Министерство образования и науки Российской Федерации Федеральное агентство по образованию

Государственное образовательное учреждение высшего профессионального образования

«Алтайский государственный технический университет им. И. И. Ползунова»

Факультет (институт) ФИТ

Кафедра Прикладная математика

наименование кафедры

Отчет защищен с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель от вуза Н.Д. Бубнова

(подпись) (и.о, фамилия)

“\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_ 2018 г.

дата

Отчет

по Преддипломной практике

название вида практики

в АлтГТУ им. И. И. Ползунова

название предприятия, организации, учреждения

**ПП 09.03.04.11.000 О**

(обозначение документа)

Студент группы ПИ-41 С.Ю. Искуснов

и.о., фамилия

Руководитель практики

от предприятия технический директор ООО «Энтера-Софт» А. В. Тамплон

должность, ученое завние и.о., фамилия

Руководитель практики

от вуза старший преподаватель Н. Д. Бубнова

должность, ученое звание и.о., фамилия

БАРНАУЛ 2018

ФГБОУ ВО «Алтайский государственный технический университет им. И. И. Ползунова»

Кафедра Прикладная математика

**Индивидуальное задание**

на Преддипломную практику

(вид, тип и содержательная характеристика практики по УП)

студенту 4 \_ курса Искуснову Сергею Юрьевичу группы ПИ-41

(Ф.И.О.)

Профильная организация ООО «Энтера-Софт»

(наименование)

Сроки практики

(по приказу АлтГТУ)

Тема Разработка системы искусственного интеллекта для игры в покер

**Рабочий график (план) проведения практики:**

|  |  |  |  |
| --- | --- | --- | --- |
| №  п/п | Содержание раздела (этапа) практики | Сроки выполнения | Планируемые результаты практики |
| 1 | Изучение предметной области | 21.05.2018 – 23.05.2018 | Усвоение правил и особенностей игры в покер |
| 2 | Изучение стратегий игры в покер и существующих алгоритмов ИИ | 24.05.2018 – 27.05.2018 | Усвоение принципов создания ИИ для игры в покер |
| 3 | Создание алгоритма для покерного бота и проектирование приложения | 28.05.2018 – 30.05.2018 | Готовый алгоритм поведения ИИ и общая архитектура приложения |
| 4 | Написание отчета | 31.05.2018 – 1.06.2018 | Готовый отчет |

Руководитель практики от вуза Бубнова Н.Д., старший преподаватель

(подпись) (Ф.И.О., должность)

Руководитель практики

от профильной организации Тамлон А.В., технический директор

(подпись) (Ф.И.О., должность)

Задание принял к исполнению Искуснов С.Ю.

(подпись) (Ф.И.О)

Содержание

[Введение 4](#_Toc515535874)

[1 Постановка задачи и обзор существующих систем 6](#_Toc515535875)

[1.1 Описание предметной области 6](#_Toc515535876)

[1.2 Покерная математика 10](#_Toc515535877)

[1.3 Обзор существующих систем 15](#_Toc515535878)

[1.4 Постановка задачи 18](#_Toc515535879)

[2 Проектирование приложения 20](#_Toc515535880)

[2.1 Логика покерного бота 20](#_Toc515535881)

[2.2 Общая архитектура приложения 24](#_Toc515535882)

[Заключение 26](#_Toc515535883)

[Список литературы 27](#_Toc515535884)

# Введение

Покер как карточная игра существует более 450 лет. Зародился он в Европе: в Испании, Франции, Италии. Правила покера тогда были несколько иными. С течением времени правила покера менялись. Первые письменные упоминания о современном варианте покера появляются в 1829 году в мемуарах артиста Джо Кауэла. В 1834 году в покер стали играть колодой из 52 карт. Однако, как бы ни менялись правила, победитель определялся по наличию у него комбинаций[8].

Однако за последние лет 20 популярность покера сильно возросла. И в общем то очевидно, что этот рост, не в последнюю очередь, связан с ростом популярности и количества онлайн сервисов, предоставляющих возможность играть в азартные игры (и в покер в том числе). Правда интерес в разработка алгоритмов для решения задачи игры в покер связан не только с этим.

Дело в том, что с точки зрения разработки искусственного интеллекта, покер представляет собой наглядный пример игр с неполной информацией. Игры с асимметрией и неполнотой информации требуют значительно более сложных подходов к принятию решений по сравнению с аналогичными по размерам играми с полной информацией, доступной в любое время (таких как шашки, нарды, шахматы и так далее). Оптимальное решение в любой момент времени зависит от знания стратегии противников, зависящей от скрытой для нас и доступной только им информации, оценить которую можно только по их прошлым действия. Однако и их предыдущие действия тоже зависят от скрытой от них информации о наших действиях и того, как наши действия эту информацию раскрывали. Этот рекурсивный процесс показывает основную сложность в построении эффективных алгоритмов принятия решений. А решение сложных не тривиальных задач – интересное занятие.

Из всего вышесказанного можно выделить несколько причин интереса к разработки искусственного интеллекта для игры в покер (покерного бота):

* Академический интерес. Он как раз связан с решением задач с неполной информацией.
* Финансовый интерес. Этот момент связан с использованием покерных ботов для зарабатывания денег. Если создать достаточного эффективного бота, то его можно запустить в какой-нибудь покер рум. Там он будет успешно зарабатывать деньги, пока создатель занимается своими делами.
* «Защитный» интерес. Этот пункт вытекает из второго. Покер румы совершенно не заинтересованы в том, чтобы на их сайтах играли боты, а не люди. В связи с этим для них встает вопрос изучения покерных ботов с целью отлова и бана. Хотя в данной плоскости изучаются скорее паттерны поведения ботов, нежели сложные алгоритмы принятия решений. В настоящее время наиболее популярные покер румы довольно успешно борется с ликвидацией ботов на своих просторах.
* И наконец игровой интерес. Создание умного ИИ для того, чтобы с ним было интересно играть. Или для обучения новичков. Конечно обучаться можно и с живыми людьми, но с людьми игра как правило идет на деньги, которые новичку не хочется лишний раз терять. К тому же обучаться с ботом удобнее, для этого даже интернет не всегда обязателен (в сингловых видеоиграх например).

# 1 Постановка задачи и обзор существующих систем

## Описание предметной области

Покер – карточная игра, цель которой – выиграть ставки, собрав как можно более высокую покерную комбинацию, используя 4 (старый классический вариант), 2, или 5 карт, или вынудив всех соперников прекратить участвовать в игре. Игра идёт с полностью или частично закрытыми картами. Конкретные правила могут варьироваться в зависимости от разновидности покера. Обобщающими элементами всех разновидностей покера являются комбинации и наличие торговли в процессе игры[8].

