

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления
Кафедра Интеллектуальных информационных технологий

К защите допустить:

Заведующий кафедрой

_____ Д. В. Шункевич

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе

по дисциплине «Модели решения задач в интеллектуальных системах»:

**Система поддержки автоматической оценки цифровых
маммографических изображений на основе технологии
OSTIS**

БГУИР КР5 1–40 03 01 01 006 ПЗ

Студент гр.921704:

П. А. Белоус

Руководитель:

Д. В. Шункевич

Минск 2021

содержание

Перечень условных обозначений	5
Введение	6
1 АНАЛИЗ ПОДХОДОВ К РЕШЕНИЮ ПОСТАВЛЕННОЙ ЗАДАЧИ	7
1.1 Характеристика системы	7
1.2 Сравнительный анализ аналогов	8
1.3 Архитектура системы	11
1.4 Анализ пользователей системы	12
1.5 Анализ целевой аудитории и задач, которые система сможет решать	12
2 РАЗРАБОТКА АГЕНТОВ ДЛЯ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ДЛЯ МАММОЛОГА	13
2.1 Технологии и модели, используемые для проектирования баз знаний	13
2.2 Модель решателя	13
2.3 Используемые инструменты	14
2.4 Агент генерации отчета пациента	17
2.4.1 Спецификация агента генерации отчета пациента.	17
2.4.2 Входные данные	17
2.4.3 Алгоритм агента по генерации отчета пациента	18
2.5 Агент генерации отчета исследования	19
2.5.1 Спецификация агента генерации отчета исследования.	19
2.5.2 Входные данные	19
2.5.3 Алгоритм агента по генерации отчета исследования	19
Заключение	22

Перечень условных обозначений

В курсовой работе используются следующие условные обозначения:

БЗ — база знаний;

ИСС — интеллектуальная справочная система;

ПрО - предметная область;

SC — Semantic Code;

SCn - Semantic Code Natural;

SCs —Semantic Code String.

Введение

Целью данного курсового проекта было создание системы агентов для автоматической оценки метаданных цифровых изображений. А именно создание отчетов об исследовании пациента.

Для достижения поставленной цели был выделен перечень задач к выполнению:

- а) Спроектировать модель агента, установить связи между используемыми и генерируемыми сущностями.
- б) Реализовать агент.
- в) Протестировать агент на заданных и провалидированных тестовых данных.

Разрабатываемый агент является частью системы для маммографа. Общей задачей системы создаваемой нашей командой является создание отчетов об исследовании пациента, просчет вероятности заболеваний для упрощения работы врачей.

1 АНАЛИЗ ПОДХОДОВ К РЕШЕНИЮ ПОСТАВЛЕННОЙ ЗАДАЧИ

1.1 Характеристика системы

Исследования в сфере искусственного интеллекта и, в частности, прикладном применении искусственных нейронных сетей и их результаты являются очень полезным как теоретическим, так и практическим продуктом. Но на данном этапе все большую сложность вызывает внедрение таких результатов в реальные производственные процессы, которые не основаны или не предусматривают работу с такими технологиями. Разные задачи в одном и том же контексте решаются на разном множестве входных моделей, знаний и операций для их манипулирования. Также сложность представляется в том, что часто эти различные модели разрабатываются асинхронно - в своем темпе и направлении развития.

Одним из самых изучаемых и применяемых на сегодняшний день направлений в искусственном интеллекте является машинное обучение. Популярность данных методов вызвана двумя факторами: повышением производительности аппаратно-вычислительных средств и, как следствие, расширение возможностей теоретического вывода для различных задач в области математики.

Но какими бы качественными результатами не характеризовались такие методы, в распределении таких результатов и принятии решений над ними в конечном итоге участвует человек.

1.2 Сравнительный анализ аналогов

Для работы со знаниями в области искусственных нейронных сетей на данном этапе можно рассмотреть несколько справочных систем, которые могут являться аналогами нашей разрабатываемой системы, каждая из которых имеет свои достоинства и недостатки.

Для работы со знаниями в области искусственных нейронных сетей на данном этапе можно рассмотреть несколько справочных систем, каждая из которых имеет свои достоинства и недостатки.

1. Интернет-ресурс "Нейронные сети"; [1]
2. Интернет-ресурс "Tensorflow Playground"; [2]
3. Интернет-ресурс "Neural Network Console". [3]

Интернет-ресурс "Нейронные сети"

Данный ресурс представляет собой интерактивную площадку для изучения и обсуждения основ проектирования искусственных нейронных сетей. Изучение проходит по учебнику, а статьи - новости индустрии и автора ресурса - находятся в соответствующем разделе.

