_selection import train_test_split
X=df_item_num_v2.drop(['win'],axis=1).copy()
y=df_item_num_v2['win']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1, random_state=42)
executed in 886ms, finished 16:07:21 2020-11-03
```

```
from sklearn.metrics import accuracy_score
random_forest = RandomForestClassifier(n_estimators=100)
rf=random_forest.fit(X_train, y_train)
y_pred = rf.predict(X_test)
accuracy_score(y_test, y_pred)
executed in 4m 19s, finished 16:11:46 2020-11-03
```

0.7321912609779898

```
with open("item_tree.pkl", "rb") as file:
    rf = pickle.load(file)
win_rate=rf.predict_proba(X_test)[0,:]*100
win_rate[0][1]
executed in 598ms, finished 13:25:47 2020-11-04
```

7.000000000000001

벡터의 크기가 커서 시간이 오래 걸리는 바람에 이번 구현에서는 랜덤 포레스트 하나만 확인해서 사용했으나 더 많은 머신러닝 모델과 하이퍼 파라미터를 비교해 성능 분석을 할 예정이다



챔피언 분석(챔피언 추천)

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix

챔피언 추천은 소환사 상세 분석에서 유저의 전적기록을 바탕으로 주로 사용하는 챔피언을 가지고 그와 가장 비슷한 아이템 트리를 가진 챔피언을 유저에게 추천해줄수 있는 시스템이다.

matchid	win	championID	position	item0	item1	item2	item3	item4	item5	
0	0	Fail	57	DUO_CARRY	1054	1029	3076	3065	3068	3047
1	0	Fail	235	DUO_SUPPORT	2055	3864	3009	3071	3179	3134
2	0	Fail	33	JUNGLE	1401	3075	3001	1029	1011	3009
3	0	Fail	142	MIDDLE	1082	3285	2033	3111	3165	3108
4	0	Fail	83	TOP	3140	2421	3078	3047	3812	3053
...
62395	6239	Win	103	MIDDLE	2033	3030	3905	0	1082	3020
62396	6239	Win	81	DUO_CARRY	3285	2422	3144	3025	3042	1055
62397	6239	Win	84	TOP	3146	2421	1026	3191	3047	1055
62398	6239	Win	59	JUNGLE	2055	3077	3047	1412	3071	0
62399	6239	Win	111	DUO_SUPPORT	1028	3401	2055	3117	3024	1029

62400 rows x 10 columns



championID	item0	item1	item2	item3	item4	item5	position	
0	266	0	0	3047	3053	3071	3812	TOP
1	266	1037	1055	3047	3052	3071	3812	TOP
2	266	1028	1037	3047	3052	3071	3812	TOP
3	266	1028	1054	3047	3052	3071	3812	TOP
4	266	0	1054	3047	3053	3071	3812	TOP
...
1495	876	1402	2421	3020	3116	3123	3151	JUNGLE
1496	876	2420	3027	3108	3110	3111	3194	JUNGLE
1497	876	1056	2421	3020	3116	3151	3191	JUNGLE
1498	876	1031	1402	3020	3116	3151	3157	JUNGLE
1499	876	1056	1056	1056	1056	1058	1058	JUNGLE

Raw 데이터에서 아이템트리와 전적정보 추출

챔피언별 최빈 아이템트리 상위 10개 추출

championID	item0_1001	item0_1004	item0_1006	item0_1011	item0_1018	item0_1026	item0_1028	item0_1029	item0_1031	...	item5_3863	item5_3
0	266	0	0	0	0	0	0	2	0	0	0	0
1	103	0	0	0	0	0	1	0	0	0	0	0
2	84	0	0	0	0	0	1	0	0	0	0	0
3	12	0	0	0	0	0	0	1	0	0	0	0
4	32	1	0	0	0	0	1	1	0	0	0	0
...