Далее мы будем рассматривать Техасский Холдем.

В игре может участвовать от 2 до 10 человек. Каждому игроку в Техасском Холдеме раздается по две карты. Эти карты раздаются рубашкой вверх, поэтому игроки знают только значения своих карт[9].

После раздачи начинается торговля, которая проходит в несколько раундов.

Два игрока по левую руку от дилера делают обязательные ставки, которые взимаются еще до начала торгов. Это делается для того, чтобы стимулировать игроков на активную игру. Эти обязательные ставки называются блайндами. Таковы правила игры в покер.

После того, как игроки поставили блайнды, начинается первый этап торговли.

Правила игры в покер подразумевают наличие определенных действий, которые игрок совершает в процессе торговли:

* Поставить, бет (англ. bet) – сделать ставку
* Ответить, колл (англ. call) – поставить столько же, сколько поставил соперник – уравнять
* Поднять, рейз (англ. raise) – увеличить ставку – поставить больше, чем соперники
* Cбросить карты, фолд (англ. fold) – отказаться от дальнейшего участия в игре и сбросить карты
* Пропустить, чек (англ. check) – в ситуациях, когда ставка уже была сделана или ставки не были сделаны соперниками – не добавлять ничего в банк, оставить «как есть»

Круг торговли заканчивается тогда, когда все игроки сделали равные ставки или сбросили карты.

После первого круга торговли, если в раздаче остается больше одного человека, то, по правилам игры в покер, на стол кладутся три общие открытые карты, которые называют флоп. Общие карты нужно использовать для составления комбинаций.

Если и после этого раунда в раздаче остается больше одного человека, то кладут еще одну общую карту, которую называют терном. Аналогично флопу, после сдачи терна проводится еще один круг торговли.

После терна, если это необходимо, кладут последнюю общую карту – ривер.

После ривера следует еще один раунд торговли, и если после него на банк претендует два или более игроков, то происходит вскрытие.

Комбинации по правилам покера составляются из пяти общих карт и двух закрытых. Когда была осуществлена и уравнена последняя ставка, все оставшиеся в игре люди по очереди начинают открывать для противников свои карты. Из них составляются и оцениваются итоговые выигрышные комбинации.

Теперь разберемся с комбинациями.

В покере существует 10 возможных комбинаций карт. Расположим их по старшинству, от самой старшей к самой младшей:

1. **«Роял Стрит Флеш»** или просто **«Роял Флеш»** – 5 самых старших одномастных карт.
2. **«Стрит Флеш»** – 5 карт одной масти по порядку.
3. **«Каре»** – 4 карты одного ранга.
4. **«Фулл Хаус»** – комбинация, включающая в себя «Пару» и «Тройку» одновременно.
5. **«Флеш»** – 5 одномастных карт.
6. **«Стрит»** – 5 собранных по порядку карт любой масти.
7. **«Сет»** или **«Тройка»** – 3 карты одного ранга.
8. **«Две пары»** – 4 карты, среди которых собраны по 2 одинаковых по рангу.
9. **«Пара»** – это 2 одинаковые карты.
10. Самая младшая комбинация **«Старшая карта»** – это 1 карта. Чем она выше по рангу, тем вероятнее ее победа.

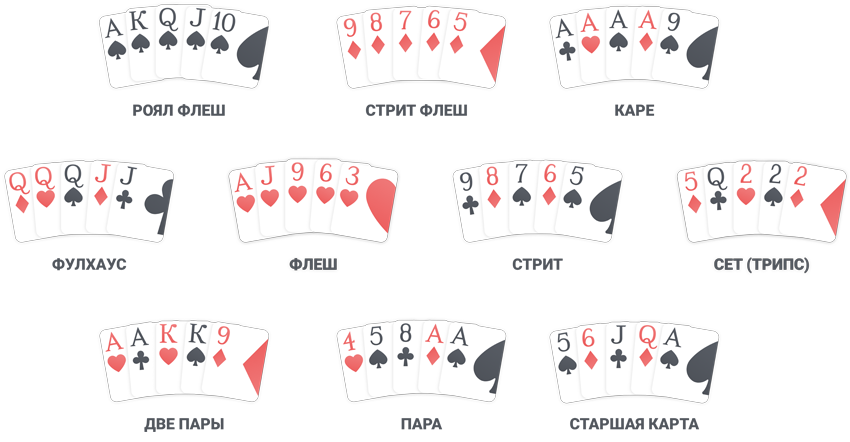


Рисунок 1.1 - Покерные комбинации

В случае если комбинации совпадают, то результат определяется по старшинству карт в этих комбинациях. К примеру у двух игроков выпала пара, но у одного пара королей, а у второго пара дам. Выигрывает тот, у кого пара королей, поскольку «Король» старше «Дамы». Если же у обоих игроков оказались одинаковые пары (пара дам и пара дам), то победитель определяется оп кикеру, то есть старшей карте из оставшихся у игрока.

Думаю стоит указать старшинство карт. Расположим карты по старшинству от самой старшей к самой младшей:

1. Туз (А)
2. Король (К)
3. Дама (Q, Д)
4. Валет (J, В)
5. Десять (10)
6. Девять (9)
7. Восемь (8)
8. Семь (7)
9. Шесть (6)
10. Пять (5)
11. Четыре (4)
12. Три (3)
13. Два (2)

Все вышеописанные правила относятся к непосредственной раздаче. Необходимо объяснить, что представляет из себя игра (в смысле игровая сессия) в целом. У каждого игрока есть стек, т.е. денежная сумма или фишки, из которого он делает ставки. Делая ставку в раздаче, игроки отдают свои деньги или фишки из стека в банк. Победитель раздачи забирает банк себе и прибавляет выигрыш к своему стеку. Если стек пустеет (или сумма в стеке становится недостаточной для ставки), игрок выбывает. Игра заканчивается, когда остался один игрок с непустым стеком.

Разберемся в чем различие лимитного и безлимитного Холдема.

Лимитный Холдем отличается фиксированным размером ставок, и ограниченным числом повышений в каждом раунде торговли (не более четырех). В объявлении типа игры (например, Limit $1/$2) указываются размеры ставок в первых двух ($1) и в последующих двух ($2) раундах торговли. Большой блайнд в такой игре будет равен $1, а малый - половине большого ($0,5). Таким образом на префлопе и на флопе можно сделать не более 4 повышений на $1 (для приведенного примера), т.е. каждый может поставить не более $4. А на терне и ривере - не более 4 повышений по $2, т.е. всего $8. В безлимитном Холдеме соответственно ограничения на ставки отсутствуют. Безлимитный холдем отличается большими игровыми рисками по сравнению с лимитным. Ставки имеют фиксированный предел, поэтому вы можете без труда наперед провести расчет собственных расходов на просмотр терна, флопа и ривера. Другими словами, в лимитном холдеме по сравнению с безлимитным имеется возможность разыгрывать больше рук.

