***ФГБОУ ВО «Кубанский государственный аграрный университет имени***

***И.Т. Трубилина»,***

***Российская Федерация***

***Колесников Роман Юрьевич, ПИ2102***

***roman563412@gmail.com***

**Работа в Эйдос. Описательные шкалы и градации.**

**Заголовок**

Обзор типов шкал данных, этапов анализа и устранения неполадок в системе ИДС

**Резюме**

В видео лектор объясняет различные типы шкал данных, используемых в системе анализа данных ИДС (Интеллектуальная Система Данных), демонстрирует этапы работы с системой и помогает студентке решить проблему с установкой.

1. Типы шкал данных:

Порядковая шкала (Размер 1): Градации имеют осмысленный порядок (маленький, средний, большой), но арифметические операции невозможны. Можно определить "больше/меньше".

Числовая шкала (Размер 2): Данные представлены числами (или диапазонами с числовыми средними), возможны арифметические операции (сложение, вычитание и т.д.).

Логическая (Булева) шкала (Наличие экрана/кнопок): Имеет два значения (Да/Нет, Истина/Ложь). Важна правильная разработка анкет, чтобы респондент мог явно выбрать один из вариантов.

Номинальная шкала (Форма, Цвет): Категории без естественного порядка, возможны только операции эквивалентности/неэквивалентности. Сортировка по алфавиту не дает осмысленного порядка.

2. Представление данных в системе ИДС:

Исходные данные кодируются с использованием шкал.

Система отображает:

Название источника данных.

Коды категорий объектов (согласно классификационным шкалам).

Коды значений свойств объектов (согласно описательным шкалам).

Данные можно просматривать в табличном виде, похожем на Excel, где вместо текстов и чисел отображаются их коды-ссылки на справочники шкал.

3. Возможности и ограничения системы ИДС:

ADS Off (локальная версия): Ограничение размера каждой базы данных (заголовки, классы, признаки) – 2 ГБ.

ADS On (с сервером): Ограничения на размер баз данных снимаются. Лектор тестировал систему на 8 млн примеров и файлах баз данных до 400 ГБ, теоретический предел – миллионы терабайт. Ограничения связаны в основном с физическими возможностями компьютера.

Система может обрабатывать большое количество объектов, классов (до 2000-2305) и признаков (тысячи).

Возможно использование GPU для ускорения расчетов (верификации моделей) в тысячи раз.

4. Этапы автоматизированного системно-когнитивного анализа (АСКА) в ИДС:

Когнитивно-целевая структуризация: Постановка задачи, определение объекта моделирования, факторов и целевых состояний.

Формализация предметной области: Разработка шкал (справочников) и кодирование исходных данных.

Синтез и верификация моделей: Выявление причинно-следственных связей (статистические и системно-когнитивные модели), проверка их достоверности. Используется режим 3.5.

Верификация: Трудоемкий процесс распознавания/идентификации объектов обучающей выборки для оценки точности модели.

Бутстреппинг: Возможность использовать часть выборки для синтеза, а другую – для верификации, или проверять достоверность на части выборки для оценки времени полного расчета.

Система позволяет удалять наименее достоверные результаты распознавания для экономии места.

Решение задач: Использование достоверных моделей для идентификации, прогнозирования, принятия решений и исследования предметной области (режим 6.4).

5. Устранение неполадок при установке:

Студентка Арина сталкивается с ошибками при запуске (не найдены библиотеки DLL, ошибка доступа к интернет-ресурсу).

Рекомендации лектора:

Удалить старую версию и скачать последнюю с официального сайта (а не с Яндекс.Диска).

Убедиться, что папка установки и путь к ней не содержат пробелов и кириллицы, и это не папка "Загрузки" или "Рабочий стол".

Запускать систему из основной папки (например, C:\IDSX), а не из подпапок (C:\Data\IDSX).

Лектор отправляет недостающие библиотеки и исполняемый модуль через чат.

Ошибка доступа к сайту может быть связана с настройками безопасности ОС или брандмауэра, запрещающими программе выход в интернет.

Детальная расшифровка текста

1. Введение и типы шкал данных

1.1. Источник данных и неточности

Исходных данных вводили студенты. И я оставил, как они делали там неточности, допускали, я так и оставил, чтобы вам их показывать, эти неточности.

