МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное

образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ

ЭКОНОМИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информатики и прикладной математики

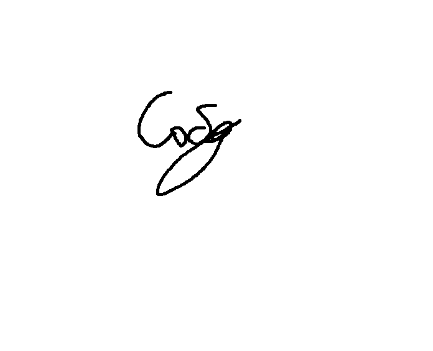
Кафедра прикладной математики и экономико-математических методов

**КУРСОВАЯ РАБОТА**

по дисциплине:

**«Методы анализа данных»**

Тема: «Сбор и анализ данных о результатах матчей в компьютерной игре Dota 2**»**

Направление: 01.03.02 Прикладная математика и информатика

Студент: Соболев Даниил Дмитриевич

Группа: ПМ-1801 Подпись:

Проверил: Карахтанова А. Б.

Должность: к.т.н., доцент

Оценка: Дата:

Подпись:

Санкт-Петербург

2021

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc90948968)

[1. ПОСТАНОВКА ЗАДАЧИ 5](#_Toc90948969)

[2. ОПИСАНИЕ ДАННЫХ 8](#_Toc90948970)

[2.1. Сбор данных 8](#_Toc90948971)

[2.2. Обработка данных 10](#_Toc90948972)

[2.3. Описание признаков 11](#_Toc90948973)

[3. АНАЛИЗ ДАННЫХ И ВИЗУАЛИЗАЦИЯ 14](#_Toc90948974)

[ЗАКЛЮЧЕНИЕ 20](#_Toc90948975)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 21](#_Toc90948976)

[Приложение А SQL запросы для извлечения данных о матчах и игроках 22](#_Toc90948977)

[Приложение Б Функция для обработки данных о предметах у игроков 23](#_Toc90948978)

[Приложение В Функция для обработки данных о строениях 25](#_Toc90948979)

ВВЕДЕНИЕ

В настоящее время киберспорт является одним из самых популярных молодых видов спорта. Он пользуется активной популярностью среди молодежи, в него вкладываются инвесторы, что соответственно привлекает еще больше внимания. По данным на январь 2020 года российский рынок киберспорта вырос на 42% и оценивается в 12.7 млн долларов США. Эксперты прогнозируют, что рынок будет расти примерно на 20% в год и к 2023 году может достигнуть объема 31.2 млн долларов США. Исследователи считают, что чем моложе люди, тем они более склонны соглашаться, что киберспорт относится к настоящим видам спорта.

Из-за того, что киберспортивные дисциплины отличаются от привычных видов спорта, таких как, например, футбол или теннис, и имеют свою специфику, для них очень удобно собирать различного рода данные и впоследствии проводить анализ, строить модели и предсказывать различные игровые ситуации. Таким образом, киберспортивные дисциплины привлекают множество людей, интересующихся обработкой и анализом данных.

Одной из самых популярных киберспортивных дисциплин на сегодняшний день является Dota 2 – компьютерная многопользовательская онлайн игра в жанре MOBA (multiplayer online battle arena), разработанная американской корпорацией Valve.

С точки зрения предмета для анализа данных данная игра интересна тем, что в ней присутствует огромное количество различных факторов, которые как по отдельности, так и в совокупности оказывают влияние на победу той или иной команды, а также тот факт, что все эти данные собираются автоматически для каждого матча.

В перспективе результаты анализа могут быть полезны широкому кругу людей, например, тренерам команд, аналитикам, букмекерам или же самим игрокам.

Цель данный курсовой работы заключается в сборе и анализе данных о матчах за определенный промежуток времени, а также в подготовке этих данных для обучения моделей.

# ПОСТАНОВКА ЗАДАЧИ

Сперва рассмотрим описание и процесс игры. Как уже было отмечено, Dota 2 представляет собой многопользовательскую игру в жанре MOBA, в каждом матче которой участвуют две команды по пять игроков. С одной стороны карты располагаются силы Света (англ. The Radiant), с другой стороны карты находятся силы Тьмы (англ. The Dire). Основная цель каждой из команд – сломать главную вражескую крепость – Трон (англ. Ancient), при этом защищая свои строения.