145	26	0	0	0	0	0	2	0	0	0	0	0
146	142	0	0	0	0	0	3	0	0	0	0	0
147	143	0	0	0	0	0	3	0	0	0	0	0
148	777	0	0	0	0	1	0	0	0	0	0	0
149	876	0	0	0	0	0	0	0	0	1	0	0



```

from sklearn.cluster import KMeans

km = KMeans(n_clusters=50, random_state=0)
y_km = km.fit_predict(df_sc_v5[x_columns])
print(y_km)

executed in 339ms, finished 15:29:20 2020-11-03

[20  9 24  6 40 19  3  5 25 22 12 46 32  8 32  5 43 22  9 10 14 28  5  2
 44 16 40 41 16 43 13 18 10 10  2 16 29 10  1 20  0 46  4 34 10 29  5  5
 36 25  4 16 22 24 21  7 18 29 48 38 17 12 34  6 12  0  4 27 12 12 33  2
 15 20 13 46  4  2  6 12 45 29  9 34 12 30 29 33 23  7 36  6 31 34 20 20
 20 22 22 31 49  9 39 33 16 37 33 15  9  4 46 22 12  3  9 22  7  6 22 11
  5 40  0 42 25  2 20 17 17  3 27 14  1 18  9 10 15 27 34 47 10  4 31 29
  3 27 12  8 47 26]

# 결과
df_sc[df_sc['championID']==64]
executed in 9ms, finished 13:29:34 2020-11-04

championID  cluster
62           64      10

# 최종 결과 출력, 클러스터, 클러스터, 챔피언, 아이템, 아이템
df_sc[df_sc['cluster']==10]
executed in 9ms, finished 13:29:34 2020-11-04

championID  cluster
43           59      10
58           203     10
62           64      10
63            2       10
93           421     10
131          254     10
138            5       10
    
```

아이템트리 데이터에 클러스터링을 적용하기 위해 아이템별 빈도수를 세는 방법으로 벡터화를 한다. 챔피언간의 거리를 측정하여 군집화를 하는 K-Means 클러스터링을 사용하여 챔피언 군집분석을 수행한 뒤 유저에게 챔피언을 추천해준다



챔피언 분석(챔피언 추천)

결과물을 확인해보면 모든 챔피언의 주요 라인이 정글로 동일하고 아이템트리 유사도도 어느정도 일치하는걸 확인할수 있다

주요 아이템 빌드	
시작 아이템	
주요 빌드	
신발	

자르반 4세

주요 아이템 빌드	
시작 아이템	
주요 빌드	
신발	

올라프

주요 아이템 빌드	
시작 아이템	
주요 빌드	
신발	

킨드레드

주요 아이템 빌드	
시작 아이템	
주요 빌드	
신발	

렉사이

주요 아이템 빌드	
시작 아이템	
주요 빌드	
신발	

바이

주요 아이템 빌드	
시작 아이템	
주요 빌드	
신발	

신짜오

추천 아이템 빌드	
시작 아이템	
주요 빌드	> >
신발	

리신

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix



KUGG WEB

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