Поговорим об эффективной игре в покер.

Покер, в отличии от шахмат например, является игрой завязанной на случайности. Невозможно предугадать какие карты придут тебе в руку или какие карты окажутся на столе. Поэтому невозможно выигрывать каждую раздачу. Отсюда становится очевидно, что эффективная игра в покер представляет из себя оценку рисков и управление стеком. Каждый игрок в зависимости от своей руки оценивает шанс на победу в раздаче. Если шансы на победу низкие, то игрок делает пас и сохраняет сумму в своем стеке. Если шансы высокие, то максимально повышает ставку, рискуя при этом потерять большую сумму, в случае неудачи. Такая простая логика добавляет в игру еще одну особенность – блеф, когда игрок повышает ставку, сознательно понимая, что его шансы на победу низки. Но поскольку противникам значения карт игрока неизвестны, стратегия блефа вполне может сработать.

## Покерная математика

Первый шаг в изучении покерной математики – научиться высчитывать “ауты”.

«Ауты» – это карты в колоде, которые могут дать вам выигрышную руку. К ним относятся карты, которые могут выпасть на стол, собирая или улучшая вашу комбинацию[1].

К примеру у вас в руке туз пик и король пик. На флопе приходит пятерка пик, валет треф и бубновая дама. Любая Десятка дает вам стрит и, предположительно, лучшую руку. Если выпадут Король или Туз, у вас будет топ пара. Таким образом, эти карты тоже могут считаться аутами. В колоде предположительно есть 3 туза, 3 короля и 4 десятки. В сумме у вас есть 10 аутов.

Сдается терн на столе оказывается двойка пик. Теперь на столе лежат пятерка пик, валет треф, бубновая дама и двойка пик. У нас все еще нет готовой комбинации, но если выпадет еще одна пика, то мы получим флеш. А значит количество аутов увеличилось. Имеем все те же 3 туза, 3 короля и 4 десятки. Осталось посчитать сколько пик возможно в колоде. Тройка, четверка, шестерка, семерка, восьмерка, девятка, десятка, валет и дама – всего 9. Но десятку не нужно учитывать дважды (она нужна для двух комбинаций). В итоге мы получаем 18 аутов.

На основе количества возможных аутов можно вычислить вероятность удачного исхода для нас. Здесь все просто: количество аутов нужно поделить на количество карт в колоде. Всего в колоде пятьдесят две карты. Две у нас на руках, 3 на столе, соответственно в колоде 47 карт. Конечно в реальности карт в колоде будет меньше, так как у ваших соперников тоже есть каты на руках. Но поскольку нам карты соперника не известны, можно считать, что следующей выпадет любая из неизвестных нам карт с равной вероятностью. Ниже приведена таблица 1.1 с посчитанными вероятностями. Это достаточно популярная таблица и ее легко найти в сети по соответствующему запросу.

Теперь рассмотри ситуацию раннер-раннер. Раннер-раннер – это альтернативное название бэкдорного дро в покере. Термин раннер-раннер используется для обозначения дро комбинаций, для успешного завершения которых игроку необходимо получить две определенные карты на терне и на ривере. Дро комбинация – это недостроенная комбинация, которой не хватает одной или двух карт для усиления. То есть суть ситуации в том, что нам последовательно на терне и ривере должны выпасть нужные нам карты. Для того чтобы получить общую вероятность нужного нам события, мы должны перемножить вероятности этих двух событий и поделить на сто.

Например у нас в руке две девятки, на флопе десятка, двойка и шестерка. Мы хотим получить стрит. Для этого необходимо выпадение семерки и восьмерки. В колоде четыре семерки и четыре восьмерки, значит на терне у нас восемь аутов, а на ривере четыре. Теперь считаем: 17,02\*8,7/100=1,48. Это значит, что наш шанс получить стрит равен 1,48 процента.

Таблица 1.1 – Вероятности успешного выпадения аутов

|  |  |  |  |
| --- | --- | --- | --- |
|  | Приходящие карты | | |
| Ауты | 1 карта | 1 карта | 2 карты |
| Терн | Ривер | Терн и ривер |
| 1 | 2,13% | 2,17% | 4,26% |
| 2 | 4,26% | 4,35% | 8,42% |
| 3 | 6,38% | 6,52% | 12,49% |
| 4 | 8,51% | 8,70% | 16,47% |
| 5 | 10,64% | 10,87% | 20,35% |
| 6 | 12,77% | 13,04% | 24,14% |
| 7 | 14,89% | 15,22% | 27,84% |
| 8 | 17,02% | 17,39% | 31,45% |
| 9 | 19,15% | 19,57% | 34,97% |
| 10 | 21,23% | 21,47% | 38,39% |
| 11 | 23,40% | 23,91% | 41,72% |
| 12 | 25,53% | 26,09% | 44,96% |
| 13 | 27,66% | 28,26% | 48,10% |
| 14 | 29,79% | 30,43% | 51,16% |
| 15 | 31,91% | 32,61% | 54,12% |
| 16 | 34,04% | 34,76% | 56,98% |
| 17 | 36,17% | 36,96% | 59,76% |
| 18 | 38,30% | 39,13% | 62,44% |
| 19 | 40,43% | 41,30% | 65,03% |
| 20 | 42,55% | 43,48% | 67,53% |
| 21 | 44,68% | 45,65% | 69,94% |

Займемся расчетом шансов банка. Когда мы определяем шансы банка, мы хотим узнать сколько денег мы можем выиграть и какую сумму нам для этого нужно поставить. Фактически это показатель, во сколько раз банк должен превосходить нашу ставку, чтобы ставка была оправданной. Вычисляются шансы банка по следующей формуле:

X – количество карт в колоде;

Y – количество аутов;

Z – шансы банка.

Записывается как Z к 1. Это означает, что банк должен быть в Z раз больше чем ставка, чтобы она была оправданной. Ниже приведена таблица 1.2 с посчитанными шансами банка. Эту таблицу также можно найти в сети.