Понятно, да?

Да.

1.2. Порядковая шкала (Размер 1)

Вот. Теперь смотрим.

Вот эта шкала размер 1, она является порядковой шкалой, потому что при сортировке по алфавиту, градация этой шкалы расположена в разумном порядке, а именно в порядке увеличения размера.

Вот этого шкала 1 размер, да? Маленький, средний, под руку, большой. И между этими градациями есть отношение больше-меньше, но нет возможности осуществлять какие-либо арифметические операции, то есть их нельзя складывать, вычитать. То есть вот мы знаем, что средний больше, чем маленький, но насколько, не знаем.

1.3. Числовая шкала (Размер 2)

Вот размер 2 – это тоже по смыслу такая же шкала, как и размер 1, но степень формализации выше. Потому что это числовая шкала, найдена там минимальное-максимальное значение и разделено на 12, как мы задали там в процессе ввода данных исходных, числовых диапазонов равного размера. Вот. И каждому числовому диапазону соответствует среднее значение. И эти средние значения являются числами, то есть их можно складывать, вычитать, умножать, делить. То есть если мы, допустим, возьмём из третьей… из среднего третья градация, вычтем среднее второй градации, то мы узнаем разницу между ними.

1.4. Логическая (Булева) шкала и дизайн анкет

Вот. Ну, наличие экрана – это логическая шкала, True/False, есть/нет. Да/нет. Истина/ложь.

Вот. И здесь я хотел бы ваше внимание обратить вот на такой момент. Иногда встречаются анкеты, в которых спрашивают мнение населения там или студентов по каким-то вопросам. И, допустим, может быть какой-то вопрос в анкете. Ну, допустим, согласны вы ли с тем, что нужно при дистанционном обучении объединить все группы потока по какой-то дисциплине, когда у нас практическое занятие. Вот то, что сейчас вопрос возникал. И, значит, те, кто согласен, ставят птичку, а те, кто не согласен, они там птичку не ставят.

И вот анкета из таких вопросов состоит. Как вы считаете, это нормальная анкета или там что-то не то? Какая-то есть погрешность в её разработке?

Если ты не выберешь ничего, ну, не поставишь галочку, то может кто-то другой за тебя поставит?

Правильно, правильно.

Я об этом говорил вам, нет?

Нет.

Нет, вы сами догадались. Вот. Так вот, да, конечно, нужно предусмотреть все варианты ответа и сделать анкету так, чтобы респондент имел возможность поставить галочку, эту птичку против того варианта ответа, который он, с которым он согласен.

Вот в системе ИДС поэтому так и сделано. Наличие экрана – есть, нет. Наличие кнопок – есть, нет. И так далее.

1.5. Номинальная шкала (Форма, Цвет)

Форма – это номинальная шкала. То есть при сортировке по алфавиту мы здесь не видим никакого разумного порядка, осмысленного. Такого, чтобы там какое-то свойство, значение какого-то свойства возрастало, к примеру. Как вот в шкале размер.

А теперь давайте я вас немножко поспрашиваю. Вот шкала цвет – это какая шкала? Номинальная, порядковая или числовая?

Ну, по сути, каждый цвет же закодирован своим кодом. Возможно, это числовая?

Нет.

Ну хотя, может, порядковая?

Нет.

Значит, я когда рассказывал про исходные данные, к этому вот интерфейсу, где мы вводим данные из таблицы, я подробно рассказывал про шкалы, и там это и написано в хелпе в этом. Ну, в общем, если между градациями шкалы, вернее так, на шкале можно осуществлять только операции эквивалентности/неэквивалентности, то это номинальная шкала.

Вот, допустим, список студентов – это номинальная шкала. То есть там каждый студент соответствует или не соответствует строчке в списке.

1.6. Сравнение шкал: Порядковая vs Числовая

Вот. Порядковая шкала. Значит, кроме операций эквивалентности, ещё есть операции больше-меньше.

Вот. Пример – размер 1, шкала вот здесь, это порядковая шкала. То есть мы можем сказать, что 2, 4 – средний размер больше, чем маленький, а большой больше, чем под руку.

Вот. Но насколько больше, мы из этой порядковой шкалы не видим этого.