Каждый из десяти игроков в течение матча управляет одним персонажем, которого принято называть героем (англ. Hero). Перед началом игры каждый из игроков выбирает себе одного из 121 персонажа. Каждый герой по-своему уникален: он имеет свою узнаваемую внешность, уникальные способности и характеристики. По ходу матчу каждый игрок выполняет определенную роль такую как, например, саппорт (англ. Support) или керри (англ. Carry) и таким образом каждый из игровых персонажей может в большей или меньшей степени подходить для игры на определенной роли. В дальнейшем в течение матча герой может получать опыт, зарабатывать золото, на что впоследствии может приобретать различные предметы, которые всячески будут усиливать характеристики и способности персонажа. Если в какой-то момент времени в матче очки здоровья героя снижаются до нуля, то в таком случае герой в течение некоторого короткого промежутка времени выбывает из игры и считается «погибшим». По окончании этого времени герой возрождается и возвращается в бой.

Каждый матч проходит на квадратной карте специального вида, которая представлена на рисунке 1. У обоих команд крепости находятся в противоположных углах, а игроки сосредотачиваются по линиям, которые и соединяют две крепости. Помимо игровых персонажей во время матча принимают участие специальные существа, которые называются крипами. Их управление берет на себя компьютер. Также, у каждой из команд на каждой линии (всего их 3 на всей карте) стоят по 3 башни, которые запрограммированы атаковать вражеских существ и крипов, тем самым помогая своей команде.

Изображение выглядит как карта

Автоматически созданное описание

Рисунок 1 – Карта игры Dota 2

В свою очередь для более честной игры существуют туман войны, который покрывает большую часть карты и таким образом не позволяет следить за передвижениями противника на большом расстоянии.

Для добычи золотая в течение игры герой может добивать крипов (англ. Last Hit), таким образом получая за каждого добитого крипа определенную сумму золота и опыта. И впоследствии накапливая опыт для увеличения уровня, а золото для покупки предметов и повышения навыков и способностей.

В самом начале игры на севере реки возрождается особое нейтральное существо, называемое Рошаном (англ. Roshan). Для его убийства понадобится получить достаточно высокий уровень и иметь необходимые предметы. В качестве награды за убийства Рошана выпадает специальный предмет – Аегис (англ. Aegis). Данный предмет дарует владельцу вторую жизнь, то есть после того, как у игрока закончится определенное количество очков здоровья, спустя 5 секунд игрок вернется на место смерти, тем самым это является неотъемлемым преимуществуем в драках 5 на 5.

# ОПИСАНИЕ ДАННЫХ

## Сбор данных

Разработчики игры предусмотрели специальную функцию, которая сохраняет всю информацию о сыгранных матчах в игре Dota 2. Данная информация сохраняется в качестве видео с расширением . После игры пользователь может скачать такой файл локально к себе на компьютер и пересмотреть весь матч целиком, что делает эту функцию очень полезной для анализа матчей. У компании-разработчика Valve существует специальный сервис под названием Steam Web API, который предоставляет интерфейс для взаимодействия с данными из Steam. Среди этих данных также находятся и вышеупомянутые файлы с записями игр. Данные записи хранятся определенное количество времени – как правило 6 месяцев после того, как был сыгран определенный матч.

Соответственно для того, чтобы получать необходимые данные о матче, необходимо подключиться к сервису Steam и получить специальный файл для воспроизведения. Но есть более удобный способ для извлечения данных о матчах. Существует сервис под названием OpenDota, который умеет предоставлять данные в текстовом формате.

Данный сервис является проектом с открытым исходным кодом, таким образом каждый желающий может внести какие-либо улучшения или модификации в сбор данных. Также, несомненным плюсом этого сервиса является очень удобный интерфейс через GET или POST запросы. Единственным минусом данного приложения является тот факт, что на количество запросов существует ограничение – можно отправлять только 60 бесплатных запросов в минуту, либо, если вам нужно больше запросов, придется доплатить.

Соответственно, необходимо было решить данную проблему, так как стояла задача сформировать достаточно объемный набор данных из, как минимум, 3 тысяч матчей. А затем еще бы пришлось для каждого такого матча запрашивать данные для каждого из игроков, а их в каждом матче по 10 штук. К счастью, данный сервис также предоставляет доступ к функции «Проводник» – она позволяет формировать и выполнять дополнительные запросы о профессиональных матчах. Таким образом можно обходить ограничения в 60 запросов в минуту, так как с помощью SQL запроса можно запрашивать не один матч, а сразу множество матчей.

Для дальнейшего анализ были выбраны профессиональные матчи различных команд в период с 1.06.2021 по 1.10.2021. Для этого были сформированы специальные SQL запросы, которые представлены в приложении (Приложение А). Данный запрос отправляется на сервер с помощью HTTP GET запроса. После получения ответа данные от сервера сохраняются в JSON файл . По итогу было извлечено 4 тысячи матчей. Пример содержимого файла представлен на рисунке 2.