Таблица 1.2 - Шансы банка

|  |  |  |  |
| --- | --- | --- | --- |
|  | Приходящие карты | | |
| Ауты | 1 карта | 1 карта | 2 карты |
| Терн | Ривер | Терн и ривер |
| 1 | 46,00 к 1 | 45,00 к 1 | 22,50 к 1 |
| 2 | 22,50 к 1 | 22,00 к 1 | 10,88 к 1 |
| 3 | 14,67 к 1 | 14,33 к 1 | 7,01 к 1 |
| 4 | 10,75 к 1 | 10,50 к 1 | 5,07 к 1 |
| 5 | 8,40 к 1 | 8,20 к 1 | 3,91 к 1 |
| 6 | 6,83 к 1 | 6,67 к 1 | 3,14 к 1 |
| 7 | 5,71 к 1 | 5,57 к 1 | 2,59 к 1 |
| 8 | 4,88 к 1 | 4,75 к 1 | 2,18 к 1 |
| 9 | 4,22 к 1 | 4,11 к 1 | 1,86 к 1 |
| 10 | 3,70 к 1 | 3,60 к 1 | 1,60 к 1 |
| 11 | 3,27 к 1 | 3,18 к 1 | 1,40 к 1 |
| 12 | 2,92 к 1 | 2,83 к 1 | 1,22 к 1 |
| 13 | 2,62 к 1 | 2,54 к 1 | 1,08 к 1 |
| 14 | 2,36 к 1 | 2,29 к 1 | 0,95 к 1 |
| 15 | 2,13 к 1 | 2,07 к 1 | 0,85 к 1 |
| 16 | 1,94 к 1 | 1,88 к 1 | 0,75 к 1 |
| 17 | 1,76 к 1 | 1,71 к 1 | 0,67 к 1 |
| 18 | 1,61 к 1 | 1,56 к 1 | 0,60 к 1 |
| 19 | 1,47 к 1 | 1,42 к 1 | 0,54 к 1 |
| 20 | 1,35 к 1 | 1,30 к 1 | 0,48 к 1 |
| 21 | 1,24 к 1 | 1,19 к 1 | 0,43 к 1 |

В реальности для расчетов можно пользоваться только первой таблицей. Для этого нужно рассчитать ставку в процентах по формуле:

, где

N – размер вашей ставки;

M – размер банка;

P – ставка процентах.

В этом случае решение о ставке принимается по следующему принципу: если вероятность выпадения комбинации больше, чем ставка в процентах, значит колл оправдан; если вероятность выпадения комбинации меньше, чем ставка в процентах, значит колл не оправдан.

Теперь рассмотрим математическое ожидание в рамках покера.

Математическое ожидание (англ. Expected Value) - в покере, средняя выгода от того или иного решения при условии, что подобное решение может быть рассмотрено в рамках теории больших чисел и длительной дистанции.

+EV игра - это игра с положительным математическим ожиданием, которая в долгосрочной перспективе будет приносить вам деньги (т.е. вы будете выигрывать при принятии +EV решений).

-EV игра - это игра с отрицательным математическим ожиданием. Играя таким образом в долгосрочной перспективе вы будете «проигрывать» свои деньги (т.е. вы будете проигрывать при принятии -EV решений).

EV рассчитывается по следующей формуле[3]:

Pi – вероятность события;

Vi –сумма выигрыша или проигрыша;

EV – математическое ожидание (или ожидаемая выгода).

В простейшем случае эта формула преобразуется к виду:

P – вероятность успешного выпадения нужных аутов;

А – сумма ставки;

B – сумма банка;

EV – ожидаемая выгода.

Использование математического ожидания позволяет оценивать не только коллов, но и других действий (фолдов и райзов).

Приведем пример. В нашей руке туз пик и двойка пик, на столе дама пик, тройка пик, червовый король и бубновая семерка. У нас есть возможность собрать флеш. Размер банка составляет $100 и оппонент ставит $50. Нам необходимо сделать колл, чтобы получить шанс выиграть банк в размере $150. Скажем нам для победы нужно обязательно получить флеш. Для этого у нас есть 9 аутов, значит шанс победы примерно 20%. Тогда считаем EV=150\*0,2-50\*0,8=-10. Получаем отрицательное математическое ожидание, следовательно нам не выгодно делать этот колл.

## Обзор существующих систем

Рассмотрим две наиболее передовые системы искусственного интеллекта для игры в покер – Сepheus и DeepStack.

Программа Cepheus была создана группой разработчиков из университета Альберты. Это первая программа, которая практически совершенно играет в покер. Правда играет она только в лимитный Техасский Холдем один на один.

Работа над проектом продолжалась 11 лет. Прорыв состоялся после подключения соавтора из Финляндии Оскари Таммелина, более чем на порядок улучшившего эффективность старого алгоритма канадцев. Кодирование новой программы началось в октябре 2013 года. Компьютерные мощности для расчета предоставил исследовательский консорциум Calcul Québec[6].

Cepheus – это самообучающаяся система. Причем обучение шло без участия человека вообще. Программа решала холдем полным перебором: играла множество раздач сама с собой, принимая случайные решения. При этом вероятность принятия неудачных решений понижалась. Программа училась играть два месяца, используя более четырех тысяч процессоров, каждый из которых отыгрывал более шести миллиардов раздач каждую секунду. В рамках подготовки было сыграно больше раздач, чем за всю историю человечества. Конечно же программа выигрывает не абсолютно все партии, это невозможно. Но на дистанции показывает себя практически идеально.

Вот только стоит количеству игроков возрасти на 1, задача вновь становится неподъемной. Надо понимать, что Cepheus – это не математическое решение лимитного Холдема. Поэтому возможно нас еще ожидает система способная решать задачу лимитного Техасского Холдема, потребляя хотя бы меньшее количество мощностей.

Алгоритм DeepStack также был создан группой разработчиков из университета Альберты, под руководством профессора информатики Майкла Боулинга. Этот алгоритм предназначен для большого класса последовательных игр с неполной информацией. В том числе для игры в безлимитный Техасский Холдем. Опишем принцип его работы.

ИИ переоценивает свои действия на каждом этапе, когда от него требуется принятие решения. Для расчёта полезности каждой ставки используется дерево предвидения (lookahead tree), значения для поддеревьев которого вычисляются с использованием нейросети, заранее обученной на случайных игровых ситуациях[7]. Принцип работы дерева решений показан на рисунке 1.2[5]. Красные и бирюзовые узлы – действия игроков. Зелёные – карты на столе. Листья с фишками представляют собой конец игры, где выигрыш может быть определен если ИИ сбросил или определил возможные руки игрока с учетом предыдущих действий. Общая архитектура алгоритма изображена на рисунке 1.3[5]. (a) DeepStack производит повторное решение в каждом открытом состоянии, в котором он должен действовать, используя lookahead дерево, где значения поддерева вычисляются с использованием нейронной сети (b), обученной перед игрой на случайно созданных покерных ситуациях (c).