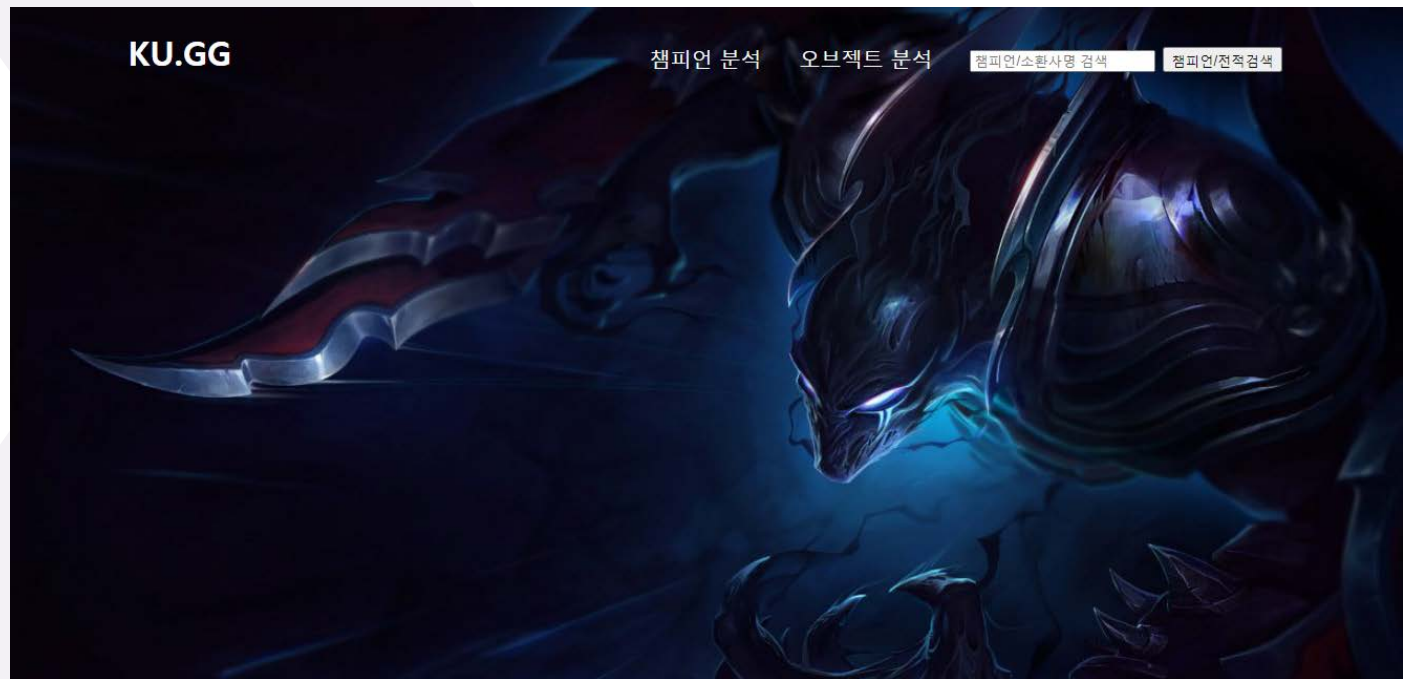
KU.GG
WEB

System
Test

Traceability
Matrix

1. KUGG WEB의 홈 화면

배너의 각 탭을 누르거나 검색하면 해당 페이지로 이동



2. DB에 존재하는 소환사명 검색

사과 챔피언/전적검색

sname	profileIconId	revisionDate	summonerLevel
사과	4103	1601479095214	349

소환사 정보와 게임 기록 출력

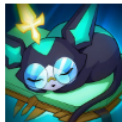
KU.GG

챔피언 분석

오브젝트 분석

챔피언/소환사명 검색

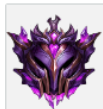
챔피언/전적검색



Level 349

사과
66 LP

"최근 업데이트 : " 2020-10-01 00:18:15



솔로랭크
MASTER I
337승 321패
승률 51%
66LP
아리의 주문술사를

솔로랭크

챔피언 이름 cs 142.3 (5.0) 1.95:1 평점 50%
4.0 / 5.1 / 6.128 게임

솔랭
~일 전
승리
15분 12초



0 / 1 / 4
4.0:1 평점

8 Level
3 CS
딜량 7572



rune1 rune2



작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

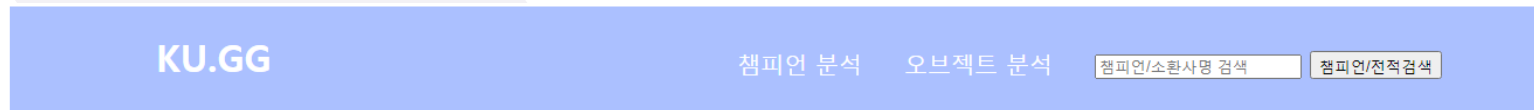
System
Test

Traceability
Matrix

2. DB에 존재하지 않는 소환사명 검색



소환사 정보와 게임 기록이 출력되지 않음



해당 소환사/챔피언 정보가 없습니다

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

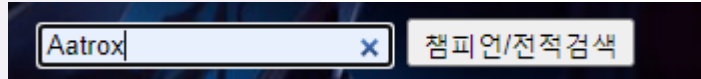
System
Test

Traceability
Matrix



KUGG WEB

3. 챔피언 이름 검색



챔피언 분석 페이지로 이동하며, 해당 챔피언이 자동으로 선택되고, 상대 챔피언과 아이템을 입력하면 해당 승률을 확인할 수 있다.

KU.GG 챔피언 분석 오브젝트 분석 챔피언/전적검색

챔피언 선택
티어를 선택하세요 ▼ 라인을 선택하세요 ▼

검색한 챔피언: Aatrox

챔피언 분석

신발 ▼ 상대신발 ▼ 방어구 ▼ 상대방어구 ▼ ad무기 ▼ 상대ad무기 ▼ ap무기 ▼ 상대ap무기 ▼
서포템 ▼ 상대서포템 ▼ 정글템 ▼ 상대정글템 ▼

 ○ 상대챔피언
 ○ 상대챔피언

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix



KUGG WEB

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix

5. 챔피언 분석 페이지

티어, 라인, 내 챔피언, 상대 챔피언을 선택하면 상대 승률을 확인할 수 있다.
추가로 아이템 정보를 입력하면 아이템 승률 시뮬레이션도 확인할 수 있다.

KU.GG

챔피언 분석

오브젝트 분석

챔피언/전적검색

챔피언 선택

마스터 MID 챔피언 분석

신발 헤르메스의 발걸음 방어구 란두인의 예언 ad무기 질흑의 양날도끼 ap무기 라바돈의 죽음모자
서폿템 구원 정글템 검:갯불거인



내챔피언 상대챔피언



내챔피언 상대챔피언



내챔피언 상대챔피언

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix

6. 챔피언 분석 결과

내 챔피언과 상대 챔피언 사이의 상대승률과 추천 아이템 트리를 확인할 수 있다.