Достоинства ресурса:

1. Простота изложения понятий предметной области
2. Ориентированность на широкую аудиторию
3. Примеры работы нейросетей, доступные объяснения и ссылки на сторонние полезные ресурсы

Недостатки ресурса:

1. Почти полное отсутствие математического обоснования используемых методов
2. Неактуальность материала
3. Поверхностное затрагивание программных средств проектирования искусственных нейронных сетей

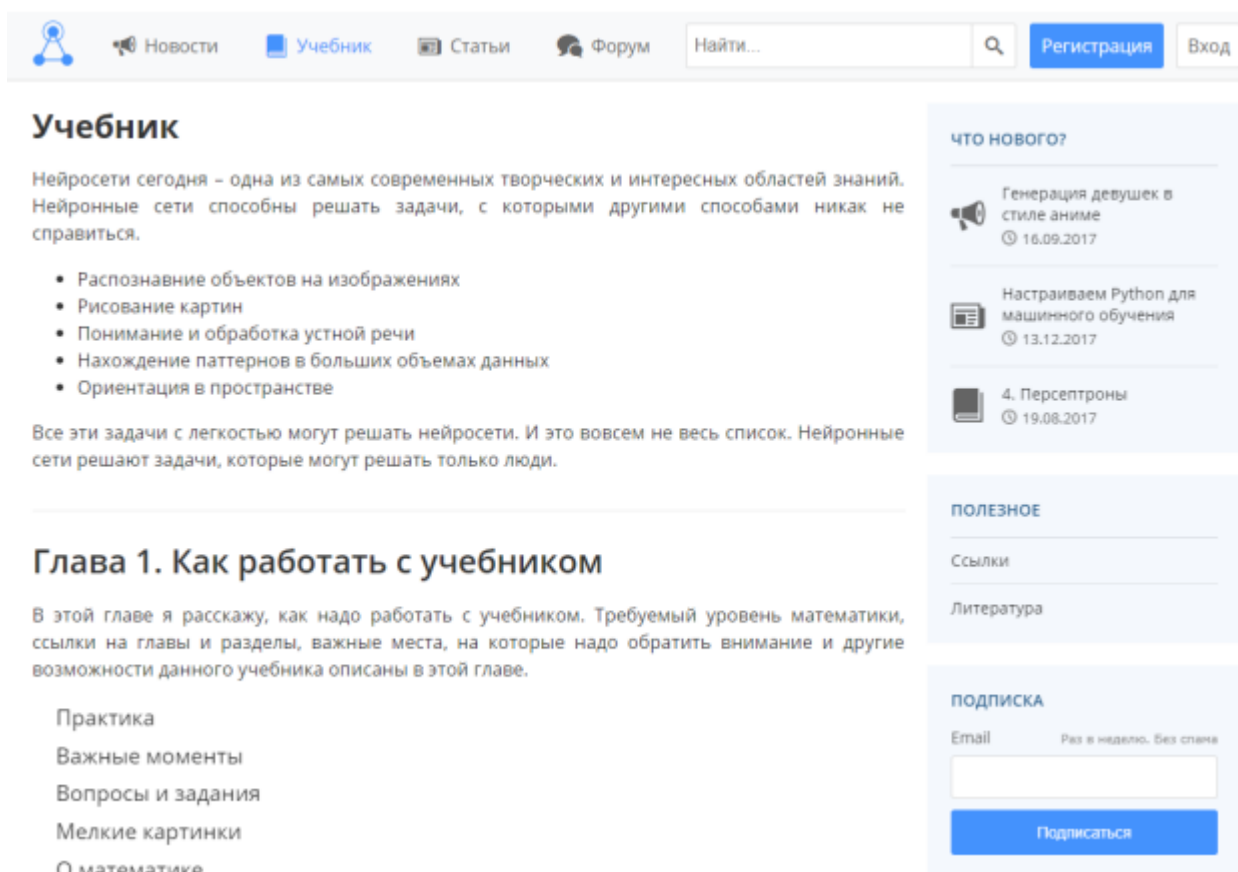


Рисунок 1.1 – Сайт neuralnet.info

Интернет-ресурс "Tensorflow Playground"

Цель данного ресурса - наглядно продемонстрировать различные подходы к проектированию нейронных сетей, показать вариации их архитектур и ее зависимость от задачи, которую нейронная сеть призвана решать. Площадка предоставляет визуальные пользовательские элементы для проектирования архитектуры нейронной сети в режиме реального времени.