Также порядковой шкалой является такая шкала возраст, возраста, которая не в годах возраст, а в языке ещё есть такие слова: младенец, там, ребёнок, юноша, девушка, мужчина, женщина, старик, старуха, там, допустим, да? Понимаете, да?

То есть понятно, что мужчина старше, чем юноша, понятно, что старик старше, чем мужчина.

Ну насколько старше, там не поймёшь из такой шкалы.

А когда ты говоришь, допустим, там, вот тебе там сколько лет? Ты говоришь: 15 там или 20. И вот кто-то говорит, сколько ему лет, можно вычесть и сказать: да он старше на 5 лет, там, или на 7 лет, или на 8 лет там. Понятно, да?

Вот. Это, значит, если мы можем сказать только то, что больше или меньше, то это порядковая шкала. А если ещё можем сказать, насколько больше-меньше, тогда это числовая шкала.

1.7. Итог по типам шкал

И вот у нас здесь номинальная шкала – цвет. Вот. Порядковая – размер 1. Числовая – размер 2.

Ещё есть у нас так называемые категориальные переменные. Это логическая шкала. Она тоже является текстовой по типу данных. Но смысл её в том, что градации являются, значит, ответами истинным и ложным на вопрос, который является названием шкалы. То есть наличие кнопок: если есть, значит это да, там, если нет – нет, ложь. Понятно, да? То есть это булева переменная, двоичная.

2. Представление данных в системе ИДС

2.1. Кодирование и структура отображения

Вот с помощью этих вот шкал закодированы исходные данные. И получились у нас вот здесь вот, видите, в верхнем окне название источника данных. В нижнем левом окне – коды в соответствии с классификационными шкалами тех категорий, к которым относятся данные объекты выборки. А в правом нижнем окне – коды значения свойств в соответствии с описательными шкалами.

И есть… Да, вот в этой форме особых нет ограничений на количество объектов, кодов классов, к которым они относятся, кодов признаков, которые им соответствуют. Особых ограничений нет.

2.2. Альтернативный табличный вид

Мы можем эту же самую таблицу исходных данных посмотреть в таком виде вот. Очень похожем на тот, в котором она в экселевском файле. Я вам этот вид показывал, ребят, нет? Скажите?

Нет.

Нет. Тогда сейчас покажу, как она связана, как она соотносится с экселевской табличкой.

Ну, наверное, вот так вот сделаем. Вот экселевская. Экселевскую лучше вверх поместить. Исходные данные. А здесь вот мы видим, смотрите. Я вот так поставлю, чтобы как-то оно примерно соответствовало.

И что мы видим? Мы видим, что у нас в обучающей выборке, которую мы видим в таком вот табличном виде, у нас все те же самые, в общем-то, колонки, которые в экселевском файле. Но теперь вместо текстов и чисел, которые в этих колонках, у нас коды, коды этих текстов и этих чисел. И, то есть, ссылки на справочники классификационных и описательных шкал. Там, где вот у нас жёлтый фон – это классификационные шкалы, там, где белый фон – это описательные шкалы.

И мы видим, тут колоночка A, B, C, да? И тут у нас первая, вторая, третья. Они точно соответствуют тем, которые в Excel по порядку. То есть здесь вот внизу написаны прямо вот те номера, которые мы вводим в этом интерфейсе. Там, со второй по третью – классификационные шкалы, с четвёртой по двенадцатую – описательные шкалы. Прямо вот мы их и вводим, эти вот номера колоночек.

Но этот интерфейс, вернее, в таком виде мы исходные данные, обучающую выборку, закодированную уже с помощью справочников, можем увидеть только тогда, когда для этого кодирования использовался интерфейс 2.3 2.2. А в системе ИДС таких интерфейсов шесть аналогичных, но имеющих различные особенности, отличающиеся тем, какие типы данных вводятся с помощью этих интерфейсов из внешних источников данных.

3. Этапы анализа и возможности системы

3.1. Следующие шаги после ввода данных

Дальше теперь, ребята, вопрос возникает такой: а что мы делаем дальше? То есть мы ввели данные исходные в систему. При этом было… были созданы справочники классификационных и описательных шкал и градаций и обучающая выборка. А что дальше делать? Кто знает?

Кто знает – поднять руку.

Кто ещё знает?

Не знаю.

А?

Не знаю.

А как узнать?

У вас.