Структура нейросети демонстрирует, что на входе подаётся размер банка, открытые карты и диапазоны игроков (возможные комбинации, с которыми игрок мог войти в игру таким образом, каким он в неё вошёл (колл, рейз и т.д.), вероятность каждой комбинации). Нейросеть состоит из семи полностью соединённых скрытых слоёв. Выходные значения затем обрабатываются другой нейросетью, которая проверяет, что действия удовлетворяют ограничению на нулевую сумму[7]. Нейросеть изображена на рисунке 1.4[5].

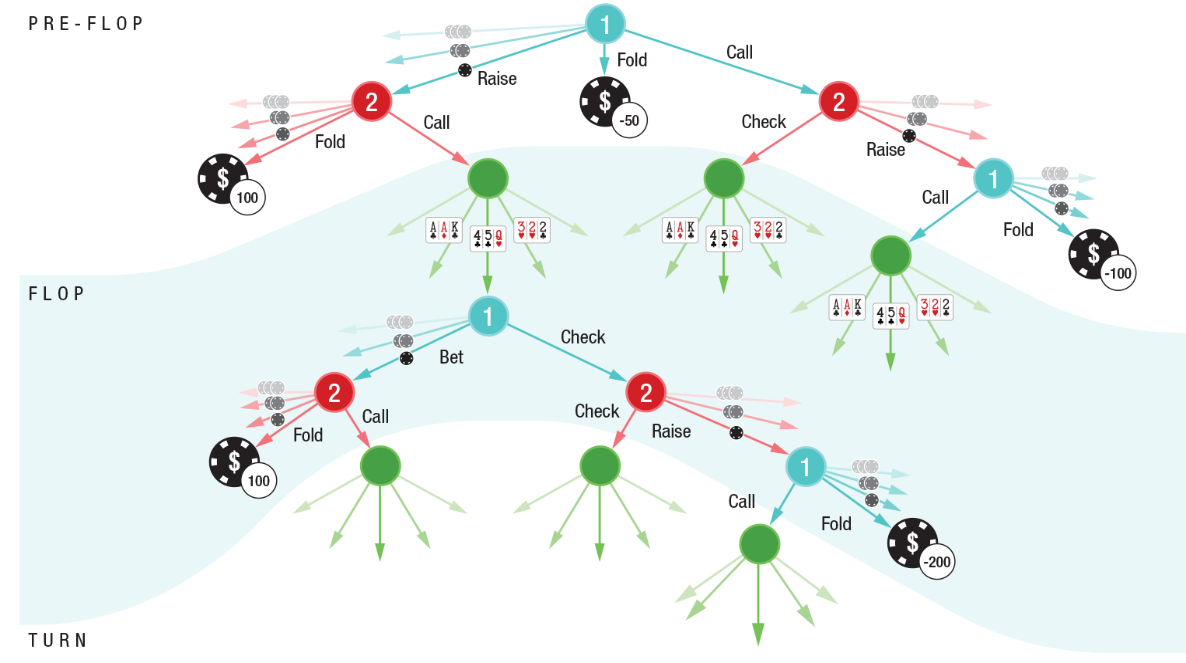


Рисунок 1.2 - Дерево решений программы DeepStack в игре один на один безлимитного холдема на префлопе и флопе.

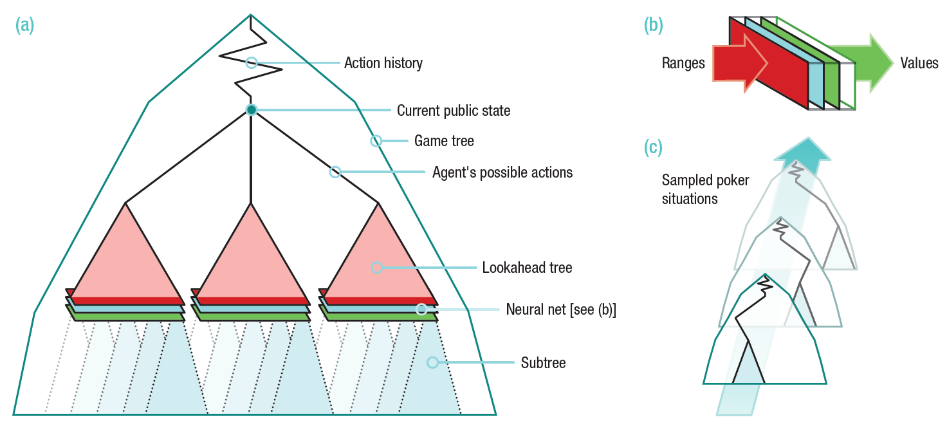


Рисунок 1.3 – Архитектура DeepStack.

Особенностью программы является то, что она активно сопротивляется анализу своей стратегии со стороны оппонента. Другими словами, программа использует равновесие Нэша — ключевое понятие теории игр. Под равновесием Нэша подразумевается набор стратегий, котором ни один участник не может увеличить выигрыш, изменив свою стратегию, если другие участники своих стратегий не меняют. С точки зрения антагонистической игры в покер основной задачей DeepStack является поиск равновесия Нэша, то есть минимизация возможности эксплуатации своей стратегии другим игроком для получения им прибыли[7].

