Министерство образования и науки РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

**факультет программной инженерии и компьютерной техники**

**ЛАБОРАТОРНАЯ РАБОТА №1-3**

по дисциплине

‘Системы искусственного интеллекта‘

Тема – Overwatch

*Выполнил:*

Студент группы P33312

Абульфатов Р.М.

*Преподаватель:*

Кугаевских А.В.



Санкт-Петербург, 2023

Введение

Целью первого блока лабораторных работ является разработка базы знаний, онтологии и программы, которая будет использовать базу знаний или онтологию для предоставления рекомендаций на основе введенных пользователем данных.

Значимость данного проекта заключается в том, что он помогает автоматизировать процесс принятия решений, повышает эффективность бизнес-процессов и создает персонализированные рекомендации.

Анализ требований

Система поддержки принятия решений должна:

* Корректно обрабатывать ввод пользователя.
* Поддерживать несколько видов входных строк.
* В случае некорректного ввода известить пользователя.
* Иметь возможность к расширению как инфраструктуры, так и базы знаний.
* При запуске приложения инициализировать базу знаний самостоятельно.

Требования к базе знаний:

* База знаний должна основываться на информации про игру Overwatch
* Должна состоять как из фактов, так и из правил.
* Должна уточнять требования к онтологии, включая определение классов, свойств и отношений между сущностями.

Изучение основных концепций и инструментов

База знаний — это структурированное хранилище информации, которое содержит данные, факты, понятия и знания, организованные таким образом, чтобы обеспечивать доступ к ним и упорядоченное использование. Базы знаний используются для хранения и управления знаниями о предметной области, описывая её в формальной или полуформальной структуре, которая позволяет компьютерным системам или людям легко извлекать и использовать информацию.

База Знаний (Knowledge Base):

Знание: это информация или факты о предметной области, которые хранятся в базе знаний.

Факты: Утверждения или данные, которые описывают сущности и их характеристики в базе знаний.

Правила: Логические конструкции, определяющие отношения между фактами и позволяющие делать выводы.

Запросы: Запросы позволяют получать информацию из базы знаний, основываясь на заданных условиях.

Онтология — это формальная модель, описывающая предметную область и отношения между понятиями в этой области. Она определяет понятия, их свойства и отношения, что позволяет компьютерным системам понимать и интерпретировать знания, представленные на основе этой онтологии.

Онтология (Ontology):

Классы: определяют категории или типы сущностей в предметной области.

Свойства: описывают атрибуты или характеристики сущностей и их взаимодействие.

Инстансы: Конкретные объекты или элементы, которые принадлежат к определенным классам.

Отношения: определяют связи и взаимодействие между сущностями и классами.

**Сходства**:

Организация информации: как онтология, так и база знаний используются для организации и структурирования информации. Онтология определяет семантическую структуру данных, в то время как база знаний хранит и обрабатывает факты и знания.

Использование в информационных системах: Оба понятия активно применяются в информационных системах и искусственном интеллекте для эффективного управления и анализа данными.

**Различия**:

Абстракция и структура: Онтология более абстрактна и описывает концепции и их отношения в предметной области, в то время как база знаний фокусируется на конкретных данных и фактах.

Семантика: Онтология уделяет особое внимание семантике и смыслу данных, в то время как база знаний может содержать данные без явного определения семантики.

Цель: Онтология используется для обеспечения семантической интерпретации данных, в то время как база знаний служит для хранения и доступа к информации.

Основное отличие между базами знаний и онтологиями заключается в способе представления и организации знаний. Базы знаний представляют собой собрание фактов и правил, в то время как онтологии моделируют понятия и отношения между ними.

В итоге, онтология и база знаний взаимосвязаны, но выполняют разные функции в организации и управлении информацией в различных предметных областях.

Prolog (Programming in Logic) - это декларативный язык программирования, который используется для решения задач искусственного интеллекта и логического программирования. Prolog основан на логике первого порядка и предназначен для решения задач, в которых логические отношения и правила играют ключевую роль.

Инструмент для Базы знаний - <https://swish.swi-prolog.org/>

Инструмент для Онтологий - <https://protege.stanford.edu/software.php#desktop-protege>

Реализация системы искусственного интеллекта на Prolog

7 основных класса:

* Team\_in\_game – команды в игре
* Player\_in\_game – игрок
* Class – класс героя в игре
* Hero – герой в игре
* Map – карты в игре
* Status\_ulta – статус ульты
* Type – режимы карт

Правила:

%Правило о принадлежности к команде "красные"

is\_red(Name\_Player):-

player\_in\_game(Name\_Player),

player\_team(Name\_Player, red).

%Правило о принадлежности к команде "синие"

is\_blue(Name\_Player):-

player\_in\_game(Name\_Player),

player\_team(Name\_Player, blue).

%Правило, проверяющее, что персонаж, которого взял игрок относится к классу "танк"

is\_character\_player\_tank(Name\_Player):-

player\_in\_game(Name\_Player),

player\_hero(Name\_Player, Hero),

hero\_class(Hero, tank).

%Правило, проверяющее, что персонаж относится к классу "саппорт"

is\_character\_heal(Hero):-

hero\_class(Hero, heal).

%Правило, проверяющее, что карта относится к режиму "control"

is\_map\_type\_control(Map):-

map(Map),

type\_map(Map, control).

%Правило, проверяющее, возможность использования ульты

ability\_player\_ulta(Name\_Player, Status\_Ulta):-

player\_in\_game(Name\_Player),

player\_status\_ulta(Name\_Player, Status\_Ulta).

%Правило, проверяющее, есть ли в команде персонаж genji

has\_team\_genji(Team):-

team\_in\_game(Team),

player\_in\_game(Name\_Player),

player\_team(Name\_Player, Team),

player\_hero(Name\_Player, genji).

%Правило, проверяющее, кем был убит игрок "kriper2004"

is\_player\_kill\_kriper2004(Name\_Player):-

player\_in\_game(Name\_Player),

kill(Name\_Player, kriper2004).

%Правило, проверяющее, играет ли игрок за genji

is\_player\_genji(Name\_Player):-

player\_in\_game(Name\_Player),

player\_hero(Name\_Player, genji).

Оценка и интерпретация результатов

Запросы к нашей системе поддержки принятия решений

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Наша система соответствует всем поставленным требованиям.

Цель была достигнута – была разработана база знаний и онтология по игре Overwatch, на их основе было сделано приложение, которое помогает пользователю создавать запросы и узнавать новую информацию.

В дальнейшем можно дополнять базу знаний, прописывать новые связи. Также добавить больше шаблонов для пользовательского ввода и сделать ввод более гибким.

Заключение

Разработанное приложение может помочь новым игрокам узнать больше про игровой процесс, подобрать себе подходящего героя, узнать какие карты и режимы есть в игре. Оно может применяться на различных игровых форумах или сайтах про игру Overwatch.