

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Национальный исследовательский университет ИТМО»

Факультет Программной инженерии и компьютерной техники

Модуль №1

Базы знаний и онтологии

Вселенная The Witcher

Группа: Р33312

Выполнил: Обляшевский С.А.

Проверил:

Кугаевских А. В.

Санкт-Петербург, 2023

Введение:

Цели: основной целью данного проекта было создание Базы знаний с помощью языка Prolog, а также онтологии в Protégé. Помимо этого, одна из целей – разработка программы, способная принимать решения на основе базы знаний или онтологии.

Значимость: основной значимостью данного проекта я считаю создание системы поддержки принятия решений, которая помогает автоматизировать процессы поиска информации в базе знаний и принятия решений.

Анализ требований:

Требования к базе знаний/онтологии:

- База знаний должна основываться на вселенной The Witcher.
- База знаний должна состоять из минимум 20 фактов с одним аргументом, 10–15 фактов с двумя аргументами и 5-7 правил.
- Должна уточнять требования к онтологии, включая определение классов, свойств и отношений между сущностями.

Требования к системе поддержки принятия решений:

- Обработать любой ввод пользователя, как корректный, так и нет.
- Должна иметь способность к расширению функционала.
- При запуске приложений инициализировать базу знаний самостоятельно.
- Иметь несколько паттернов входных строк.

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 полужформальной 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

```
%-----Факты-----  
  
%   Расы персонажей  
%   Нелюди (краснолюды, эльфы, мутанты)  
dwarf('Zoltan').  
mutant('Geralt Of Rivia').  
mutant('Lambert').  
mutant('Eskel').  
mutant('Vesemir').  
  
%   Люди  
human('Philippa Eilhart').  
human('Triss Merigold').  
human('Margarita Laux-Antille').  
human('Keira Metz').  
human('Fringilla Vigo').  
human('Yennefer').  
human('Emhyr var Emreis').  
human('Foltest').  
human('Vizimir').  
human('Taler').  
human('Dijkstra').  
human('Vernon Roche').  
human('Cahir').  
  
%   Роли  
%   Ведьмаки  
witcher('Geralt Of Rivia').  
witcher('Lambert').  
witcher('Eskel').  
witcher('Vesemir').  
  
%   Чародейки  
enchantress('Philippa Eilhart').  
enchantress('Triss Merigold').
```

```

enchantress('Margarita Laux-Antille').
enchantress('Keira Metz').
enchantress('Fringilla Vigo').
enchantress('Yennefer').

%   Правители (императоры и короли)
imperator('Emhyr var Emreis').
king('Foltest').
king('Vizimir').

%   Военные (шпионы и предводители)
spy('Taler').
spy('Dijkstra').
commander('Vernon Roche').
commander('Cahir').

%-----Свойства-----
%   Страны, где находятся персонажи
%   Ведьмаки не указаны из-за постоянных путешествий
country('Emhyr var Emreis', 'Nilfgaard').
country('Foltest', 'Temeria').
country('Vizimir', 'Redania').
country('Philippa Eilhart', 'Redania').
country('Triss Merigold', 'Temeria').
country('Margarita Laux-Antille', 'Temeria').
country('Keira Metz', 'Temeria').
country('Fringilla Vigo', 'Nilfgaard').
country('Yennefer', 'Caedven').
country('Vernon Roche', 'Temeria').
country('Cahir', 'Nilfgaard').
country('Taler', 'Temeria').
country('Dijkstra', 'Redania').

%   Специфичные места, где находятся некоторые персонажи
%   Несмотря на то что страна, где находится Аретуза - Темерия, ее обитатели -
%   выходцы из разных мест. Они также не являются подданными определенного короля
place('Margarita Laux-Antille', 'Arethusa').
place('Keira Metz', 'Arethusa').

%   Расположение стран на Континенте
location('Redania', 'North').
location('Temeria', 'North').
location('Caedven', 'North').
location('Nilfgaard', 'South').

%-----Правила-----
%   2 персонажа являются соотечественниками, если они из одной страны.
compatriots(Person1, Person2) :- country(Person1, X), country(Person2, X),
Person1 \= Person2.
%   2 персонажа являются(лись) любовниками, если один из них - Геральт, а второй
%   - чародейка.