Ну да. Мы тут такой способ, знаете, не всегда возможно это сделать.

3.2. Использование справочной системы (Help)

А вот в самой системе тоже есть хелпы, то есть поясняющие режимы, в которых написано, что делать дальше. Я про это вам говорил. В режиме 1.3 есть хелп, в котором перечислена красная нить, так сказать, логика обработки данных, информации, знаний в системе и решения различных задач. И есть ещё… там и написано в этом хелпе, что есть режим 6.4, в котором тоже это приводится, эта логика.

3.3. Общая логика АСКА

Вот, что система, собственно, что обеспечивает? Она обеспечивает ввод данных в систему, преобразование данных в информацию, информацию в знания и решение с помощью этих знаний различных задач: задачи идентификации, прогнозирования и принятия решений, а также исследование, моделирование предметной области путём исследования модели этой предметной области. Это у нас хелп, который описан в режиме… крутится в режиме 6.4. Он весь предназначен для этого.

3.4. Этапы Автоматизированного Системно-Когнитивного Анализа (АСКА)

И вот мы видим, что есть определённые этапы обработки данных, информации, знаний. И эти этапы я бы хотел, чтобы вы осознали это, запомнили. То есть, ну, как бы переварили через свою психическую эту систему то, что я сейчас скажу. Вот эти вот зелёные блоки большие: формализация предметной области, синтез и верификация моделей, решение задач. Это вот и есть этапы автоматизированного системно-когнитивного анализа.

То есть первый этап – это когнитивно-целевая структуризация предметной области. На этом этапе мы решаем, какую задачу мы будем решать. То есть постановка задачи. То есть мы определяемся с тем, что мы рассматриваем как объект моделирования, что мы рассматриваем как факторы, которые влияют на объект моделирования, и в какие будущие состояния объект моделирования переходит под действием этих факторов. Это вот и есть постановка задачи, которая называется когнитивно-целевая структуризация предметной области.

Потом идёт этап формализация предметной области. На этом этапе мы разрабатываем справочники классификационных и описательных шкал и градаций. А потом с помощью этих справочников кодируем исходные данные.

Таким образом мы подготавливаем полностью всё необходимое для того, чтобы перейти к синтезу и верификации моделей.

3.5. Синтез и верификация моделей

Я вам, помните, рассказывал, что информация – это осмысленные данные, а смысл – это знание причинно-следственных зависимостей. И эти причинно-следственные зависимости существуют между событиями, которые отражены в эвентологической базе данных, то есть в базе событий или в обучающей выборке. И взаимосвязи эти отражаются в моделях. То есть модели статистические и системно-когнитивные модели и отражают причинно-следственные связи между событиями.

Так вот, мы уже базу событий создали, а причинно-следственные связи ещё не выявили. То есть следующий этап – это выявление причинно-следственных связей между событиями. Это осуществляется в режиме 3.5 и представляет собой, собственно говоря, синтез и верификацию моделей. То есть модели создаются статистические, классические модели, и модели знания, интеллектуальные модели на их основе.

Потом проверяются эти модели на достоверность обязательно. Потому что если модель недостоверная, нет достоверной модели, то никакие задачи решать нельзя с помощью такой модели. Поэтому мы определяем достоверность модели, выбираем наиболее достоверную. И если она есть такая, которая приемлемая, имеет достоверность, то тогда мы решаем задачи.

И здесь у нас описано, какие задачи, в каких режимах они решаются. И соотношение между различными задачами здесь ниже на жёлтом фоне написано, что на самом деле эти задачи, они взаимосвязаны друг с другом. То есть нельзя рассматривать их как совершенно друг от друга не зависящие.

Таким образом, у нас следующий этап, который мы должны выполнить – это этап синтез и верификация моделей. И там написано было, в каком режиме мы это делаем. Мы это делаем в режиме 3.5.

3.6. Режим 3.5: Детали синтеза и верификации

В этом режиме мы видим, что… какие модели создаются. Мы видим, что можно использовать такой подход, когда осуществляется синтез модели и проверка их на достоверность. Или можно сделать только синтез модели, только верификацию.

Значит, зачем это сделано, ребята? Значит, дело в том, что синтез модели – это процесс сравнительно быстрый, даже когда очень большое количество объектов обучающей выборки. Он идёт там, ну, минуты там, ну, может, полчаса там максимум. Это когда их там миллионы.