Изображение выглядит как стол

Автоматически созданное описание

Рисунок 2 – Пример ответа от сервера по запросу на извлечение матчей

На OpenDota находится очень много данных для каждого матча, но для создания набор данных были отобраны самые значимые признаки.

## Обработка данных

После извлечения данных из сервиса OpenDota далее следует обработать данные и приготовить их для обучения моделей. Так как данные пришли в формате JSON, то работать с ними очень удобно, так как в данном формате данные можно представлять в виде объектов пар ключ-значение. Сперва нужно было каждый матч разделить на временные промежутки, а именно – по 60 секунд каждый. Соответственно, если у нас матч длится, например, 40 минут – в результирующем наборе данных у такого матча будет 40 строк. Одна строка – конкретная минута конкретного матча.

Изображение выглядит как текст, белый

Автоматически созданное описание

Рисунок 3 – Пример данных для выбранного матча

Далее, необходимо было как-то разделить игроков по признакам, чтобы для каждого игрока было определенное количество признаков. Чтобы впоследствии как-то можно было разделять игроков для каждого признака были добавлены специальные префиксы и , которые определяют, что первый префикс соответствуют игроку команды сил Тьмы с номером , а второй соответствует игроку команды сил Света с каким-то номером .

В течение игры каждый из игроков покупает себе определенные предметы. Данная статистика также очень важна при анализе игр. Соответствующая функция, которая обрабатывала покупки для каждого из персонажей представлена в приложении (Приложение Б). У каждого предмета есть свой определенный тип, а именно один из следующий: расходники, атрибуты, снаряжение, аксессуары, магия, оружие, поддержка, броня и артефакты. У каждого предмета также есть соответствующая стоимость. Вся эта информация была обработана и добавлена в результирующий набор данных

У каждой команды на половине находится по 18 строений: вышки, бараки и трон. Данная информация также довольна важна, так как падение вышки сильно ударяет по стратегической игре команды. С каждой упавшей у врага башни союзные крипы противоположной команды становятся сильнее, таким образом они смогут сломать следующие вышку немного быстрее и таким образом наращивать преимущество по игре. Функция, которая обрабатывает статистику для башен и создает соответствующий набор данных представлена в приложении (Приложение В).

Также были собраны данные по количеству убийств Рошана – по командам и отдельно для каждого из игроков в случае, если игрок после убийства Рошана забирал выпавший из него аегис себе.

Также, для каждого из игроков собиралась такая статистика как количество убийств и смертей, количество помощи, количество подобранных рун и поставленных вардов в течение игры.

## Описание признаков

Подробнее рассмотрим данные, которые были получены с ресурса OpenDota. Ниже расположена таблица с основными признаками, а также соответствующее описание для каждого из признаков.

Таблица 1 – Основные признаки для каждого матча

|  |  |
| --- | --- |
| Наименование признака | Описание |
| match\_id | Идентификатор матча |
| time | Время |
| radiant\_win | Победа сил света |

Данные по каждому матчу поделены на временные промежутки по 60 секунд каждый. Среди основных признаков, характерных для каждого из матчей, можно выделить уникальный идентификатор для каждого матча, а также статус победы команды сил Света ().

Таблица 2 – Основные признаки для игроков

|  |  |
| --- | --- |
| Наименование признака | Описание |
| plr{number}\_account\_id | ID аккаунта игрока |
| plr{number}\_xp\_t | Количество опыта на текущий момент |
| plr{number}\_gold\_t | Количество золота на текущий момент |
| plr{number}\_lh\_t | Количество добитых крипов |
| plr{number}\_dn\_t | Количество денаев |
| plr{number}\_{item\_type}\_cost | Стоимость предмета определенного типа |
| plr{number}\_kills\_log | Статистика убийств |
| plr{number}\_buybacks\_log | Статистика выкупов |
| plr{number}\_obs\_log | Статистика поставленных вардов |
| plr{number}\_aegis\_count | Количество подобранных аегисов |
| plr{number}\_mean\_kills | Среднее значение по убийствам |
| plr{number}\_mean\_deaths | Среднее значение по смертям |
| plr{number}\_mean\_assists | Среднее значение по помощам |
| plr{number}\_mean\_gold | Среднее значение по золоту |
| plr{number}\_mean\_last\_hits | Среднее значение по добитым крипам |

В таблице 2 представлены основные признаки для игроков. Каждый из таких признаков присваивается каждому из 10-и игроков в матче. Основными признаками являются количество золота и опыта в минуту, а также количество убийств, так как количество золота влияет на качество предметов – чем больше золота, тем лучшие предметы может позволить себе игрок, следовательно шанс победить у него также возрастает.