선택한 티어 : Master

선택한 라인 : MID



내 챔피언[Aatrox]

승률 : 57.99999999999999%

{

추천 아이템 트리 :

추천 아이템1
Boots of Speed/

추천 아이템2
Doran's Shield/

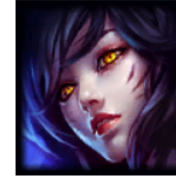
추천 아이템3
Refillable Potion/

추천 아이템4
Phage/

추천 아이템5
Kindlegem/

추천 아이템6
Caulfield's Warhammer

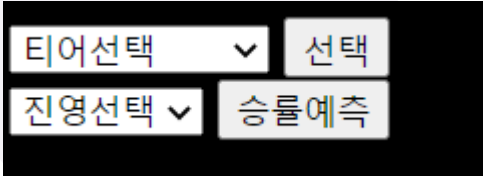
vs



상대 챔피언[Ahri]

7. 오브젝트 분석 페이지

분석을 희망하는 티어와 진영을 선택하고, 획득한 오브젝트를 선택하고 승률 예측 버튼을 클릭하면 오브젝트 승률 시뮬레이션을 확인할 수 있다.



The image shows a UI control panel with a black background. It contains four white buttons: a dropdown menu labeled '티어선택' (Tier Select) with a downward arrow, a button labeled '선택' (Select), a dropdown menu labeled '진영선택' (Team Select) with a downward arrow, and a button labeled '승률예측' (Win Rate Prediction).



작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix

8. 오브젝트 승률 시뮬레이션 결과



System Test

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix



Test number	Test 항목	Description	Expected result	System function	P/F	비고
1	페이지 이동	1. 메뉴에서 페이지 탭 중 하나를 클릭한다. 2. 메인 배너를 클릭한다.	페이지 탭을 누르면 선택한 페이지로 이동하고 메인 배너를 누르면 홈페이지로 이동한다.	3.2.1.1	●	
2	챔피언 명 or 소환사명 검색	1. 메뉴에서 검색창에 챔피언 명 또는 소환사명을 검색한다. 2. 입력한 챔피언 정보 또는 소환사 정보 페이지로 이동한다.	DB에 해당하는 챔피언 또는 소환사 정보가 있다면 해당 챔피언 또는 소환사 정보가 보인다.	3.2.1.2	●	
3	존재하지 않는 챔피언 명 or 소환사명 검색	1. 메뉴에서 검색창에 챔피언명 또는 소환사명을 검색한다. 2. 챔피언 정보 또는 소환사 정보 페이지로 이동한다.	DB에 해당 챔피언 또는 소환사 정보가 없으면 '존재하지 않는 챔피언 or 소환사입니다.'가 출력된다.	3.2.1.2	●	
4	유저 챔피언 정보	1. 챔피언을 하나 선택한다	유저가 선택한 챔피언과 일치하는 라인과 아이템 트리를 보여준다.	3.2.2.1	✕	모듈삭제
5	유저 챔피언 정보 취소 후 재선택	1. 챔피언을 하나 선택한다. 2. 선택을 취소하고 다른 챔피언을 선택한다.	선택한 챔피언과 일치하는 라인과 아이템 트리를 보여주고 유저가 다른 챔피언을 선택하면 그에 맞추어 라인과 아이템트리도 바뀐다.	3.2.2.1	●	
6	상대 챔피언 정보	1. 챔피언과 라인을 선택한다.	선택한 챔피언과 라인이 표시되고 상대승률과 승리전략을 보여준다.	3.2.2.2	▲	아직 웹에 반영되어 있지 않음