Достоинства ресурса:

1. Пресеты архитектуры для решения определенных задач
2. Кастомизация параметров обучения
3. Визуализация результата работы сети

Недостатки ресурса:

1. Отсутствие глоссария понятий предметной области
2. Выполнение вычислений на машине пользователя
3. Отсутствие возможности ведения аккаунта пользователя, и, соответственно, сохранения сетей и параметров для обучения

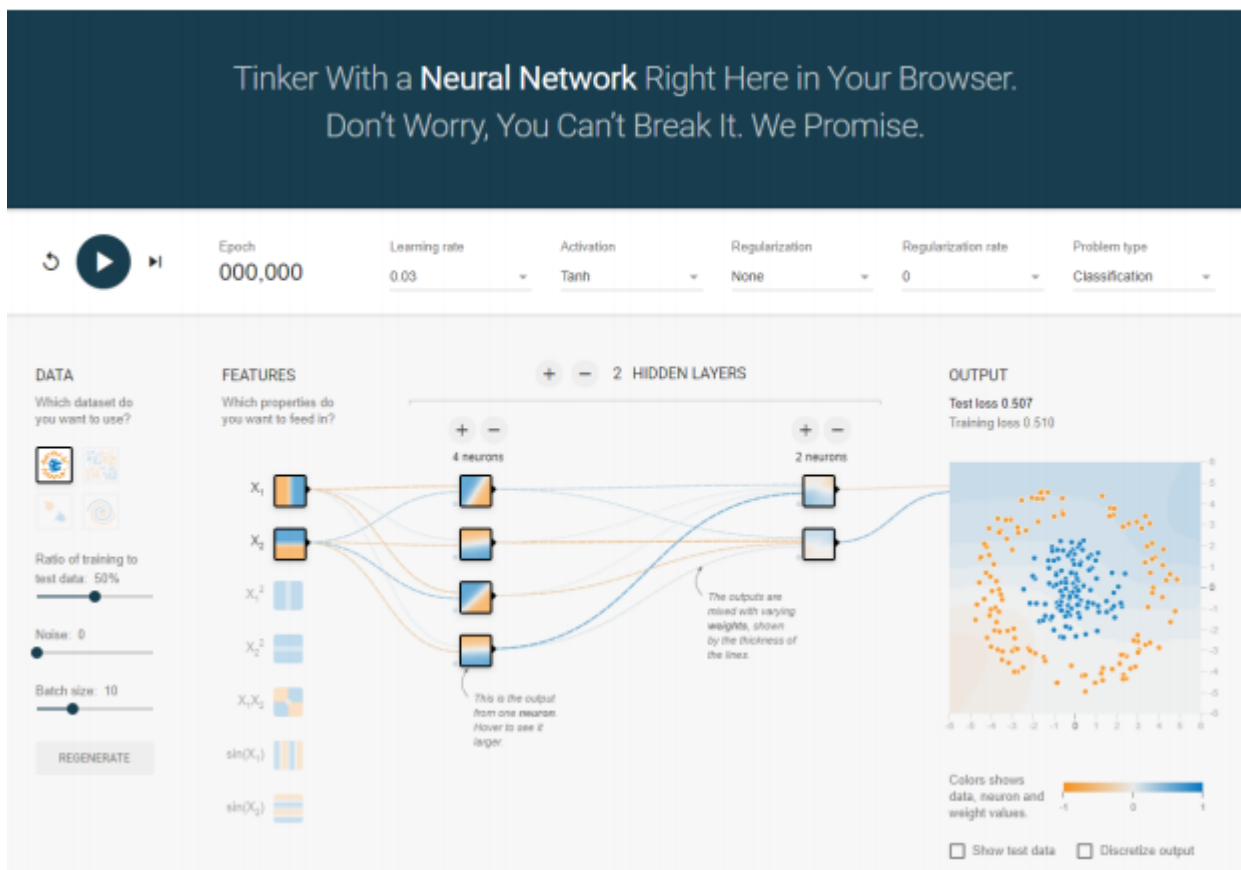


Рисунок 1.2 – Сайт playground.tensorflow.org

Интернет-ресурс "Neural Network Console" Данная платформа представляет собой набор различных инструментов проектирования, анализа архитектуры и статистики работы искусственных нейронных сетей. Позволяет производить вычисления как на машине пользователя, так и с помощью облачных сервисов.

Множество инструментов статистического анализа работы сетей • Простой drag-and-drop-интерфейс создания архитектуры сети • Наличие множества проектов-примеров для демонстрации вариативности задач, решаемых нейронными сетями

Достоинства ресурса:

1. Множество инструментов статистического анализа работы сетей
2. Простой drag-and-drop-интерфейс создания архитектуры сети
3. Наличие множества проектов-примеров для демонстрации вариативности задач, решаемых нейронными сетями

Недостатки ресурса:

1. Отсутствие глоссария понятий предметной области
2. Выполнение вычислений на машине пользователя
3. Неинтуитивный пользовательский интерфейс