Вот. А вот верификация модели – это очень трудоёмкая в вычислительном отношении операция, потому что это осуществляется путём распознавания или идентификации каждого объекта обучающей выборки. Их может быть много, я уже говорил. Ну, допустим, их миллион, если экселевский файл используется в качестве исходных данных, то там до миллиона может быть объектов обучающей выборки. Ну, тоже довольно прилично.

И представьте себе, каждый объект обучающей выборки может относиться к одному из 2.000 классов. В системе ИДС можно до 2.305 классов использовать. И он может быть описан, скажем, ну, 10.000 признаков, этот объект обучающей выборки. И вот, чтобы его сравнить с одним классом, для этого нужно просуммировать количество информации, которое содержится во всех его признаках о принадлежности к каждому из классов. То есть у нас получается такой тройной цикл. Сначала цикл по всем объектам обучающей выборки, которые используются для тестирования модели, верификации. Потом внутри него цикл по классам, по классам. И ещё внутри – по признакам. И мы получаем в результате, когда цикл по признакам завершается, мы получаем, какое количество информации содержится в этом… в этой системе признаков о принадлежности к данному классу, текущему. И так мы делаем для каждого класса, а их может быть до 2.000, и для каждого объекта обучающей выборки, а их может быть до, я сказал, несколько миллионов может быть. Ну, то есть это огромное количество вычислений. И этот процесс может идти длительно из-за этой причины.

Кроме того, создаются… ведь не только ж распознавание проводится. Само по себе оно никому не нужно. Нужны выходные формы, которые получаются в результате распознавания. А эти формы надо рассчитывать. Вот, и на это тоже уходит время.

Поэтому получается такая ситуация иногда, что этот процесс затягивается. То есть если мы зададим синтез и верификацию модели, то это может, скажем, занять там несколько дней или неделю там. То есть часы, по крайней мере. Ну, то есть это может затянуться.

3.7. Бутстреппинг и оптимизация верификации

Поэтому, значит, есть вариант такой: создать только модели, потом проверить достоверность модели на небольшом числе объектов обучающей выборки, скажем, там, с первого по сотый. Вот. И определить… система сама покажет, сколько она времени потратила на это. После этого мы можем построить пропорцию простую, в которой написать, что вот, когда мы проверяли достоверность модели на 100 признаках, это заняло там 3 минуты, а сколько нужно… то есть не признаках, а объектах выборки. А сколько можно использовать объектов выборки для проверки достоверности модели, если у нас 8 часов? Вот я вечером сейчас поставлю на верификацию и пойду себе. И я знаю, что к утру она посчитает, какая достоверность модели. И по этой пропорции мы можем посчитать, что оказывается, нам достаточно для этого указать 150.000 объектов обучающей выборки. Тогда за 8 часов она посчитает. При том количестве классов и том количестве признаков, которые у нас в модели. Понимаете?

Поэтому здесь разделено эти операции синтеза-верификации разделяются, и можно выбирать подмножество объектов обучающей выборки для верификации. Ну, когда модель не очень большая, то тогда, конечно, никакого смысла это не имеет.

Теперь смотрим здесь вот. Это называется вообще вот, использовать часть выборки для синтеза модели, а часть выборки для проверки этих моделей на достоверность. Такой подход называется бутстрепный, бутстрепный подход. Это один из скандинавских математиков предложил такой вариант, когда у нас нет возможности проверить на внешних данных модели, то часть выборки используют для синтеза модели, а часть – для их проверки на достоверность. И тогда можно вот эти объекты, которые мы использовали для проверки на достоверность, можно не использовать для синтеза модели. Вот. То есть вот здесь вот мы всё это я реализовал.

3.8. Использование GPU и ограничения системы

Значит, кроме того, вы вот здесь вот можете обратить внимание, что мы можем для расчётов использовать либо центральный процессор, либо графический процессор, видеокарты. Вот. И если мы используем графический процессор, ГПУ, то расчёты могут идти примерно там несколько тысяч раз быстрее.