7	상대 챔피언 정보 취소 후 재선택	1. 챔피언과 라인을 선택한다. 2. 선택을 취소하고 다른 챔피언과 라인을 선택한다.	선택한 챔피언과 라인이 표시되고 상대승률과 승리전략을 보여주고 유저가 다른 챔피언과 라인을 선택하면 그에 맞추어 상대승률과 승리전략이 바뀐다.	3.2.2.2	▲	아직 웹에 반영되어 있지 않음
8	아이템 트리 시뮬레이션	1. 챔피언을 선택한다. 2. 상대 챔피언을 선택한다. 3. 새로운 테스트 데이터를 대입하여 승패여부를 예측한다.	선택한 챔피언과 상대 챔피언이 표시되고 아이템마다 승리할 확률이 나타난다.	3.2.2.3	▲	
9	소환사 검색	검색 창에 소환사명을 입력하고 검색버튼을 누른다.	입력한 소환사명이 DB에 있다면 해당소환사의 전적 정보 페이지로 이동한다.	3.2.3.1	●	
10	존재하지 않는 소환사 검색	검색창에 소환사명을 입력하고 검색버튼을 누른다.	입력한 소환사명이 DB에 없을 경우 '존재하지 않는 소환사명입니다.'가 출력된다.	3.2.3.1	●	

System Test

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix



Test number	Test 항목	Description	Expected result	System function	P/F	비고
11	소환사 분석	소환사 분석 페이지에서 소환사명을 검색한다.	입력한 소환사명의 정보를 다이어그램으로 나타내고 다른 챔피언을 추천한다. 입력한 소환사명의 자주 선택하는 챔피언, 아이템트리, 특성, 자주 가는 라인을 출력한다.	3.2.4.1, 3.2.4.2	✖	모듈삭제
12	존재하지 않는 소환사 분석	소환사 분석 페이지에서 소환사명을 검색한다.	입력한 소환사명이 DB에 없을 경우 '존재하지 않는 소환사명입니다.'가 출력된다.	3.2.5.1, 3.2.5.2	✖	모듈삭제
13	첫번째 오브젝트 승률 예측	1. 드롭다운에서 티어 하나를 선택한다. 2. 첫번째로 획득한 오브젝트의 체크박스를 클릭한다. 3. 승률 예측 버튼을 클릭한다.	선택된 티어와 첫번째 오브젝트에 대해 예측된 승률 정보와, 이번 시즌 가장 승률이 높은 첫번째 오브젝트를 출력된다.	3.2.5.1, 3.2.5.2	▲	오브젝트 분석의 컨셉이 바뀜
14	오브젝트 킬당 승률 예측	1. 드롭다운에서 티어 하나를 선택한다. 2. 각 오브젝트 드롭다운에서 킬수를 선택한다. 3. 승률 예측 버튼을 클릭한다. 4. 승률과 추천 오브젝트 출력을 확인한다.	선택된 티어와 오브젝트 킬수에 대해 예측된 승률과 추천 오브젝트가 화면에 출력된다.	3.2.5.3, 3.2.5.4	▲	오브젝트 분석의 컨셉이 바뀜
15	첫번째 오브젝트 & 오브젝트 킬당 승률 예측	1. 드롭다운에서 티어 하나를 선택한다. 2. 첫번째로 획득한 오브젝트 체크박스를 선택한다. 3. 각 오브젝트 드롭다운에서 킬수를 선택한다. 4. 승률 예측 버튼을 클릭한다. 5. 승률 출력을 확인한다.	선택된 티어와 첫번째 오브젝트, 그리고 오브젝트 킬수에 대해 예측된 승률이 화면에 출력된다.	3.2.5.5, 3.2.5.6	▲	오브젝트 분석의 컨셉이 바뀜
16	유저 티어 정보 요청	1. 시스템이 티어정보를 불러오기 위해 api에 request를 보낸다. 2. request의 status 코드가 정상적인지 확인한다. 3. 정상적인 코드가 아닐 경우 다시 request를 보낸다. 4. request를 통해 얻은 데이터를 메모리에 저장한다.	시스템이 DB의 유저 티어 정보 데이터 중 아직 request를 보내지 않는 유저 티어 정보데이터에 대해 request를 보낸다. request에 대한 status 코드가 200이 나온다. request에 대한 status 코드가 200이 아닐 경우 다시 request를 요청한다. request를 통해 얻은 데이터를 정상적으로 메모리에 올린다. 데이터를 DB에 저장할 때 중복되는 것은 제외하고 저장한다.	3.2.6.1	●	