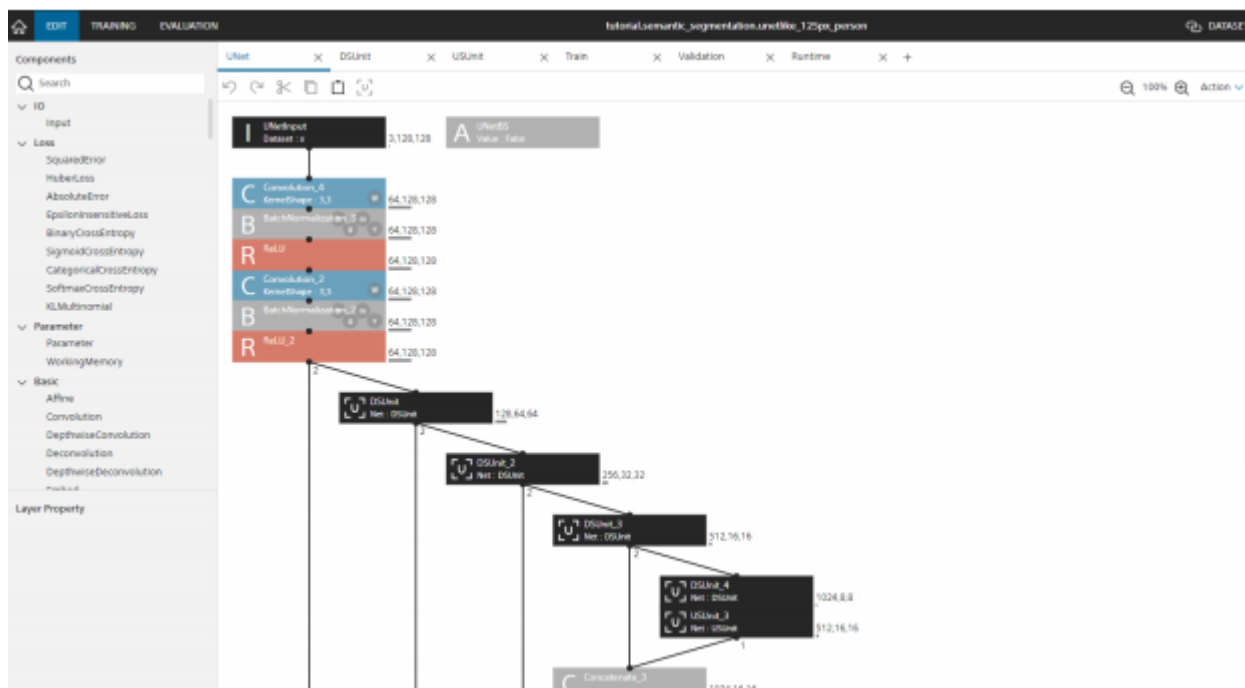


Рисунок 1.3 – Сайт dl.sony.com

1.3 Архитектура системы

Было принято решение в разработке базы делать описание исследования пациента опираясь на язык разметки, состоящий из тегов и значений. Также использовалась международная система BI-RADS для маммографических обследований. Исходя из особенностей исследования, было формализовано описание исследования двух латеральностей, у каждой из которых по 2 проекции MLO и CC. Так же описание структуры молочной железы соответствует протоколу ACR.

В качестве источника информации были использованы...

1.4 Анализ пользователей системы

Технология OSTIS тесно взаимосвязана со структуризацией и концепцией разрабатываемой БЗ. Данная технология не имеет широкого распространения в кругу обывателей. Для восприятия и понятия данной разработки необходимы знания не только в области OSTIS-технологий, а также представление, базовые знания о маммографических исследованиях.

Пользователь должен выполнить запрос в системе, при этом он может не углубляться в само исследование, а всего лишь знать ключевой элемент исследования. При это он получит полный отчет и вероятности наличия заболевания.

1.5 Анализ целевой аудитории и задач, которые система сможет решать

Основную долю пользователей системы должны составить врачи, использующие систему в качестве дополнительной помощи и экономии времени при анализе снимков.

Основной функционал системы:

- Просмотр отчета пациента и его исследования;
- Прогнозирование вероятности наличия заболевания;
- Отслеживание динамики заболеваний пациента.

2 РАЗРАБОТКА АГЕНТОВ ДЛЯ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ДЛЯ МАММОЛОГА

2.1 Технологии и модели, используемые для проектирования баз знаний

OSTIS

OSTIS (Open Semantic Technology for Intelligent Systems) — это проект, который направлен на создание массовой технологии компонентного проектирования интеллектуальных систем.

Технология, где знания представлены в виде семантически-структурированных графов, которые позволяют представлять сложноструктурированную информацию в виде графа, упрощать машинное кодирование посредством бинарных связей в качестве связей и узлов в качестве объектов. Решатель задач представляет собой графодинамическую sc-машину (память в качестве модели представления знаний использует семантическую сеть), состоящую из двух частей: графодинамической sc-памяти; системе sc-операций. Система операций является агентно-ориентированной и представляет собой набор sc-операций, условием инициирования которых является появление в памяти системы некоторой определенной конструкции. При этом операции взаимодействуют между собой через память системы посредством генерации конструкций, являющихся условиями инициирования для другой операции. При таком подходе становится возможным обеспечить гибкость и расширяемость решателя путем добавления или удаления из его состава некоторого набора операций. База знаний, как правило, содержит в себе описание множества понятий, примеры принадлежащих классу элементов. Также для некоторых понятий описаны правила их использования на практике. Для абсолютных понятий указаны надклассы, их содержащие, для относительных понятий указана область определения и область значения отношения. Для некоторых из понятий приведены синонимы.

