

Системы искусственного интеллекта

Лабораторная работа №3

Создание информационной системы на базе семантической сети

Выполнил: Анищенко А.А.

Группа Р33113

Преподаватель: Болдырева Е.А.

Цель работы

Изучение семантической сети как инструмента создания информационных и обучающих систем?

А также исследование методов логического вывода на основе правил.

Предметная область

Генеалогическо дерево

Code Semantic

facts.pl

```
1 f("anischenko_alexey", "is_parent", "anischenko_anatoliy").
2 f("anischenko_julia", "is_parent", "anischenko_anatoliy").
3 f("anischenko_alexey", "is_parent", "anischenko_timofey").
4 f("anischenko_julia", "is_parent", "anischenko_timofey").
5 f("anischenko_vitaliy", "is_parent", "anischenko_alexey").
6 f("anischenko_olga", "is_parent", "anischenko_alexey").
7 f("anischenko_vitaliy", "is_parent", "anischenko_igor").
8 f("anischenko_olga", "is_parent", "anischenko_igor").
9 f("kukushkin_michail", "is_parent", "anischenko_julia").
10 f("kukushkina_tatiana", "is_parent", "anischenko_julia").
11 f("kukushkin_pavel", "is_parent", "kukushkin_michail").
12 f("kukushkina_vera", "is_parent", "kukushkin_michail").
13 f("kukushkin_pavel", "is_parent", "tsygonyaeva_elena").
14 f("kukushkina_vera", "is_parent", "tsygonyaeva_elena").
15 f("drobov_aleksandr", "is_parent", "kukushkina_tatiana").
16 f("drobova_nina", "is_parent", "kukushkina_tatiana").
17 f("drobov_aleksandr", "is_parent", "ilyicheva_luda").
18 f("drobova_nina", "is_parent", "ilyicheva_luda").
19 f("tsygonyaev_yuriy", "is_parent", "tsygonyaev_alexey").
20 f("tsygonyaeva_elena", "is_parent", "tsygonyaev_alexey").
21 f("tsygonyaev_yuriy", "is_parent", "tsygonyaev_vitaliy").
22 f("tsygonyaeva_elena", "is_parent", "tsygonyaev_vitaliy").
23 f("tsygonyaev_alexey", "is_parent", "tsygonyaev_ivan").
24 f("tsygonyaeva_tatiana", "is_parent", "tsygonyaev_ivan").
25 f("tsygonyaev_vitaliy", "is_parent", "tsygonyaev_michail").
26 f("tsygonyaeva_anna", "is_parent", "tsygonyaev_michail").
27 f("tsygonyaev_vitaliy", "is_parent", "tsygonyaev_sergey").
28 f("tsygonyaeva_anna", "is_parent", "tsygonyaev_sergey").
29
30 f("anischenko_alexey", "is_a", "Man").
31 f("anischenko_anatoliy", "is_a", "Man").
32 f("anischenko_timofey", "is_a", "Man").
33 f("anischenko_igor", "is_a", "Man").
34 f("anischenko_vitaliy", "is_a", "Man").
35 f("kukushkin_michail", "is_a", "Man").
36 f("kukushkin_pavel", "is_a", "Man").
37 f("drobov_aleksandr", "is_a", "Man").
38 f("tsygonyaev_yuriy", "is_a", "Man").
39 f("tsygonyaev_alexey", "is_a", "Man").
40 f("tsygonyaev_vitaliy", "is_a", "Man").
41 f("tsygonyaev_ivan", "is_a", "Man").
42 f("tsygonyaev_michai", "is_a", "Man").
43 f("tsygonyaev_sergey", "is_a", "Man").
44
45 f("anischenko_julia", "is_a", "Woman").
46 f("anischenko_olga", "is_a", "Woman").
47 f("kukushkina_tatiana", "is_a", "Woman").
48 f("kukushkina_vera", "is_a", "Woman").
49 f("drobova_nina", "is_a", "Woman").
50 f("ilyicheva_luda", "is_a", "Woman").
51 f("tsygonyaeva_elena", "is_a", "Woman").
```

```

52 f("tsygonyaeva_tatiana", "is_a", "Woman").
53 f("tsygonyaeva_anna", "is_a", "Woman").
54
55 f("anischenko_alexey", "is_spouse", "anischenko_julia").
56 f("anischenko_olga", "is_spouse", "anischenko_vitaliy").
57 f("kukushkin_michail", "is_spouse", "kukushkina_tatiana").
58 f("drobov_aleksandr", "is_spouse", "drobova_nina").
59 f("kukushkin_pavel", "is_spouse", "kukushkina_vera").
60 f("tsygonyaeva_elena", "is_spouse", "tsygonyaev_yuriy").
61 f("tsygonyaev_alexey", "is_spouse", "tsygonyaeva_tatiana").
62 f("tsygonyaeva_anna", "is_spouse", "tsygonyaev_vitaliy").
63
64 t("x").
65
66 onto("ontology.pl").