System Test

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix



Test number	Test 항목	Description	Expected result	System function	P/F	비고
17	유저 티어 정보 DB 저장	1. 시스템은 DB에 연결을 요청한다. 2. 시스템은 메모리에 있는 유저 티어 데이터를 DB에 저장한다.	DB에 정상적으로 연결된다. 메모리에 있는 유저 티어 정보 데이터가 정상적으로 DB에 저장된다.	3.2.6.1	●	
18	유저 상세정보 요청	1. 시스템이 유저 티어 정보 DB에 새롭게 추가된 유저들을 뽑아낸다. 2. 시스템이 새롭게 추가된 유저들의 상세정보를 얻기 위해 api에 request를 보낸다. 3. request의 status 코드가 정상적인지 확인한다. 4. 정상적인 코드가 아닐경우 다시 request를 보낸다. 5. request를 통해 얻은 데이터를 메모리에 저장한다.	시스템이 DB의 유저 상세정보 데이터 중 아직 request를 보내지 않는 유저 상세정보 데이터에 대해 request를 보낸다. request에 대한 status code가 200이 나온다. request를 통해 얻은 데이터를 정상적으로 메모리에 올린다. 데이터를 DB에 저장할 때 중복되는 것은 제외하고 저장한다.	3.2.6.2	●	
19	유저 상세정보 DB 저장	1. 시스템은 DB에 연결을 요청한다. 2. 시스템은 메모리에 있는 유저 상세정보 데이터를 DB에 저장한다.	DB에 정상적으로 연결된다. 메모리에 있는 유저 상세 정보 데이터가 정상적으로 DB에 저장된다.	3.2.6.2	●	
20	유저 matchlist 데이터 요청	1. 시스템은 유저들이 최근에 플레이한 게임을 DB에서 가져온다. 2. 시스템은 가장 최근에 플레이한 데이터 이후의 게임들에 대한 리스트를 받기 위해 api에 request를 보낸다. 3. request의 status 코드가 정상적인지 확인한다. 4. 정상적인 코드가 아닐경우 다시 request를 보낸다. 5. request를 통해 얻은 유저의 matchlist 데이터를 메모리에 저장한다.	시스템이 DB의 matchlist데이터 중 아직 request를 보내지 않은 matchlist에 대해 request를 보낸다. request에 대한 status code가 200이 나온다. request를 통해 얻은 데이터를 정상적으로 메모리에 올린다. 데이터를 DB에 저장할 때 중복되는 것은 제외하고 저장한다.	3.2.6.3	●	
21	유저 matchlist 데이터 DB 저장	1. 시스템은 DB에 연결을 요청한다. 2. 시스템은 메모리에 있는 matchlist 데이터들을 DB에 저장한다.	DB에 정상적으로 연결된다. 메모리에 있는 유저 matchlist 정보가 정상적으로 DB에 저장된다.	3.2.6.3	●	

System Test

Test number	Test 항목	Description	Expected result	System function	P/F	비고