2.2 Модель решателя

В рамках курсового проекта был разработан агент. Целью агента является создание отчета с информацией о пациенте, генерация и вывод итогового ответа.

Данный решатель был образован как отдельная система со своей базой знаний, машиной обработки знаний. Все знания, представленные на данный момент в системе, хранятся в виде SCs-конструкций.

Использовались такие инструменты как итераторы, шаблоны, генерация шаблона, а также вспомогательные методы на уровне контекста sc-памяти. 14 Данный проект (SC-machine C++ API) является частью проекта OSTIS [6] и предоставляет альтернативу написанию агентов обработки знаний на языке SCr. Исходные коды можно найти на GitHub.

СК-машина

Дом

Сложение

развитие

общий

API

Метапрограммирование

Агенты

шаблон

общий

питон

Нтр

инструменты


Лицензия

Изменения

Третье лицо

итераторы

Вы можете использовать 3-элементные итераторы и 5-элементные итераторы из кода C++. На рисунке ниже показана нумерация элементов в 3-элементных (слева) и 5-элементных (справа) итераторах.



Для них обоих у вас есть два подхода. Первый подход - использовать простой цикл while. Это подходит, когда вам нужно разорвать этот цикл в зависимости от логики:

```
...
// example of f_a_a iterator
ScIterator3Ptr it3 = ctx.Iterator3(
    set,
    ScType::EdgeAccessConstPosPerm,
    ScType::Unknown);
while (it3->Next())
{
    // to get values use it3->Get(index), where index in range [0; 2]
    ... // there is you code
}

...
// example of f_a_a_f iterator
ScIterator5Ptr it5 = ctx.Iterator5(
    objAddr,
    ScType::EdgeDCommonConst,
    ScType::NodeConstAbstract,
    ScType::EdgeAccessConstPosPerm,
    KeyNodes::nrel_mass);

while (it5->Next())
{
    // to get values use it5->Get(index), where index in range [0; 4]
    ... // there is you code
}
```

Рисунок 2.2 – Раздел "Итераторы" в документации к SC-machine C++ API

Поскольку для корректной установки модулей в существующую базу знаний необходимо использовать CMake-файлы, то более удобной и предпочтительной является разработка в CLion [9], но VS Code [10] также удобна в использовании. Несмотря на это всегда можно использовать блокнот.



Рисунок 2.3 – Среда разработки "Блокнот"

Qt Creator (ранее известная под кодовым названием Greenhouse) — кроссплатформенная свободная IDE для разработки на C, C++ и QML. Разработана Trolltech (Digia) для работы с фреймворком Qt. Включает в себя графический интерфейс отладчика и визуальные средства разработки интерфейса. Программа была использована для работы с файлами написанных на языке c++.

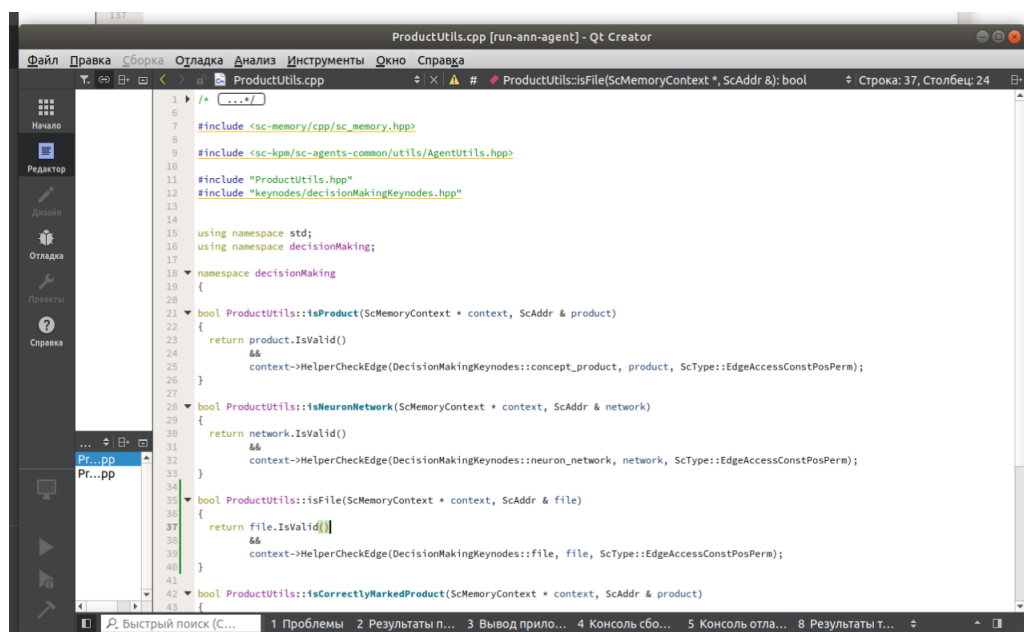


Рисунок 2.4 – Среда разработки "Qt creator"

Рассмотрим два подхода к реализации агентов в рамках системы OSTIS на сегодняшний день.