Таблица 3 – Основные командные признаки

|  |  |
| --- | --- |
| Наименование признака | Описание |
| {side}\_tower\_{tower\_number} | Статистика по вышкам |
| {side}\_melee | Статистика по баракам |
| {side}\_fort | Статистика по трону |
| {side}\_roshan\_kill | Количество убийств Рошана для команд |

В таблице 3 расположены основные признаки для команд. К командным признакам можно отнести такие признаки как, например, количество не сломанных башен, статус главной крепости на карте – союзного Трона, а также статистика по убийствам Рошана.

Впоследствии были также добавлены уникальные идентификаторы игроков для каждого матча, идентификаторы героев, а также для каждого игрока в каждом матче была добавлена статистика за предыдущие игры: средние значение по основным характеристикам, таким как, например, количество золота и опыта в минуту, количество убийств и смертей за игру, и так далее.

# АНАЛИЗ ДАННЫХ И ВИЗУАЛИЗАЦИЯ

В получившемся наборе данных содержится очень большое количество признаков, а именно порядка 350 штук. Некоторые из них не то, что не важны, в какой-то степени они даже могут портить результаты предсказания будущих моделей. Для того, что понять, какие признаки друг на друга влияют всего, сформируем уменьшенный набор данных, состоящий из информационных данных для команд. То есть необходимо суммировать статистики для игроков. Далее, на рисунке 4 представлена тепловая карта коэффициентов корреляции для сгенерированного набора данных по командам.

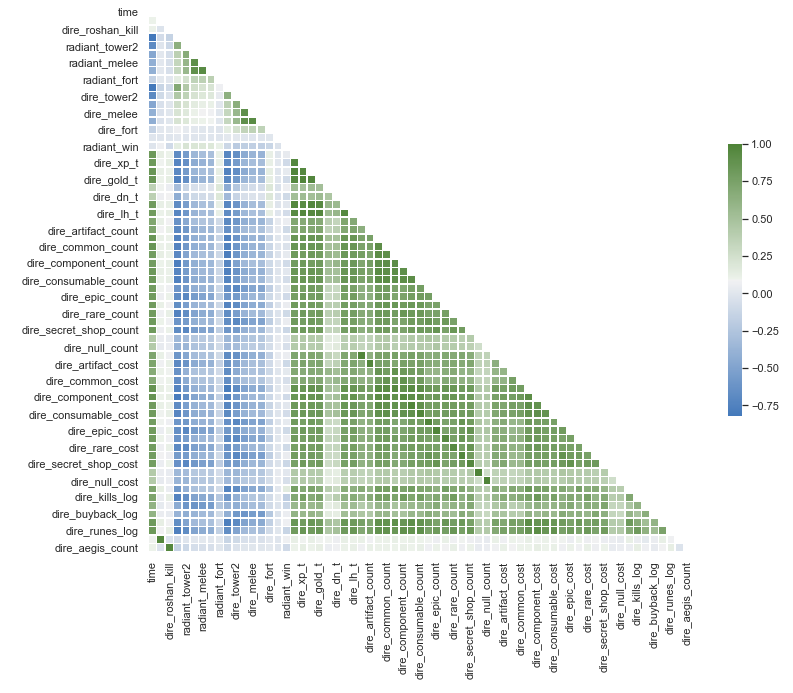


Рисунок 4 – Тепловая карта коэффициентов корреляции для набора данных по командам

На представленном рисунке видно, что с течением времени в основном увеличиваются все показатели, за исключением построек, так как их показатели уменьшаются из-за того, что постройки постепенно ломают. Также, в связи с тем, что за убийство крипов и героев игрок получается некоторое количество золота и опыта, то соответственно все показатели, характеризующие данные признаки, увеличиваются.

Исследуем, какая сторона в среднем побеждает чаще другой. Для этого был построен следующий график, который представлен на рисунке 5.

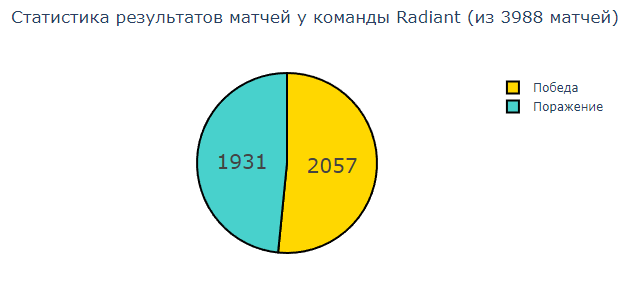


Рисунок 5 – Статистика побед команды сил Света

Как видно из рисунка 5, силы Света побеждали чаще сил Тьмы. На самом деле, если бы мы взяли большую выборку из матчей (а не 4 тысячи), скорее всего показатель приблизился бы к 50%, так как баланс в доте является неотъемлемой частью игры и разработчики к нему стремятся.