Значит, я фиксировал реальные задачи. Ну, само ускорение зависит от того, насколько у нас хорошая видеокарта, и от того, какие характеристики модели, сколько там классов, признаков, объектов выборки, потому что всё это влияет. Ну, у меня был такой случай, ну, обычно там несколько сотен раз ускоряется. Ну, так, допустим, считаю час, а так там секунду, к примеру. Вот, на ГПУ. То есть это значительно ускоряет работу, расчёты вот эти. Значит, но я видел, когда до 4.000 раз ускорение было. То есть была одна из моделей у меня, когда, ну, примерно 4.000 раз.

Вот. Ну, что такое 4.000 раз? Это значит, у нас…

Ограничения довольно слабые. Ну, в системе ИДС, вот я проводил исследования специально на предельные её возможности по обработке данных, то есть смотрел, какие массивы данных она может обрабатывать, не вылетать. Есть ограничения в самом языке программирования некоторые. И вот, значит, я проверял данные, значит, работу системы на 8 миллионах примеров. Здесь сейчас 20 примеров, а я прогонял систему, режимы системы, когда было 8 миллионов примеров. Понятно, да? То есть возможности довольно большие.

Если мы посмотрим, насколько кодов классов, здесь два. А я проверял, когда их там 200-300 этих классов. То есть вот здесь идёт вниз, вниз, вниз идёт. И когда признаков, допустим, там 700, там, или 1.000, или 2.000 признаков. То есть это вот окошко полностью заполнено было.

Ограничение было такое. Слушайте внимательно. В системе. Значит, что это три базы данных, связанных отношениями один ко многим: база данных заголовков, база данных кодов классов и база данных таблицы, база данных кодов признаков. Каждая база данных не больше 2 ГБ может быть в языке программирования. Но так было в той версии, которая была до вот этой версии профессиональной. Видите, здесь написано система X Pro. ADS Off. Видите, да, вверху в окне написано?

Вот. А если будет ADS On, то тогда ограничения на размеры базы данных снимаются. Я сделал учебную программу для освоения этого сервера. Я его использую как локальный сервер. Его можно использовать на хостинге где-то, на сервере в сети. Вот. Но я использую как локальный. То есть там же, где система, находится этот сервер. Мы его не видим. Это просто библиотеки, к которым обращается система. Поэтому там она побольше немножко стала по размеру инсталляция, потому что там добавились библиотеки.

Так вот, ребята, я в учебной программке пробовал сделать файлы базы данных именно. И у меня получалось делать там и 40 ГБ, и 400 ГБ. Ну, больше я не стал пробовать большего размера. Там, в самом этом Advantage Database сервере, в документации написано, что, я даже сейчас не помню, по-моему, там миллион терабайт размер базы данных, что-то типа такого. Что-то около миллиона терабайт. Ну, короче, у меня просто нет таких носителей, на которых такие базы данных размещать. И и не предвидится, я вам скажу, в ближайшие там 100 лет. Ну, это я так подшучиваю, ну, может, не 100 лет, там, не знаю, может, сделают там лет через 20-30. Но пока что, ну, у меня идёт речь о терабайтах, там, ну, 6 ТБ, 7 ТБ вот на компьютере. Ну не миллион же ТБ, правильно?

Вот. То, что ограничения есть, но оно связано только с физическими ограничениями компьютера.

Так вот, эта версия, которая сейчас у вас на экране, система ИДС, если там будет ADS On, то это означает, что там уже нет ограничения на количество объектов обучающей выборки, количество классов, количество признаков. Вернее, количество классов там остаётся всё равно, 2.000 ограничений. Количество признаков не будет ограничения.

То есть получается, что эти ограничения, так сказать, есть, конечно, на самом деле, но они связаны уже не с языком программирования, а с самим компьютером. С возможностями компьютера, с тем, какое на нём стоит аппаратное обеспечение, внешняя память какая по размеру.