SCp - это графовый язык процедурного программирования, предназначенный для эффективной обработки семантических сетей с теоретикомножественной интерпретацией, представленных в SC-коде. Язык SCP является языком параллельного асинхронного программирования. Представление данных для текстов языка SCp построено на SC-коде: SCs и SCg. Язык SCp сам построен на основе SC-кода, вследствие чего SCp-программы сами по себе могут входить в состав данных SCp-программ, в т.ч. по отношению к самим себе.

Агентные SCp-программы представляют собой частный случай SCp-программ и представляют собой реализации программ агентов обработки знаний, имеют жестко фиксированную структуру множества входных параметров.

В рамках данного семестра был разработан агент:

- Агент генерации отчета информации о пациенте;

2.4 Агент генерации отчета пациента

2.4.1 Спецификация агента генерации отчета пациента.

Задачей агента является составление отчета пациента. Данный агент иницируется при условии появления в памяти вопросной конструкции, соответствующей запросу генерации отчета пациента.

2.4.2 Входные данные

Входным аргументом для данной системы является узел пациента, который далее используется для составления отчета.

```
|concept_study->study1;;  
..patient<-concept_patient;;  
study1=>nrel_patients_study:..patient;;  
..patient=>nrel_patients_hormonal_status:..hormonal;;  
concept_menopause->..hormonal;;  
concept_menopause<-concept_hormonal_status;;  
..patient=>nrel_age:[034Y](*<-concept_age_string;*);;  
  
study1=>nrel_image:..medical_image_1;;  
study1=>nrel_image:..medical_image_2;;  
study1=>nrel_image:..medical_image_3;;  
study1=>nrel_image:..medical_image_4;;
```

Рисунок 2.5 – Фрагмент информации о пациенте

2.4.3 Алгоритм агента по генерации отчета пациента

Поиск всех необходимых конструкций производится с помощью пяти или трехэлементных итераторов, а каждое действие сопровождается проверкой: отсутствие необходимого узла или несоответствие его целевому типу - все это является ошибками в работе агента и они логируются, после чего агент завершает свою работу. Для инициирования агента необходимо добавить исходящую дугу из узла запуска агента к произвольному узлу, валидация которого как процесс входит в работу проектируемого агента.

1. Проверяем, является ли узел - конец дуги-инициатора - узлом класса пациента и иницируем пациента.
2. Создаем переменную типа строка содержащую отчет пациента
3. С помощью пятиэлементного итератора находим узел гормонального статуса пациента

```
ScIterator5Ptr iterator5 = ms_context->Iterator5(patient, ScType::EdgeDCommonConst, ScType::Unknown, ScType::EdgeAccessConstPosPerm, Keynodes::nrel_patients_hormonal_status);  
ScAddr hormonal;  
if (iterator5->Next())  
{  
    hormonal = iterator5->Get(2);  
}
```

Рисунок 2.6 – Получение гормонального статуса

4. С помощью трехэлементного итератора находим класс к которому принадлежит гормональный статус пациента
5. Узнаем идентификатор узла класса гормонального статуса пациента и добавляем его в отчет

```
ScAddr hormonal_status;  
ScIterator3Ptr iterator3 = ms_context->Iterator3(ScType::Unknown, ScType::EdgeAccessConstPosPerm, hormonal);  
while (iterator3->Next())  
{  
    if (isHormonalStatus(ms_context.get(), iterator3->Get(0)) == true) {  
        hormonal_status = iterator3->Get(0);  
    }  
}
```

Рисунок 2.7 – Класс гормонального статуса

6. С помощью пятиэлементного итератора находим узел возраста пациента
7. Узнаем идентификатор узла возраста пациента и добавляем его в отчет

```
ScIterator5Ptr iterator5age = ms_context->Iterator5(patient, ScType::EdgeDCommonConst, ScType::Unknown, ScType::EdgeAccessConstPosPerm, Keynodes::nrel_age);  
ScAddr age;  
if (iterator5age->Next())  
{  
    age = iterator5age->Get(2);  
}  
  
ScAddr answer = ms_context->CreateNode(ScType::NodeConstStruct);
```