Чтобы посмотреть какие персонажи выигрывают чаще всего, а какие чаще всего проигрывают, для каждого матча были добавлены соответствующие идентификаторы для каждого из игроков – номер персонажа игры, на котором он играл. Благодаря этому получилось составить список из четырех персонажей, на которых чаще всего выигрывали матчи, а также список из четырех персонажи, на которых чаще все проигрывали. Список из 4 героев по победам представлен на рисунке 6.

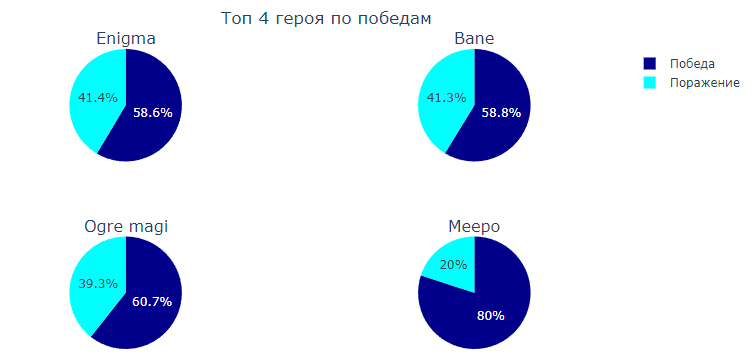


Рисунок 6 – Герои, которые побеждали чаще всего

Из рисунка можно видеть, что самый большой показатель побед у героя под названием . Хоть этого героя и взяли всего в 10 матчах, но зато в 8 матчах команды одержали с ним победу. На втором месте располагается персонаж под названием – его показатели равны 122 играм и 74 победам соответственно. На рисунке 7 представлены герои, которые проигрывали чаще других.

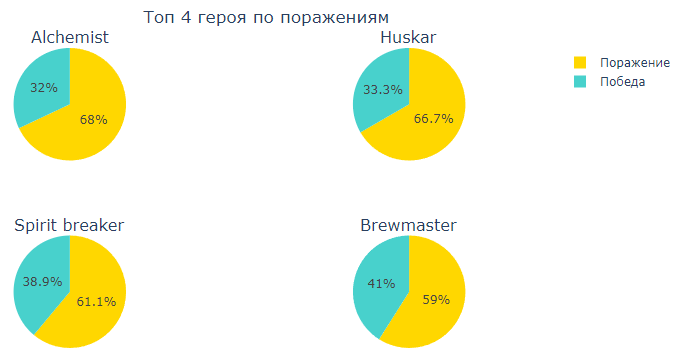


Рисунок 7 – Герои с наихудшим показателем по победам

Из рисунка выше можно заметить, что самым неудачливым героем является – его взяли в 50 матчей и только в 16 он смог одержать победу. На втором месте находится герой с ником . Его суммарно взяли в 24 играх и 16 из них по итогу были проиграны.

Рассмотрим пример матча между командами и , который был сыгран в июне 2021 года. График, отражающий опыт и золото команды сил Света (в данном случае команды ) представлен на рисунке 8.

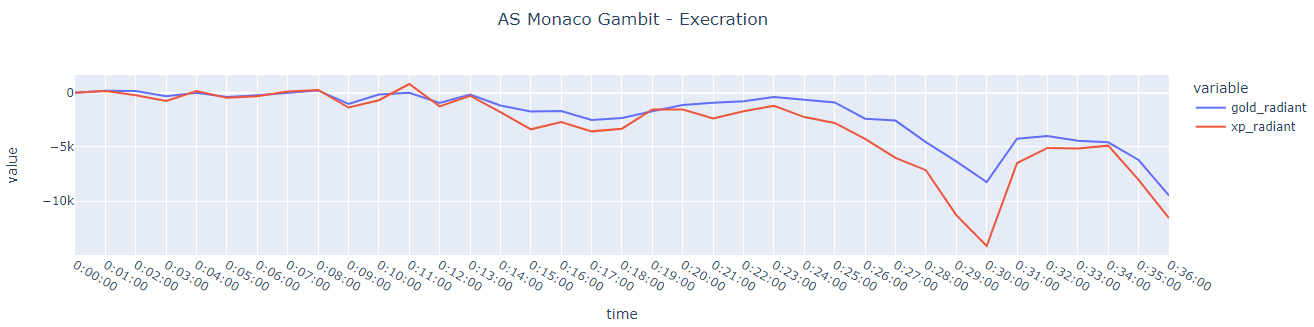


Рисунок 8 – Графики основных показателей у команды сил Света

На графике можно наблюдать заметный спад как по золоту, так и по опыту, начиная с 14 минуты. В общей сложности к концу матча команда сил Света проигрывала примерно 12 тысяч по золоту и примерно 10 тысяч по опыту, что на самом деле не является таким уж большим отрывом. Примерно на 30-й минуте матча команда сил Света смогла немного отыграться благодаря драке, которая, очевидно, пришлась им на пользу. Стоит также отметить, что команда сил Тьмы играла героями, которые заметно проигрывают по статистики побед соответствующим героям из команды сил Света. Данная статистика представлена на рисунке ниже.