```

ontology.pl

```

1 c("Person").
2 c("Woman").
3 c("Man").
4
5 o("Woman", "is_a", "Person").
6 o("Man", "is_a", "Person").
7
8 r([t("?x", "is_a", "&y"), t("?y", "is_a", "?z")], [t("?x", "is_a", "?z")]).
9
10
11
12 r([t("?x", "is_spouse", "?y"), [t("?y", "is_spouse", "?x")]]).
13 r([t("?x", "is_spouse", "?y"), t("?x", "is_a", "Man")],
14     [t("?x", "is_husband", "?y")]).
15 r([t("?x", "is_spouse", "?y"), t("?x", "is_a", "Woman")],
16     [t("?x", "is_wife", "?y")]).
17
18 r([t("?x", "is_parent", "?y"), [t("?y", "is_child", "?x")]]).
19 r([t("?x", "is_a", "Woman"), t("?x", "is_child", "?y")],
20     [t("?x", "is_daughter", "?y")]).
21 r([t("?x", "is_a", "Man"), t("?x", "is_child", "?y")],
22     [t("?x", "is_son", "?y")]).
23
24 r([t("?x", "is_parent", "?y"), t("?x", "is_a", "Woman")],
25     [t("?x", "is_mom", "?y")]).
26 r([t("?x", "is_parent", "?y"), t("?x", "is_a", "Man")],
27     [t("?x", "is_dad", "?y")]).
28
29 r([t("?x", "is_parent", "?y"), t("?y", "is_parent", "?z")],
30     [t("?x", "is_grandparent", "?z")]).
31 r([t("?x", "is_a", "Man"), t("?x", "is_grandparent", "?y")],
32     [t("?x", "is_granddad", "?y")]).
33 r([t("?x", "is_a", "Woman"), t("?x", "is_grandparent", "?y")],
34     [t("?x", "is_grandmom", "?y")]).
35
36 r([t("?x", "is_grandparent", "?y"), [t("?y", "is_grandchild", "?x")]]).
37 r([t("?x", "is_a", "Man"), t("?x", "is_grandchild", "?y")],
38     [t("?x", "is_grandson", "?y")]).
39 r([t("?x", "is_a", "Woman"), t("?x", "is_grandchild", "?y")],
40     [t("?x", "is_granddaughter", "?y")]).
41
42 r([t("?x", "has_child", "?y"), t("?x", "has_child", "?z"), t("?y", "differs", "?z")],
43     [t("?y", "is_sibling", "?z")]).