Рисунок 2.8 – Получение возраста пациента

8. Создаем узел типа link и записываем в него отчет пациента

```
string hormonal_status_str = CommonUtils::getIdtfValue(ms_context.get(), hormonal_status, Keynodes::nrel_main_idtf );
string age_str = CommonUtils::getIdtfValue(ms_context.get(), patient, Keynodes::nrel_age );
string patien_rep_str = "hormonal status - " + hormonal_status_str + "\n age - " + age_str;

SC_LOG_INFO(patien_rep_str);
ScAddr answer_link = ms_context->CreateLink();
string value = patien_rep_str;

ScStreamPtr stream;
stream.reset(new ScStream(value.c_str(), (sc_uint32) value.size(), SC_STREAM_FLAG_READ ));
ms_context->SetLinkContent(answer_link, *stream);
```

Рисунок 2.9 – Составление отчета о пациенте

9. Создаем направленную бинарную дугу между узлом пациента и узлом с отчетом пациента с отношением отчет пациента

10. Создаем дуги между узлом ответа и узлом пациента, с узлом содержащий отчет, с узлом отношения отчет пациента, с дугами, созданными в предыдущем пункте.

```
ScAddr edge_answer = ms_context->CreateEdge(ScType::EdgeDCommonConst, patient, answer_link);
ScAddr edge_edge_answer = ms_context->CreateEdge(ScType::EdgeAccessConstPosPerm, Keynodes::nrel_patient_report, edge_answer);

ms_context->CreateEdge(ScType::EdgeAccessConstPosPerm, answer, answer_link);
ms_context->CreateEdge(ScType::EdgeAccessConstPosPerm, answer, patient);
ms_context->CreateEdge(ScType::EdgeAccessConstPosPerm, answer, edge_answer);
ms_context->CreateEdge(ScType::EdgeAccessConstPosPerm, answer, edge_edge_answer);
ms_context->CreateEdge(ScType::EdgeAccessConstPosPerm, answer, Keynodes::nrel_patient_report );

SC_LOG_INFO("-----patient end-----");
AgentUtils::finishAgentWork(ms_context.get(), questionNode, answer);
return SC_RESULT_OK;
```

Рисунок 2.10 – Формирование ответа

11. Заканчиваем работу агента, выводя узел ответа.

2.5 Агент генерации отчета исследования

2.5.1 Спецификация агента генерации отчета исследования.

Задачей агента является составление отчета исследования. Данный агент инициируется при условии появления в памяти вопросной конструкции, соответствующей запросу генерации отчета исследования.

2.5.2 Входные данные

Входным аргументом для данной системы является узел исследования, который далее используется для оставления отчета.

2.5.3 Алгоритм агента по генерации отчета исследования

Поиск всех необходимых конструкций производится с помощью пяти или трехэлементных итераторов, а каждое действие сопровождается

проверкой: отсутствие необходимого узла или несоответствие его целевому типу - все это является ошибками в работе агента и они логируются, после чего агент завершает свою работу. Для инициирования агента необходимо добавить исходящую дугу из узла запуска агента к произвольному узлу, валидация которого как процесс входит в работу проектируемого агента.

1. Проверяем, является ли узел - конец дуги-инициатора - узлом класса исследования и иницируем исследование.

2. Находим с помощью итератора на пять 4 узла изображения

3. Создаем строковую переменную отчета

4. Для каждого изображения находим с помощью итератора на пять узел метаданных

5. Находим с помощью итератора на пять узел проекции

6. Находим идентификатор узла проекции и записываем его в отчет

7. Находим с помощью итератора на пять узел латеральности

8. Находим идентификатор узла латеральности и записываем его в отчет

9. Находим с помощью итератора на пять узел содержащий все артефакты

10. Запускаем агент подсчета вероятности и ждем ответ

11. Записываем ответ агента в отчет

12. Для каждого артефакта находим с помощью итератора на три его тип

13. Находим идентификатор узла типа артефакта и записываем его в отчет

14. Находим с помощью итератора на пять узел формы артефакта

15. Находим с помощью итератора на три тип формы артефакта

16. Находим идентификатор узла типа формы артефакта и записываем его в отчет

17. Находим с помощью итератора на пять узел краев артефакта

18. Находим с помощью итератора на три тип краев артефакта

19. Находим идентификатор узла типа краев артефакта и записываем его в отчет

20. Находим с помощью итератора на пять узел плотности артефакта

21. Находим с помощью итератора на три тип плотности артефакта

22. Находим идентификатор узла типа плотности артефакта и записываем его в отчет

23. Находим с помощью итератора на пять узел размера артефакта

24. Находим идентификатор узла размера артефакта и записываем его в отчет

25. Находим с помощью итератора на пять узел распространения артефакта

26. Находим с помощью итератора на три тип распространения арте-

факта

27. Находим идентификатор узла типа распространения артефакта и записываем его в отчет

28. Создаем узел типа ссылка и записываем в него отчет

29. Создаем направленную бинарную дугу между узлом исследования и узлом с отчетом с отношением отчет исследования

30. Создаем дуги между узлом ответа и узлом исследования, с узлом соержащий отчет, с узлом отношения отчет исследования, с дугами, созданными в предыдущем пункте.