Рисунок 9 – Статистика по героям

То есть можно заметить, что основной персонаж у команды сил Тьмы, а именно персонаж со статистикой побед, равной 49%, заметно проигрывает по статистике основному персонаже из противоположной команды. Таким образом можно предположить, что команда просто опытнее своего соперника. Для сравнения ниже представлена статистика по добитым крипам у команды сил Света.

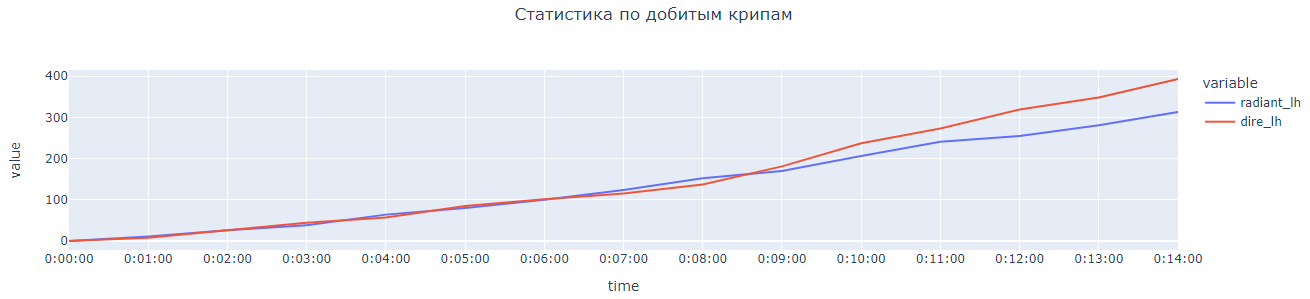


Рисунок 10 – Статистика по добитым крипам для команды сил Света

Как правило, статистика по добитым крипам имеет смысл в самом начале игры – примерно до 15–20 минуты. Как видно из графика на рисунке 10, обе команды имели примерно одинаковое количество добитых крипов, что естественно говорит о равной игре в самом начале матча.

Рассмотрим еще один матч, но теперь между другими командами, а именно – между и . График по основным показателям для данного матча представлен на рисунке 11.

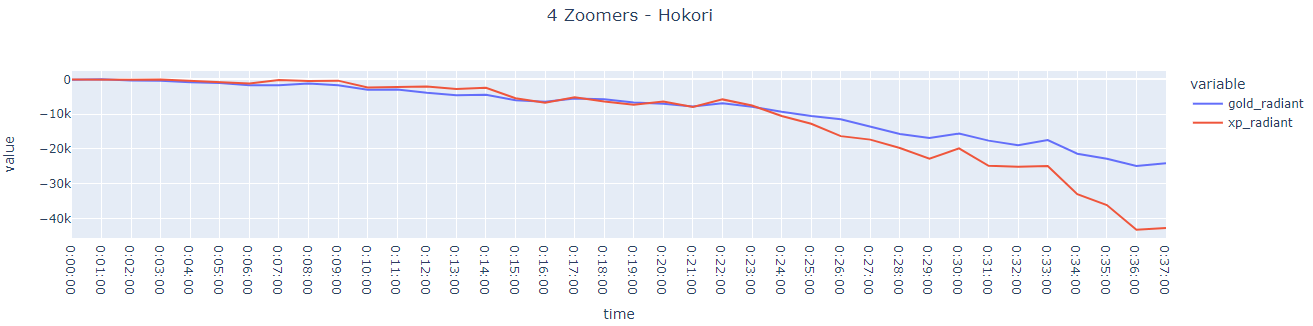


Рисунок 11 – График с основными показателями для команды

Из графика снова видно, что примерно до 10 минуты обе команды имели примерно одинаковое количество ресурсов, но после определенного этапа команды сил Света начала постепенно проигрывать по обоим показателям. По итогу команды сил Тьмы одерживают победу с преимуществом в 40 тысяч по золоту и около 25 тысяч по опыту, что намного больше, чем в предыдущем матче. Это может говорить о почти что разгромном поражении. Сравним показатели по добитым крипам для основных персонажей обоих команд. Результаты сравнения представлены на рисунке 12.

