

Принцип реализации системы

Мы решили воспользоваться прямым выводом. Для этого достаточно было воспользоваться CLIPS с помощью библиотеки на python Pyknow. Prolog мы помнили хуже, чем python, поэтому выбор лег именно на этот язык программирования. В начале система задает пользователю вопросы, а затем обрабатывает ответы и выдает результат.

Механизм вывода

Мы узнаем интересы нашего пользователя и заносим их в список фактов о нем. В Pyknow заданы правила с помощью которых факты из списка проверяются и доказываются, а впоследствии добавляются в память. Это повторяется до тех пор пока мы не найдем интересующие нас утверждения.

Правила в системе Pyknow

```
class Film(KnowledgeEngine):
    result=[]

    @Rule(OR(Fact(age='подросток(13-18)'),
            Fact('необычные способности'))
    def unreal_heroes(self):
        self.declare(Fact('нравится смотреть на вымышленных\
несуществующих героев'))

    @Rule(Fact(age='подросток(13-18)'))
    def unreal_history(self):
        self.declare(Fact('нравятся вымышленные истории'))

    @Rule(Fact(age='старше 40'))
    def world_news(self):
        self.declare(Fact('информация про мировые события'))

    @Rule(OR(Fact(grad='высшее образование'), Fact(grad='неоконченное
высшее образование'), Fact(age='подросток(13-
18)'), Fact(age='половозрелый(19-40)'))
    def found_of_history(self):
        self.declare(Fact('увлекаемся историей'))

    @Rule(OR(Fact('нравится сакура'), Fact('нравятся
самураи'), Fact('нравится кимоно'))
    def japan_cul(self):
        self.declare(Fact('нравится японская культура'))
```

```

@Rule(AND(Fact('нравится смотреть на вымышленных\несуществующих
героев'), Fact('нравятся вымышленные истории'))))
def unreal_screen(self):
    self.declare(Fact('несуществующий сюжет'))

@Rule(OR(AND(Fact(sex='женщина'), Fact(age='половозрелый(19-
40)')), Fact(temp='флегматик'), Fact(temp='сангвинник'), Fact(temp='меланхо
лик'))))
def emotional(self):
    self.declare(Fact('хочется эмоционального'))

@Rule(OR(AND(Fact(sex='мужчина'), Fact(age='половозрелый(19-
40)')), Fact(temp='холерик'), Fact('драки'), Fact('убийства'))))
def cruel(self):
    self.declare(Fact('нравится жестокость'))

@Rule(OR(Fact(age='половозрелый(19-40)'), Fact('нравится борьба двух
сторон'))))
def tension(self):
    self.declare(Fact('нравится напряжение'))

@Rule(OR(Fact('нравятся самураи'), Fact('нравится
оружие'), AND(Fact(sex='мужчина'), Fact(age='половозрелый(19-40)'),
Fact('нравится военная техника'), Fact('нравится военная
культура'))))
def military(self):
    self.declare(Fact('нравится военная тема'))

@Rule(AND(Fact('нравится напряжение'), Fact('стилистические
фильмы'))))
def dark_cadr(self):
    self.declare(Fact('нравятся "тёмные" кадры'))

@Rule(OR(Fact('мрачная атмосфера'), Fact('малое количество
диалогов'))))
def style_film(self):
    self.declare(Fact('стилистические фильмы'))

@Rule(OR(Fact('нравятся животные'), Fact('увлекаемся
историей'), Fact('информация про мировые события'))))
def education(self):
    self.declare(Fact(film='документальные'))

```

```

@Rule(AND(Fact('несуществующий сюжет'), Fact('нравится японская
культура'))))
def anime(self):
    self.declare(Fact(film='аниме'))

@Rule(AND(Fact('несуществующий сюжет'), Fact('увлекаемся историей'))))
def fantastic(self):
    self.declare(Fact(film='фантастика'))

@Rule(AND(Fact('ребёнок (<12)'), Fact('элемент
сказок'), Fact('нравятся абсурдные ситуации'), Fact('семья'))))
def cartoon(self):
    self.declare(Fact(film='мультфильмы'))

@Rule(AND(Fact('грустный фильм'), Fact('хочется эмоционального'))))
def drama(self):
    self.declare(Fact(film='драмма'))

@Rule(AND(Fact('семья'), Fact('хочется
эмоционального'), OR(Fact('хочется посмеяться'), Fact('нравятся абсурдные
ситуации'))))
def comedy(self):
    self.declare(Fact(film='комедия'))

@Rule(AND(Fact('нравится напряжение'), Fact('хочется
эмоционального'), Fact('нравится жестокость'), Fact('нравится военная
тема'))))
def boevik(self):
    self.declare(Fact(film='военные\боевики'))

@Rule(AND(Fact('нравятся страшные истории'), Fact('хочется
эмоционального'))))
def screamer(self):
    self.declare(Fact(film='ужасы'))

@Rule(AND(Fact('Пораскинуть мозгами'), Fact('нравятся "тёмные"
кадры'))))
def nuar(self):
    self.declare(Fact(film='нуар'))

@Rule(AND(Fact('нравится напряжение'), Fact('стилистические