31. Заканчиваем работу агента, выводя узел ответа.

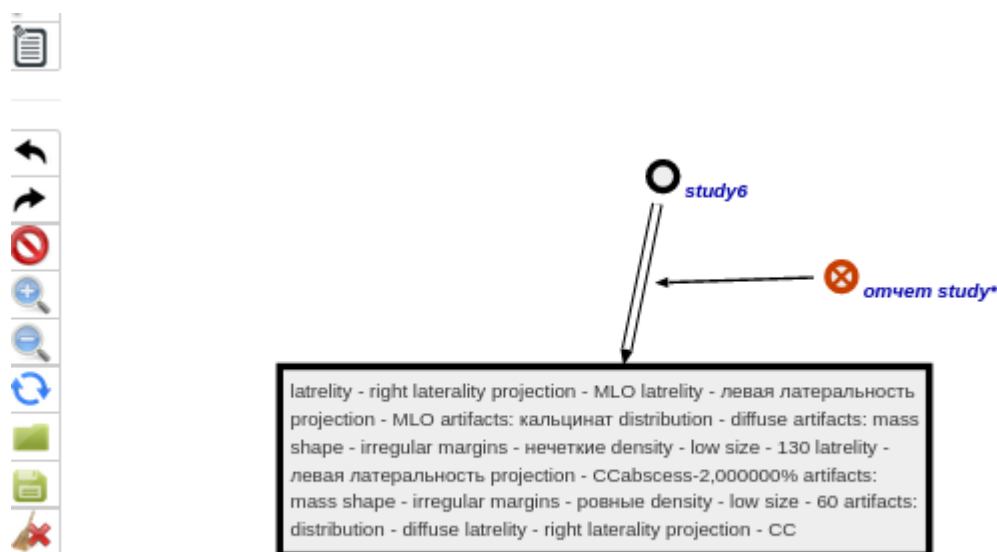


Рисунок 2.11 – выход работы агента

Заключение

В рамках курсового проекта были разработан агент генерации отчета. В перспективе будет разработан агент взаимодействия сервера на котором будет находиться нейронная сеть по распознаванию артефактов на изображениях, агент проверки правильности расположения молочной железы на снимке, агент генерации формализации исследования.

Во время работы над агентом были выделены как достоинства, так и недостатки работы с SC-machine C++ API. Достоинствами, к примеру, явились почти полная интуитивность набора методов и сущностей. Недостатком же является сложность регистрации модулей и файлов ключевых узлов в системе.

Список использованных источников

- [1] Нейронные сети [Электронный ресурс]. — Режим доступа: <https://neuralnet.info/>.
- [2] A Tensorflow Playground [Электронный ресурс]. — Режим доступа: <https://playground.tensorflow.org/>.
- [3] A Neural Network Console [Электронный ресурс]. — Режим доступа: <https://dl.sony.com/>.
- [4] В.А., Головки. Нейроинтеллект теория и практика. Часть 1 / Головки В.А. — Брест, 1999. — 260 с.
- [5] Толковый словарь по искусственному интеллекту [Электронный ресурс]. — Режим доступа: <http://www.raai.org/library/tolk/aivoc.html>. — Дата доступа: 03.05.2020.
- [6] Изданные материалы OSTIS-2018 [Электронный ресурс]. — Режим доступа: <https://conf.ostis.net/wp-content/uploads/2019/04/Pro...> — Дата доступа: 03.05.2020.
- [7] Анализ технологии OSTIS [Электронный ресурс]. — Режим доступа: <http://refleader.ru/polatyujgyfsqas.html>. — Дата доступа: 03.05.2020.
- [8] Голенков В. В., Гулякина Н. А., Елисеева О.Е. Семантическая модель сложноструктурированных баз данных и баз знаний / Голенков В. В., Гулякина Н. А., Елисеева О.Е. — Мн., БГУИР, 2004. — 263 с.
- [9] Голенков В. В. Гулякина Н. А., Гракова Н. В. Давыденко И. Т. Шункевич Д. В. Искусственный интеллект. Курсовое проектирование : пособие / Гракова Н. В. Давыденко И. Т. Шункевич Д. В. Голенков В. В., Гулякина Н. А. — 2017. — 62 с.
- [10] Гаврилова Т.А., Хорошевский В.Ф. Базы знаний интеллектуальных систем / Хорошевский В.Ф. Гаврилова Т.А. — 2000. — 384 с.
- [11] Метасистема IMS [Электронный ресурс]. — Режим доступа: <http://ims.ostis.net>. — Дата доступа: 03.05.2020.
- [12] Мак-Каллок У. С., Питтс В. Логическое исчисление идей, относящихся к нервной активности / Питтс В. Мак-Каллок У. С. — 1956. — 363 с.
- [13] Электронный учебник Statsoft. <http://statsoft.ru/home/textbook/modules/stneunet.html>.