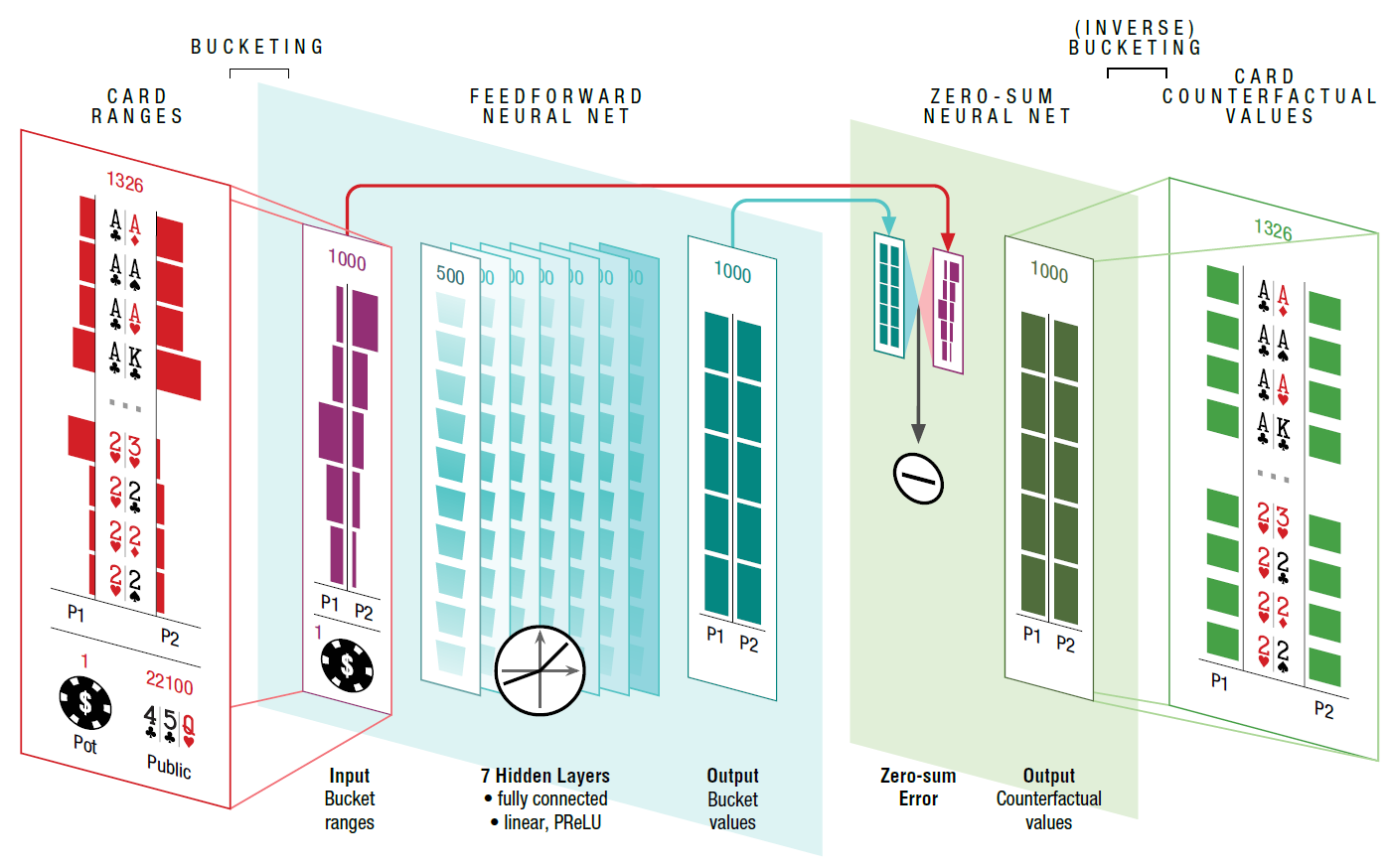


Рисунок 1.4 – Структура нейросити, используемой в алгоритме DeepStack.

## 1.4 Постановка задачи

Необходимо разработать систему искусственного интеллекта для игры в покер (покерный бот). Разрабатываемый бот должен быть способен принимать рациональные решения в рамках игры лимитный Техасский Холдем. Не предполагается использование бота в реально действующих покер румах и управление реальными денежными средствами.

Для обеспечения работы покерного бота предлагается развернуть собственный сервер, обеспечивающий работу покер рума. Данный сервер не должен предоставлять полный функционал покер рума. В его рамках необходимо реализовать логику работы покерного стола для лимитного Техасского Холдема, и предоставить возможность игры для двух или более участников.

Покерный бот должен обмениваться сообщениями с приложением развернутом на сервере через TCP/IP протоколы передачи данных. Так же должен быть обеспечен хотя бы минимальный интерфейс для игры человека с ботами.

# 2 Проектирование приложения

## 2.1 Логика покерного бота

Приведем алгоритм работы покерного бота в виде блок-схемы (рис. 2.1).