22	Match 상세 정보 요청	<ol style="list-style-type: none"> 1. 시스템은 메모리에 올라가있는 matchlist들에서 최근 match들만을 뽑아낸다. 2. 시스템은 최근 match에 대해서 해당 match들의 상세정보에 대해 api에 request를 보낸다. 3. request의 status 코드가 정상적인지 아닌지 확인한다. 4. 정상적인 코드가 아닐경우 다시 request를 보낸다. 5. request를 통해 얻은 match 상세 정보 데이터를 메모리에 저장한다. 	<p>시스템이 DB의 match 상세정보 데이터 중 아직 request를 보내지 않은 match에 대해 request를 보낸다. request에 대한 status code가 200이 나온다. request를 통해 얻은 데이터를 정상적으로 메모리에 올린다. 데이터를 DB에 저장할 때 중복되는 것은 제외하고 저장한다.</p>	3.2.6.4	●	
23	Match 상세 정보 DB 저장	<ol style="list-style-type: none"> 1. 시스템은 DB에 연결을 요청한다. 2. 시스템은 메모리에 있는 match 상세정보 데이터들을 DB에 저장한다. 	<p>DB에 정상적으로 연결된다. 메모리에 있는 match 상세 정보 데이터가 정상적으로 DB에 저장된다.</p>	3.2.6.4	●	
24	Match 타임라인 데이터 요청	<ol style="list-style-type: none"> 1. 시스템은 메모리에 올라가있는 matchlist들에서 최근 match들만 뽑아낸다. 2. 시스템은 최근 match에 대해서 해당 match들의 타임라인 정보에 대해 api에 request를 보낸다. 3. request의 status 코드가 정상적인지 아닌지 확인한다. 4. 정상적인 코드가 아닐경우 다시 request를 보낸다. 5. request를 통해 얻은 match 타임라인 데이터를 메모리에 저장한다. 	<p>시스템이 DB의 match 타임라인 데이터 중 아직 request를 보내지 않은 match 타임라인에 대해 request를 보낸다. request에 대한 status code가 200이 나온다. request를 통해 얻은 데이터를 정상적으로 메모리에 올린다. 데이터를 DB에 저장할 때 중복되는 것은 제외하고 저장한다.</p>	3.2.6.5	●	
25	Match 타임라인 데이터 DB 저장	<ol style="list-style-type: none"> 1. 시스템은 DB에 연결을 요청한다. 2. 시스템은 메모리에 있는 match 타임라인 데이터들을 DB에 저장한다. 	<p>DB에 정상적으로 연결된다. 메모리에 있는 match 타임라인 정보 데이터가 정상적으로 DB에 저장된다.</p>	3.2.6.5	●	

작품 개요

데이터 수집 모듈

데이터 분석 과정

오브젝트 분석

챔피언 분석

KU.GG WEB

System Test

Traceability Matrix



System Test – Pass/Fail Criteria

Test number	Test 항목	Description	Expected result	System function	P/F	비고
26	데이터 전처리	<ol style="list-style-type: none"> 1. 시스템이 LoL api에 정상적으로 request를 보냈는지 확인한다. 2. 시스템이 request를 통해 얻은 데이터에 전처리 작업을 정상적으로 수행했는지 확인한다. 3. 시스템이 전처리 작업을 거친 데이터를 데이터베이스에 저장했는지 확인한다. 	결측치나 이상치들을 제외한 유용한 데이터들만 선별하고 필요하면 왜곡된 데이터 수정도 마친 데이터들이 데이터 베이스에 저장된다.	3.2.6.6	▲	이 부분은 테스트가 불가능한 부분