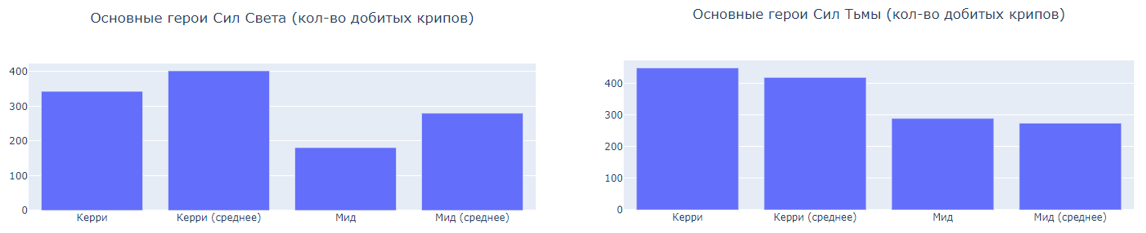


Рисунок 12 – Статистика по добитым крипам у основных персонажей

Как можно заметить из графиков, силы Света заметно проигрывают соответствующим персонажам из команды сил Тьмы. Так, керри команды за матч добил примерно 330 крипов в то время, как его оппонент больше 425. Хотя в среднем за предыдущие матчи он добивал крипов больше. Примерно такая же ситуация обстоит с мидером команды . В рассматриваемом матче он заметно отстал от своего врага, но за предыдущие матчи его показатели по добитым крипам превосходят показатели соперника.

ЗАКЛЮЧЕНИЕ

По итогу курсовой работы были получены и обработаны необходимые данные для обучения соответствующих моделей машинного обучения. Было подробно разобрано как именно данные извлекались из сервиса OpenDota, как они обрабатывались и как по итогу был сформирован конечный набор данных, содержащий информацию о конкретных матчах в конкретный промежуток времени.

После формирования необходимого набора данных был проведен анализ. С помощью тепловой карты корреляций было продемонстрировано какие признаки влияют друг на другу больше всего, а какие меньше. Затем были определены самые успешные герои по количеству победам. В конце были разобраны два матча, а именно – сравнение основных показателей у двух различных команд.

Как уже было отмечено, с помощью отобранных данных решается задача предсказания победителя матча в различные промежутки времени. Модели, которые будут обладать высокой точностью прогнозов, могут быть применены в других киберспортивных дисциплинах для решения схожих задач. Например, помимо игры Dota 2 в жанре MOBA также очень популярны такие киберспортивные дисциплины, как StarCraft 2 и League of Legends. Можно ожидать, что модели, которые будут использоваться для предсказания победителя матча в игре Dota 2, смогут давать схожий результат в других играх подобного жанра.

Помимо данных, которые накапливаются непосредственно в течение матча, можно также дополнить набор информацией об игроках до начала матча, о составе команд, а также анализом игрового чата во время игры и многим другим.

Таким образом, будущие модели, обученные на отобранных данных, могут быть полезным инструментом для анализа как для профессиональных игроков и их тренеров, так и для аналитиков или букмекеров, которые также заинтересованы в глубокой аналитике.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Johansson F., Wikström J., Result Prediction by Mining Replays in Dota 2 / Faculty of Computing – 2015.
2. Mark Lutz. Learning Python, Fourth Edition – O’Reilly / Julie Steele, Rachel Head, Robert Romano, 2017 – Pp. 682–704.
3. Wes McKinney, Python for Data Analysis, O’Reilly, 2020 – Pp. 112 –139.
4. Dota 2 Wikipedia [Электронный ресурс] / - Режим доступа: <https://ru.wikipedia.org/wiki/Dota_2>
5. OpenDota API documentation [Электронный ресурс] / - Режим доступа: <https://www.opendota.com/>
6. Pandas documentation [Электронный ресурс] / - Режим доступа:  
   <https://pandas.pydata.org/>
7. Python 3.9.6 documentation [Электронный ресурс] / - Режим доступа: <https://docs.python.org/3/>
8. SQL запросы для извлечения данных о матчах и игроках

SELECT

matches.match\_id,

match\_patch.patch,

matches.chat,

matches.radiant\_win,

matches.duration,

matches.first\_blood\_time,

matches.radiant\_score,

matches.dire\_score,

matches.objectives,

matches.teamfights,

matches.radiant\_gold\_adv,

matches.radiant\_xp\_adv

FROM matches

JOIN match\_patch using(match\_id)

WHERE TRUE

AND matches.start\_time >= extract(epoch from timestamp '2021-06-01T21:00:00.000Z')

AND matches.start\_time <= extract(epoch from timestamp '2021-10-01T21:00:00.000Z')