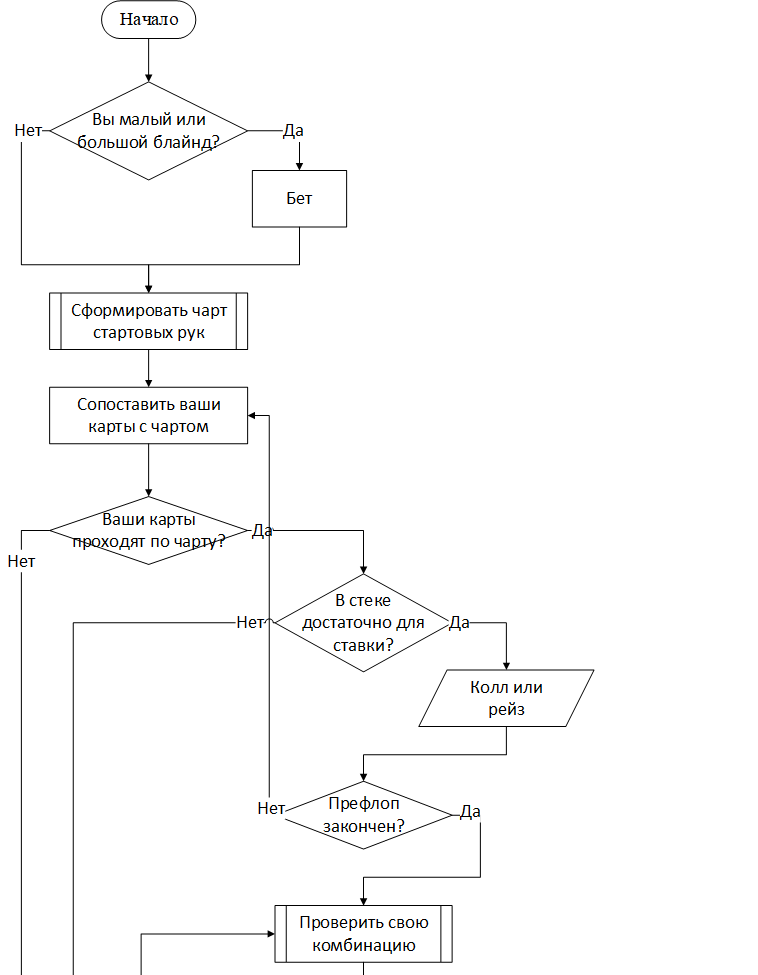


Рисунок 2.1 – Блок-схема алгоритма, часть 1

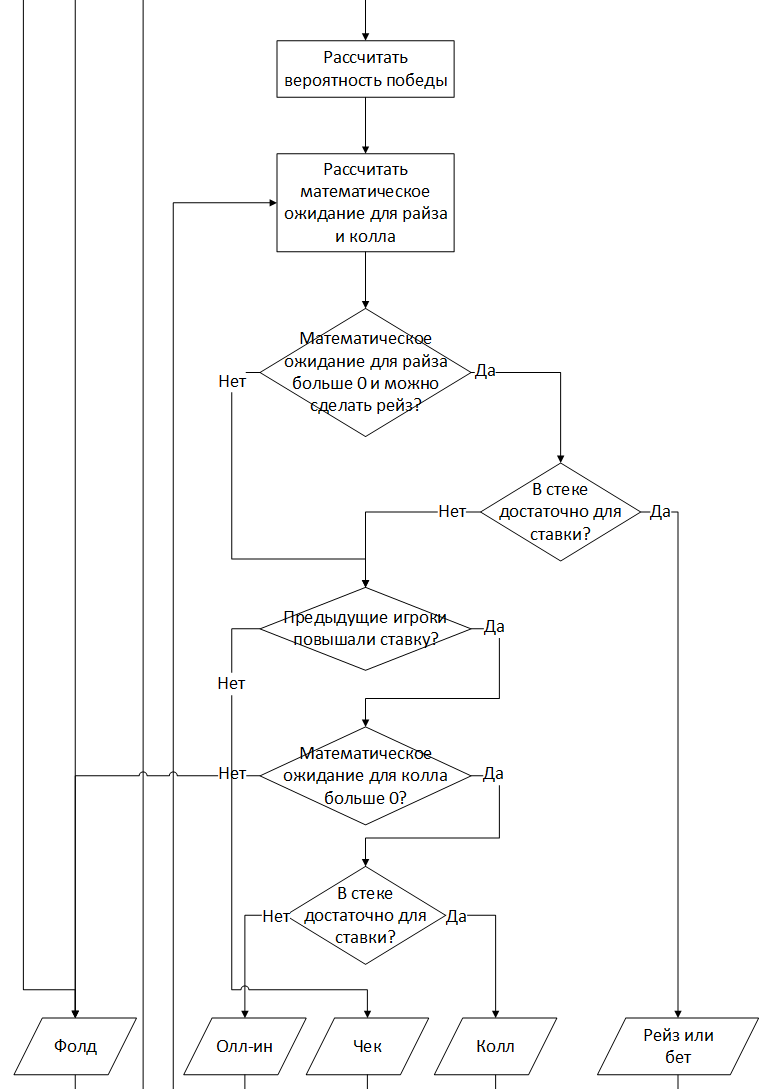


Рисунок 2.2 – Блок-схема алгоритма, часть 2

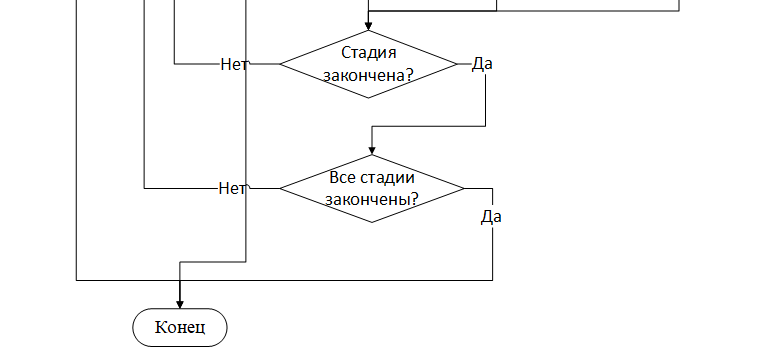


Рисунок 2.3 – Блок-схема алгоритма, часть 3

Чарт стартовых рук приведен ниже в таблице 2.1[4].

Таблица 2.1 – Таблица стартовых рук

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Таблица стартовых рук | | | | | | | | | | |
| Позиция | Ранняя | | Средняя | | Поздняя | | Малый блайнд | | Большой блайнд | |
| Нет рейзов/был рейз | НР | БР | НР | БР | НР | БР | НР | БР | НР | БР |
| AA-QQ | РР | РР | РР | РР | РР | РР | РР | РР | РР | РР |
| JJ | Р | 1Р | Р | 1Р | Р | 1Р | Р | 1Р | Р | 1Р |
| TT-99 | К | Ф | 1Р | 2К | 1Р | К | 1Р | 2К | 1Р | К |
| 88-77 | К | Ф | К | Ф | 0Р | 3К | 0Р | 3К | Ч | К |
| 66-55 | Ф | Ф | 1К | Ф | 0Р | 4К | 0Р | 3К | Ч | К |
| 44-22 | Ф | Ф | 2К | Ф | 2К | 4К | К | 3К | Ч | 2К |
| AKs, AK | РР | РР | РР | РР | РР | РР | РР | РР | РР | РР |
| AQs, AQ, AJs | Р | К | Р | К | Р | К | Р | К | Р | К |
| AJ, ATs, A9s | К | Ф | К | Ф | Р | Ф | Р | Ф | Р | К |
| AT | Ф | Ф | Ф | Ф | 0Р | Ф | 0Р | Ф | Ч | Ф |
| A9 | Ф | Ф | Ф | Ф | Ф | Ф | К | Ф | Ч | Ф |
| A8s-A7s | К | Ф | К | Ф | Р | Ф | К | Ф | Ч | 2К |
| A6s-A2s | Ф | Ф | 1К | Ф | 0Р | Ф | 0Р | Ф | Ч | 3К |
| KQs | К | Ф | 0Р | Ф | 1Р | К | 1Р | К | 1Р | К |
| KQ | 1К | Ф | К | Ф | 1Р | Ф | 1Р | Ф | 1Р | Ф |
| KJs | 1К | Ф | К | Ф | 1Р | 2К | 1Р | 4К | Ч | К |