27	모델 입출력	<ol style="list-style-type: none"> 1. 데이터 모델이 오버피팅이나 언더피팅이 되진 않았는지 확인하기 위해 여러가지 데이터 input을 준비한다. 2. 데이터 모델에 input을 입력한다. 	모델별로 input에 맞는 output 데이터를 오버피팅이나 언더피팅없이 높은 정확도로 결과를 예측한다.	3.2.6.7	▲	이 부분 역시 오버피팅은 테스트를 할 수 없음

Success Criteria

Test number	Test 항목	Description	Expected result	System function	P/F	비고
1	게임 실력 향상 여부	<ol style="list-style-type: none"> 1. LOL을 플레이하는 유저들에게 KUGG를 사용하게 한다. 2. 일정 기간 이후 그들의 게임 실력 향상 여부를 확인한다. 	실제 유저 대상으로 실력 향상에 도움이 되었다고 증언받는다.	Success Criteria 1	▲	게임을 하는데는 도움이 되지 않지만 승리에 도움이 되는지는 아직 잘 모르겠다는 의견
2	최신 정보 출력	<ol style="list-style-type: none"> 1. 소환사명을 검색한다. 2. 검색된 소환사의 최신 정보가 출력된 것을 확인한다. 	유저가 개인의 소환사명을 검색하면 5초 내에 최신정보를 받아온다.	Success Criteria 2	●	

작품 개요

데이터 수집 모듈

데이터 분석 과정

오브젝트 분석

챔피언 분석

KU.GG WEB

System Test

Traceability Matrix



Traceability Matrix

작품 개요

데이터 수집 모듈

데이터 분석 과정

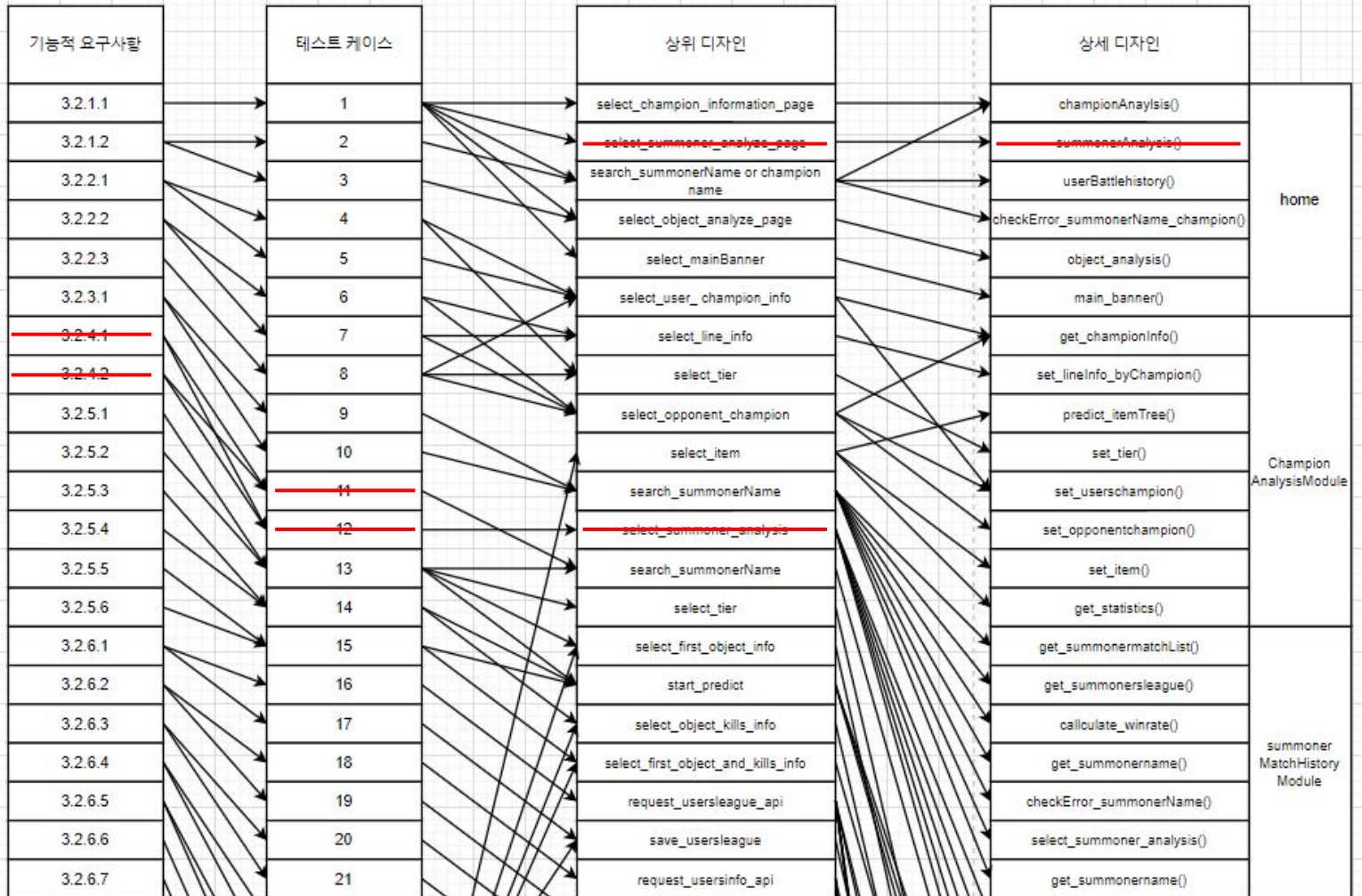
오브젝트 분석

챔피언 분석

KU.GG WEB

System Test

Traceability Matrix



Traceability Matrix

작품 개요

데이터 수집
모듈

데이터
분석 과정

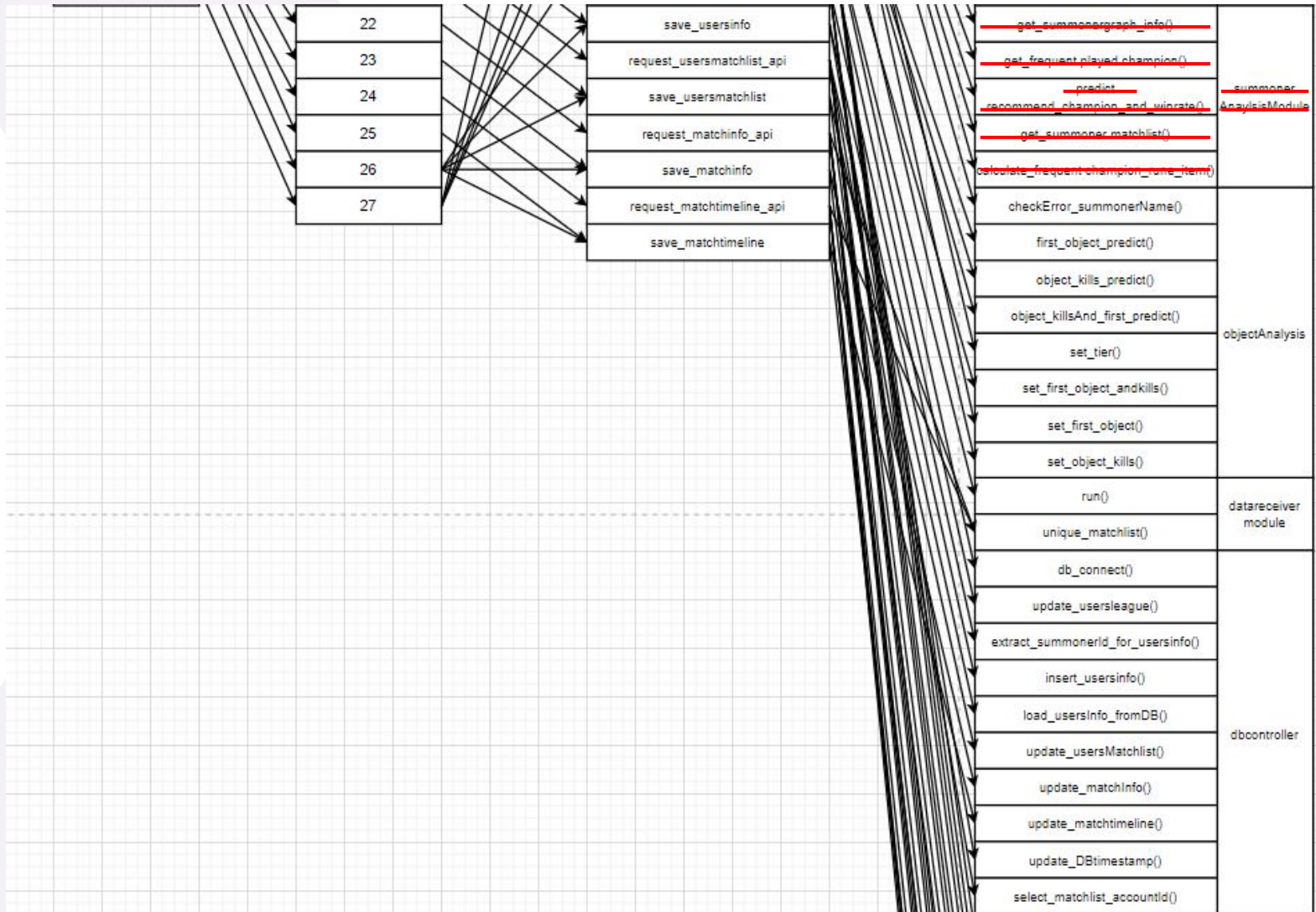
오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix



Traceability Matrix

작품 개요

데이터 수집
모듈

데이터
분석 과정

오브젝트
분석

챔피언 분석

KU.GG
WEB

System
Test

Traceability
Matrix

set_tiers()	usersleague_ receiver
make_usersleague_format()	
request_usersleague()	
run()	
make_usersinfo_format()	usersinfo_ receiver
request_userinfo()	
run()	
set_dbcontroller()	usersmatchlist_ receiver
update_db()	
make_usermatchlist_format()	
request_usersmatchlist()	
run()	
set_dbcontroller()	matchinfo_ receiver
update_db()	
make_matchinfo_format()	
request_matchinfo()	
run()	matchtimeline_ receiver
set_dbcontroller()	
update_db()	
make_matchtimeline_format()	
request_matchtimeline()	
run()	