44 r([t("?x", "is_sibling", "?y"), t("?x", "is_a", "Man")],

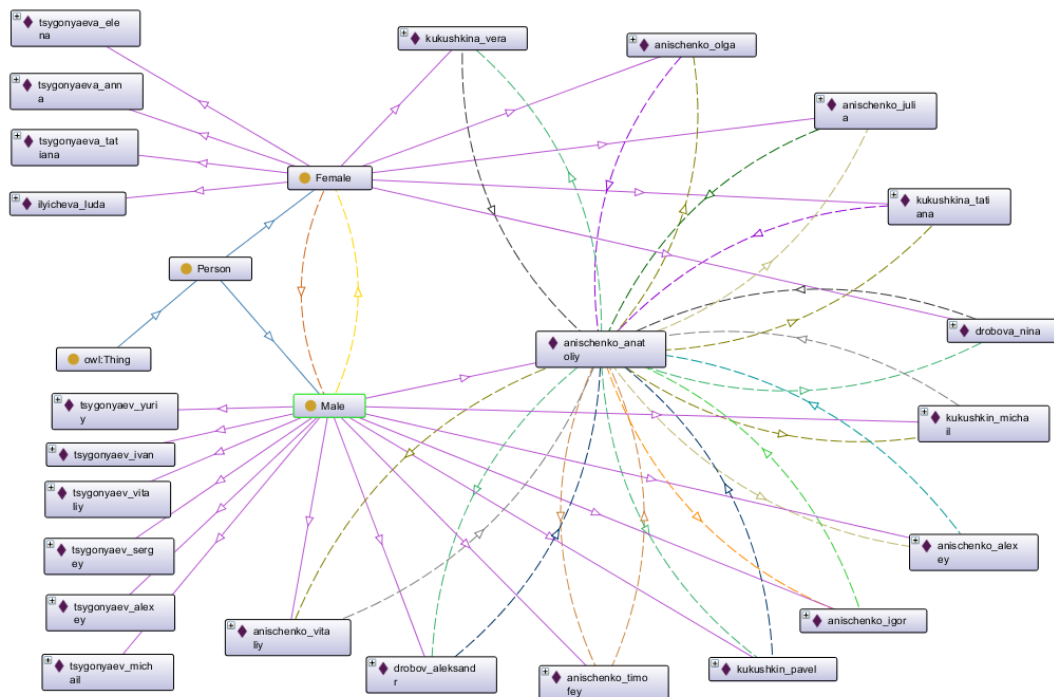
```

```

45         [t("?x", "is_brother", "?y")]).
46 r([t("?x", "is_sibling", "?y"), t("?x", "is_a", "Woman")],
47     [t("?x", "is_sister", "?y")]).
48
49 r([t("?x", "is_grandparent", "?y"), t("?y", "is_parent", "?z")],
50     [t("?x", "is_great_grandparent", "?z")]).
51 r([t("?x", "is_a", "Man"), t("?x", "is_great_grandparent", "?y")],
52     [t("?x", "is_great_granddad", "?y")]).
53 r([t("?x", "is_a", "Woman"), t("?x", "is_great_grandparent", "?y")],
54     [t("?x", "is_great_grandmom", "?y")]).
55
56 r([t("?x", "is_great_grandparent", "?y")], [t("?y", "is_great_grandchild", "?x")]).
57 r([t("?x", "is_a", "Man"), t("?x", "is_great_grandchild", "?y")],
58     [t("?x", "is_great_grandson", "?y")]).
59 r([t("?x", "is_a", "Woman"), t("?x", "is_great_grandchild", "?y")],
60     [t("?x", "is_great_granddaughter", "?y")]).
61
62 r([t("?x", "is_sibling", "?y"), t("?x", "is_parent", "?z")],
63     [t("?y", "is_uncle_or_aunt", "?z")]).
64 r([t("?x", "is_uncle_or_aunt", "?y"), t("?x", "is_a", "Woman")],
65     [t("?x", "is_aunt", "?y")]).
66 r([t("?x", "is_uncle_or_aunt", "?y"), t("?x", "is_a", "Man")],
67     [t("?x", "is_uncle", "?y")]).
68 r([t("?x", "is_uncle_or_aunt", "?y"), t("?y", "is_a", "Woman")],
69     [t("?y", "is_niece", "?x")]).
70 r([t("?x", "is_uncle_or_aunt", "?y"), t("?y", "is_a", "Man")],
71     [t("?y", "is_nephew", "?x")]).

```

Полученная сеть для выбранного элемента



Вывод

Выполнив данную работу, я узнал про семантические сети и реализовал свою собственную. Также я освоил систему Protégé.