Продолжение таблицы 2.1

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| KJ | Ф | Ф | Ф | Ф | 1Р | Ф | К | Ф | Ч | Ф |
| KTs | Ф | Ф | К | Ф | 1Р | Ф | 0Р | Ф | Ч | К |
| KT | Ф | Ф | Ф | Ф | 0Р | Ф | 0Р | Ф | Ч | Ф |
| K9s-K5s | Ф | Ф | Ф | Ф | 4К | Ф | К | Ф | Ч | 3К |
| QJs | 1К | Ф | 1К | 3К | 0Р | 3К | 0Р | 4К | Ч | К |
| QTs | Ф | Ф | 1К | Ф | 0Р | Ф | 0Р | Ф | Ч | К |
| QJ-QT | Ф | Ф | Ф | Ф | 2К | Ф | 0Р | Ф | Ч | Ф |
| Q9s-Q8s | Ф | Ф | Ф | Ф | 3К | Ф | К | Ф | Ч | 2К |
| Q7s-Q5s | Ф | Ф | Ф | Ф | Ф | Ф | К | Ф | Ч | 3К |
| JTs | Ф | Ф | 1К | Ф | 0Р | 1К | 0Р | 4К | Ч | К |
| JT | Ф | Ф | Ф | Ф | 2К | Ф | К | Ф | Ч | Ф |
| J9s-J8s | Ф | Ф | Ф | Ф | 3К | Ф | 2К | Ф | Ч | 3К |
| T9s | Ф | Ф | 3К | Ф | 2К | Ф | 2К | Ф | Ч | 2К |
| T9 | Ф | Ф | Ф | Ф | Ф | Ф | 2К | Ф | Ч | 3К |
| T8s | Ф | Ф | Ф | Ф | 3К | Ф | 3К | Ф | Ч | 2К |
| 98s | Ф | Ф | 3К | Ф | 3К | Ф | 2К | Ф | Ч | 2К |
| 98 | Ф | Ф | Ф | Ф | Ф | Ф | 2К | Ф | Ч | 3К |
| 97s | Ф | Ф | Ф | Ф | Ф | Ф | 3К | Ф | Ч | 3К |
| 87s-76s | Ф | Ф | Ф | Ф | 4К | Ф | 3К | Ф | Ч | 2К |

Используется таблица следующим образом:

* Если никто из противников до Вас не сделал рейз, то используйте данные столбца "НР" (нет рейзов). А если кто-то сделал рейз, то смотрите на столбец "БР" (был рейз).
* Если до Вас сделали рейз и ре-рейз, то разыгрывать рекомендуется только руки с символом "re-raise" (ре-рейз).
* На все рейзы, сделанные после нашего хода, нужно отвечать рейзом.
* Call (Колл): Делаем колл.
* Call 0, 1, 2, 3 и т.д (Колл 0, 1, 2, 3 и т.д.): Делаем колл, если до нас руку решило разыгрывать соответствующее количество (или больше) игроков . В противном случае - фолд. При наличии рейза неважно, был ли сначала колл, а потом рейз, или наоборот.
* Raise (Рейз): Делаем рейз.
* Raise 1, 2 и т.д. (Рейз 1, 2 и т.д.): Делаем рейз, если соответствующее количество игроков (или меньше) решило разыграть руку до нас. В противном случае - колл (таким образом, Рейз 0 означает, что делаем рейз, если еще никто не сделал колл).
* Reraise (Ре-рейз): Делаем рейз или ре-рейз как можно чаще.
* Check (Чек). Делаем чек. Чек нужно делать, если мы на большом блайнде, и никто не сделал рейз.

## 2.2 Общая архитектура приложения

Архитектура приведена на рисунке ниже.

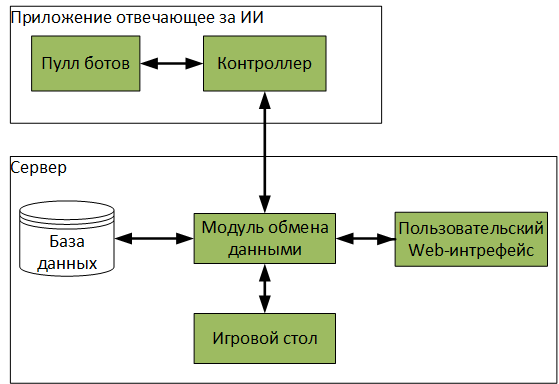


Рисунок 2.4 – Архитектура приложения

Разберем указанные составляющие:

1. Контроллер – часть приложения отвечающего за ИИ. Отвечает за: обмен данными с сервером (состояние стола, потребность в создании новых ботов); создание новых ботов и отправку их в пул; изъятие ботов из пула и обеспечение их информацией о состоянии стола.
2. Пулл ботов – пулл в котором хранятся боты работающие в текущей игре. Каждый бот состоит из модуля принятия решений и информации о состоянии стола. Пулл также самостоятельно удаляет неиспользуемых ботов.
3. База данных – в базе данных хранятся результаты игр для сбора статистики.
4. Модуль обмена данными – модуль отвечающий за обмен данными между различными составляющими программы. Отвечает за: запросы новой игры и состояния текущей игры к игровому столу, принятие сообщения об окончании игры; обмен данными с пользовательским интерфейсом и принятие команд (создание новой игры, решение о ставке в текущей и т.д.) от него; сохранение результатов игры в базу данных; обмен информацией с контроллером ИИ.
5. Игровой стол – модуль обеспечивающий работу покерного стола.
6. Пользовательский web-интерфейс – интерфейс пользователя обеспечивающий возможность игры в покер и просмотра статистики прошлых игр.

# Заключение

В результате практики была изучена предметная область и в общих чертах спроектировано приложение обеспечивающее работу покерного бота и игры в покер.

# Список литературы

1. Рой Раундер. Легкая покерная математика. Секреты безлимитного покера [Электронный ресурс] / Рой Раундер – 2006. – Режим доступа: http://pokerbunker.by/books/kniga-poker-math-made-easy.pdf.
2. Дэн Харрингтон. Харрингтон о Холдеме. Стратегическая игра. Том 1 [Текст] / Дэн Харрингтон, Билл Роберти – Самара: Сафари, 2008. – 293 c.
3. Принципы работы покерного бота. ч.2 [Электронный ресурс] // Хабрахабр – Режим доступа: https://habr.com/company/pokeroff/blog/105025/
4. Лимитированный холдем. Игра на пре-флопе [Электронный ресурс] // PokerStars School. – Режим доступа: <https://www.pokerstarsschool.ru/article/Limitirovannyj-holdem-Igra-na-pre-flope>
5. Matej Moravčík. DeepStack: Expert-Level Artificial Intelligence in No-Limit Poker [Электронный ресурс] / Matej Moravčík [и др.] // arXiv.org – 2017. – Режим доступа: <https://arxiv.org/pdf/1701.01724v2.pdf>
6. Прощай, лимитный холдем? [Электронный ресурс] // GIPSYTEAM – Режим доступа: <http://www.gipsyteam.ru/news/4362-proschay-limitnyy-holdem>
7. Покерная программа DeepStack обыгрывает профессионалов один на один [Электронный ресурс] // Хабрахабр – Режим доступа: <https://habr.com/post/400709/>
8. Покер [Электронный ресурс] // Википедия – Режим доступа: <https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BA%D0%B5%D1%80>
9. Как играть в покер? Правила игры [Электронный ресурс] // Академия покера – Режим доступа: http://academypoker.ru/rules.html