ORDER BY matches.match\_id NULLS LAST

LIMIT {limit}

OFFSET {offset}

SELECT

player\_matches.match\_id,

player\_matches.player\_slot,

player\_matches.hero\_id,

player\_matches.firstblood\_claimed,

player\_matches.xp\_t,

player\_matches.gold\_t,

player\_matches.times,

player\_matches.dn\_t,

player\_matches.lh\_t,

player\_matches.purchase\_log,

player\_matches.buyback\_log,

player\_matches.kills\_log,

player\_matches.runes\_log,

player\_matches.obs\_log,

player\_matches.sen\_log

FROM player\_matches

JOIN matches using(match\_id)

WHERE TRUE

AND matches.start\_time >= extract(epoch from timestamp '2021-06-01T21:00:00.000Z')

AND matches.start\_time <= extract(epoch from timestamp '2021-10-01T21:00:00.000Z')

ORDER BY player\_matches.match\_id NULLS LAST

LIMIT {limit}

OFFSET {offset}

1. Функция для обработки данных о предметах у игроков

def process\_player\_purchases(player: dict, match\_duration: int):

def encode\_item(bought\_item: dict, player\_prefix: str):

result = {}

qual\_collection = get\_quality\_collection(items)

count = [f'{player\_prefix}{item}\_count' for item in qual\_collection]

costs = [f'{player\_prefix}{item}\_cost' for item in qual\_collection]

for i in (count + costs):

result[i] = 0

cost = 0 if (not 'cost' in bought\_item) else bought\_item['cost']

qual = 'null' if(not 'qual' in bought\_item) else bought\_item['qual']

result[player\_prefix + qual + '\_count'] = 1

result[player\_prefix + qual + '\_cost'] = cost

return result

# longest\_match\_duration = longest\_match\_duration

purchased\_items = []

purchased\_items\_time = [purchase\_log['time'] for purchase\_log in player['purchase\_log']]

for purchased\_item in player['purchase\_log']:

item\_ = items[purchased\_item['key']]

item\_ = encode\_item(item\_, get\_player\_prefix(player))

purchased\_items.append(item\_)

# all purchases during match

all\_purchases = pd.DataFrame(data=purchased\_items)

all\_purchases['time'] = purchased\_items\_time

# only purchases before match begin

purchases\_before\_begin = all\_purchases[all\_purchases['time'] <= 0].sum()

purchases\_before\_begin['time'] = 0

purchases\_total = pd.DataFrame(purchases\_before\_begin).T

# make dataframe for all purchasing during the match in cumulative sum

for time\_ in range(60, match\_duration + 60, 60):

purchases\_time\_slice = all\_purchases[(time\_ - 60 < all\_purchases['time']) & (all\_purchases['time'] <= time\_)].sum()

purchases\_time\_slice['time'] = time\_

purchases\_total = purchases\_total.append(purchases\_time\_slice, ignore\_index=True)

columns\_ = list(purchases\_total.columns)[:-1]

purchases\_total = pd.concat([purchases\_total[columns\_].cumsum(), purchases\_total['time']], axis=1)

purchases\_total = purchases\_total.set\_index('time')

return purchases\_total.fillna(0)

1. Функция для обработки данных о строениях

def get\_buildings\_data(match: dict):

towers = {

'tower1': 3,

'tower2': 3,

'tower3': 3,

'tower4': 2,

'melee': 3,

'range': 3,

'fort': 1

}

good, bad = towers.copy(), towers.copy()

buildings\_events = [objective\_ for objective\_ in match['objectives'] if objective\_['type'] == 'building\_kill']

buildings\_objectives = pd.DataFrame(buildings\_events)

data\_for\_df = []

match\_duration = match['duration']

tower\_location = buildings\_objectives['key'].values

event\_time = buildings\_objectives['time'].values

result\_ = []

for time\_ in range(0, match\_duration + 60, 60):

for location in tower\_location:

event\_time = \

buildings\_objectives.loc[buildings\_objectives['key'] == location]['time'].values[0]

if time\_ - 60 < event\_time <= time\_:

key = location.split('\_')

tower\_name = key[3]

if(key[2] == 'goodguys'):

good[tower\_name] -= 1

else:

bad[tower\_name] -= 1

row\_for\_df = [time\_, \*[value for value in good.values()], \*[value for value in bad.values()]]

data\_for\_df.append(row\_for\_df)

buildings\_df\_columns = ['time', \*[f'radiant\_{key}' for key in good.keys()], \*[f'dire\_{key}' for key in bad.keys()]]

buildings\_dataframe = pd.DataFrame(data\_for\_df, columns=buildings\_df\_columns).set\_index('time')

return buildings\_dataframe