```

```

lovers(Person1, Person2) :- Person1 = 'Geralt Of Rivia', enchantress(Person2);
Person2 = 'Geralt Of Rivia', enchantress(Person1).
% 2 персонажа находятся в одной части Континента, если их страны расположены на
одной части.
from_one_side_of_the_Continent(Person1, Person2) :- country(Person1, Country1),
country(Person2, Country2), location(Country1, X), location(Country2, X).
% 2 персонажа являются союзниками, если они находятся на одной части
Континента, либо же ведьмаки (по профессиональной солидарности).
allies(Person1, Person2) :- from_one_side_of_the_Continent(Person1, Person2),
Person1 \= Person2; witcher(Person1), witcher(Person2), Person1 \= Person2.
% Персонаж 1 - поданный персонажа 2, если они живут в одной стране и Персонаж 2
- король/император. За исключением некоторых случаев.
subject(Person1, Person2) :- compatriots(Person1, Person2), (king(Person2);
imperator(Person2)), not(place(Person1, 'Arethusa')).
% Персонаж является военным, если он шпион или командир.
military(Person) :- spy(Person); commander(Person).
% Персонаж умеет сражаться, если он военный, ведьмак или красноряд.
can_fight(Person) :- military(Person); witcher(Person); dwarf(Person).
% Персонаж использует магию, если он чародейка или ведьмак.
magician(Person) :- enchantress(Person); witcher(Person).
% Персонаж может находиться в Новиграде, если он не чародейка и не нелюдь.
Исключение - ведьмаки.
can_be_in_Novigrad(Person) :- not(enchantress(Person)), (human(Person);
witcher(Person)).

%-----Запросы-----
% Простые для поиска фактов
% Геральт из Ривии - ведьмак?
% witcher('Geralt Of Rivia') - true
% Каэдвен находится на Севере?
% location('Caedven', 'North') - true
% Фольтест - император?
% imperor('Foltest') - false
% Золтан - человек?
% human('Zoltan') - false
% Трисс Меригольд - из Темерии?
% country('Triss Merigold', 'Temeria') - true

% Составные для поиска фактов
% Визимир - король и живет в Темерии или Редании?
% king('Vizimir'), (country('Vizimir', 'Temeria'); country('Vizimir', 'Redania'))
- true
% Золтан - красноряд-чародейка и живет в Каэдвене?
% dwarf('Zoltan'), country('Zoltan', 'Caedven'), enchantress('Zoltan') - false
% Дийкстра - шпион из Темерии?
% spy('Dijkstra'), country('Dijkstra', 'Temeria') - false
% Филиппа Эйльхарт - чаролейка и находится в Аретузе?
% enchantress('Philippa Eilhart'), place('Philippa Eilhart', 'Arethusa') - false
% Кейра Мец и Маргарита Ло-Антиль обе либо в Аретузе либо не там?

```

```

% (place('Keira Metz', 'Arethusa'), place('Margarita Laux-Antille', 'Arethusa');
not(place('Keira Metz', 'Arethusa')), not(place('Margarita Laux-Antille',
'Arethusa')))) - true

% Поиск объектов с определенными характеристиками через переменные
% Люди, живущие в Редании
% human(X), country(X, 'Redania') - X = {'Philippa Eilhart', 'Vizimir',
'Dijkstra'}
% Высокопоставленные люди (император/король, предводители) из Темерии
% human(X), country(X, 'Temeria'), (commander(X); imperor(X)) - X = 'Vernon
Roche'
% Чародейки с Юга, с эльфийской кровью
% enchantress(X), country(X, Y), location(Y, 'South'), elf(X) - false
% Северные короли/императоры
% (king(X); imperor(X)), country(X, Y), location(Y, 'North') - X = {'Foltest',
'Vizimir'}, Y = {'Temeria', 'Redania'}
% Чародейки, работающие на Темерию с Вероном Роше
% enchantress(X), country(X, Y), country('Vernon Roche', Y), not(place(X),
'Arethusa') - X = 'Triss Merigold', Y = 'Temeria'

% Требуют выполнения правил для получения рез-та
% Люди, живущие в одной стране с Фольтестом, но не являющиеся его подчиненными
% human(X), compatriots(X, 'Foltest'), not(subject(X, 'Foltest')) - X =
{'Margarita Laux-Antille', 'Keira Metz'}
% Военные, находящиеся на одной части Континента с императором
% military(X), from_one_side_of_the_Continent(X, Y), imperor(Y) - X = 'Cahir', Y
= 'Emhyr var Emreis'
% Люди, обладающие магическими силами, которые умеют сражаться и являются
союзниками Весемира
% magician(X), can_fight(X), allies(X, 'Vesemir'), X \= 'Vesemir' - X = {'Geralt
Of Rivia', 'Lambert', 'Eskel'}
% Полководец, против которого может сражаться Верон Роше
% military(X), not(allies(X, 'Vernon Roche')) - X = {'Cahir'}
% Люди, обладающие магической силой, способные находится в Новиграде и имеющие
любовников
% magician(X), can_be_in_Novigrad(X), lovers(X, _), ! - X = 'Geralt Of Rivia'

```

Оценка и интерпретация результатов

Запросы к системе поддержки принятия решений:

```
> В какой стране живет Dijkstra?
Dijkstra живет в Redania.
> В какой части Континента находится Caedven?
Caedven находится на North.
> Перечисли подданных Foltest.
Найдены следующие подданные Foltest:
Triss Merigold
Vernon Roche
Taler
> Кто является союзником Emhyr var Emreis?
Найдены следующие союзники Emhyr var Emreis:
Fringilla Vigo
Cahir
> Geralt Of Rivia может находиться в Новиграде?
Geralt Of Rivia разрешено быть в Новиграде.
> exit

Process finished with exit code 0
```

Система соответствует всем поставленным требованиям.

Цель достигнута. Была создана система поддержки принятия решений по вселенной The Witcher, с помощью которой пользователь может найти интересующую его информацию

В последствии система может расширяться, как с точки зрения базы данных, так и с точки зрения запросов, которые может выполнять пользователь.

Заключение

Данное приложение поможет как новым фанатам, которые недавно заинтересовались данной вселенной и хотят более углубленно разобраться в ней, так и тем, кто давно увлекся Ведьмаком и хочет освежить какие-то моменты в памяти.