```

```

фильмы' ))
    def western(self):
        self.declare(Fact(film='вестерн'))

    @Rule(AND(Fact('нравится заниматься спортом'), Fact('информация про
мировые события')) )
    def sport(self):
        self.declare(Fact(film='спорт'))

    @Rule(Fact(film=MATCH.a))
    def print_result(self, a):
        self.result.append(a)
        print(f'Жанр- {a}')

    def factz(self, l):
        for x in l:
            self.declare(x)

```

Механизм ввода:

```

for key in my_questions.keys():
    print(key)
    print_list(my_questions[key])
    user_answer=input()
    user_answer_list=[]
    for l in my_questions[key]:
        if l.find(user_answer+'.')>-1:
            user_answer_list=l.split('.')
    if user_answer_list[1]=='да':
        user_ans.append(key[:-1])
    elif user_answer_list[1]!='нет' :
        user_ans.append(user_answer_list[1])
ex1 = Film()
ex1.reset()

facts_my=[Fact(age=user_ans[0]),
          Fact(sex=user_ans[1]),
          Fact(grad = user_ans[2]),
          Fact(temp = user_ans[3])]

for i in range(4, len(user_ans)):
    facts_my.append(Fact(user_ans[i]))

```

Извлечение знаний и база знаний

Базу знаний жанров фильмов мы составили так: каждый из нас взял те жанры, которые ему наиболее интересны и представил каким людям они нравятся, по этому принципу составил правила.

Протокол работы системы

Сколько лет?

0. подросток (13-18)

1. половозрелый (19-40)

2. старше 40

0

Вы?

0. мужчина

1. женщина

0

У Вас?

0. неоконченное высшее образование

1. высшее образование

0

Ваш темперамент?

0. холерик

1. флегматик

2. сангвиник

3. меланхолик

1

необычные способности?

0. да

1. нет

1

нравится сакура?

0. да

1. нет

0

нравятся самураи?

0. да

1. нет

0

нравится кимоно?

0. да

1. нет

0

драки?

0. да

1. нет

0

убийства?

0. да

1. нет

0

нравится борьба двух сторон?

0. да

1. нет

0

нравится оружие?

0. да

1. нет

1

нравится военная техника?

0. да

1. нет

1

-
нравится военная культура?
0.да
1.нет
1
мрачная атмосфера?
0.да
1.нет
1
малое количество диалогов?
0.да
1.нет
1
нравятся животные?
0.да
1.нет
1
ребёнок (<12)?
0.да
1.нет
1
элемент сказок?
0.да
1.нет
1
нравятся абсурдные ситуации?
0.да
1.нет
1
семья?
0.да
1.нет
1
грустный фильм?
0.да
1.нет
1
нравятся страшные истории?
0.да
1.нет
0
Пораскинуть мозгами?
0.да
1.нет
0
нравится заниматься спортом?
0.да
1.нет
1
Жанр- военные\боевики
Жанр- ужасы
Жанр- документальные
Жанр- фантастика
Жанр- аниме

Пример 2

Сколько лет?

0.подросток(13-18)

1.половозрелый(19-40)

2.старше 40

1

Вы?

0.мужчина

1.женщина

1

У Вас?

0.неоконченное высшее образование

1.высшее образование

1

Ваш темперамент?

0.холерик

1.флегматик

2.сагвинник

3.меланхолик

2

необычные способности?

0.да

1.нет

1

нравится сакура?

0.да

1.нет

1

нравятся самураи?

0.да

1.нет

1

нравится кимоно?

0.да

1.нет

1

драки?

0.да

1.нет

1

убийства?

0.да

1.нет

1

нравится борьба двух сторон?

0.да

1.нет

1

нравится оружие?

0.да

1.нет

1

нравится военная техника?

0.да

1.нет

1

нравится военная культура?

0.да

1.нет

1

мрачная атмосфера?

0.да

1.нет

1

малое количество диалогов?

0.да

1.нет

1

нравятся животные?

0.да

1.нет

0

ребёнок (<12)?

0.да

1.нет

0

элемент сказок?

0.да

1.нет

0

нравятся абсурдные ситуации?

0.да

1.нет

1

семья?

0.да

1.нет

0

грустный фильм?

0.да

1.нет

0

нравятся страшные истории?

0.да

1.нет

1

Пораскинуть мозгами?

0.да

1.нет

1

нравится заниматься спортом?

0.да

1.нет

1

хочется посмеяться?

0.да

1.нет

0

Жанр- мультфильмы

Жанр- комедия

Жанр- драма

Выводы

Данная работа научила нас работать в команде. Было немного трудно, так как команды составлялись рандомно, а не по взаимной симпатии. Основное общение у нас было в discord. Нам очень понравилась наша концепция экспертной системы, нам кажется она могла быть полезной для людей, которым трудно сделать выбор, а также компаниям людей во время совместного выбора фильма. Мы думаем в будущем нам понадобятся навыки создания экспертных систем, ведь не всегда нужно заниматься сложным машинным обучением, когда можно воспользоваться достаточно несложной логикой экспертных систем.