Вот здесь вот рассчитано максимальное возможное по размеру база данных результатов распознавания – 2 ГБ. И говорится, что сейчас вот эта модель, которую мы сейчас будем рассчитывать, она составляет 60% от того, что возможно максимальное. Но это возможно максимальное именно когда сервер отключен. А когда он включен, тогда вот этого ограничения вообще нет. Но в системе я вот думал, ну как вот, если нам нужно больше обработать и превышает 2 ГБ база результатов распознавания, что с ней делать? Ну, ясно, что нужно удалить из неё какие-то результаты распознавания. Но какие именно? И вот я решил, что надо удалить те, которые наименее достоверные. А в системе есть, она не только распознаёт, идентифицирует, прогнозирует, но она и оценивает достоверность этих всех своих результатов идентификации, прогнозирования. Вот. И есть внутренние критерии достоверности. Мы это ещё позже рассмотрим более подробно. А сейчас я просто хочу сказать, что такой критерий есть, и он позволяет определить, какие результаты, скорее всего, ошибочными являются, как показывает опыт, и позволяет их удалить. В результате база результатов распознавания сокращается по размеру. И получается, что мы можем обрабатывать большего размера модели, чем если этого не делать. И вот здесь вот сейчас стоит параметр по умолчанию: удалить 10… то есть оставить 10% оставить, только оставить 10% наиболее достоверных результатов распознавания, остальное убрать. Вот. Можно поставить здесь 100, можно поставить 1.

3.9. Результаты текущего расчета

Вот сейчас пошло распознавание на центральном процессоре. Значит, сначала был осуществлён синтез моделей, потом проверка этих моделей на достоверность. Этот процесс закончился, ребята, у меня на моём компьютере за 6 секунд.

4. Устранение неполадок при установке (продолжение)

4.1. Помощь студентке Арине

Да, Арина.

Евгений Вениаминович, вы мне не можете минут пять уделить? Я всё ещё не могу установить программу ИДС.

Да давай, конечно.

Я могу включить демонстрацию экрана, просто у меня выскакивает это окно, интересное, непонятное.

Давай, давай, конечно, давай.

Ага, это ошибка. Так, значит, э-э, ошибка, причём языка программирования, ошибка исполнения. Так, значит, и у тебя старенькая версия системы. Значит, знаешь, что сделай? Значит, возьми сейчас сотри полностью папочку свою с системой. И скачай систему целиком с моего сайта.

А я скачивала только что, я удаляла, но я скачивала из облака, точнее с Яндекс диска.

Ну вот скачай сейчас по этой ссылочке с моего сайта: lc.kuba.ru.

Хорошо, спасибо, сейчас попробую.

Вот, вообще-то с диска тоже должно было бы работать. Я туда записывал все инсталляции. Но, может, как-то там или битый URL, не знаю. Ну, в общем, давай, скачивай и разворачивай.

Спасибо, хорошо.

И сейчас, э, и скажешь потом, получилось ли, заработала или нет.

Арина, когда у тебя установится, ты мне как-то сообщи, хорошо? И мы тогда быстро пройдём все эти этапы на твоём компьютере.

Хорошо.

Ну я надеюсь, она заработает у тебя.

Всё, готово.

Запустилось?

Ну давай сейчас, знаешь, что сделаем? Давай сейчас… Да, ты открываешь свой экран. Значит, если ты запускаешь StartAIDS с датой, то, значит, он удаляется и получается нормальный StartAIDS, который потом без даты будет. Вот запусти его.

Так, не удаётся выполнить, поскольку система не обнаружила… Ты подумай, а? Так. А ты можешь мне этот скриншотик прислать вот сейчас? То есть возьми его, увеличь. Ну вырежи его там ножницами, я не знаю. Да, вот так, ага. Вот. И пришли вот его в чат.

И сейчас мы, наверное, знаешь, чем займёмся? Я сейчас тебе пришлю эту… эту библиотеку. Ты попробуешь, будет ли она работать? Так, ACNET 10 DLL. А зачем такая библиотека? Так, ACNET 10 DLL…

Вот он всё, опять ошибку выдал. Не удаётся продолжить выполнение кода, поскольку система не обнаружила файл.

Библиотеку, да? Вот я сейчас эту библиотеку тебе пришлю в чат, а ты её себе туда запишешь, где находится у тебя система. И, по идее, должно заработать. Я сейчас не могу понять, почему у тебя её там нет. Ну ладно. Я вообще-то, когда инсталляции делал, готовил для размещения, то, естественно, я все вот эти библиотеки ACNET 10 DLL помещал туда. ACNET 10 DLL… Вот. Ну есть в инсталляции, что интересно. Каким образом у тебя его нет, я не очень понимаю. Так, как это могло получиться, что у тебя его нет? Так, значит, тогда я сейчас в чат его пришлю, этот… эту библиотеку тебе, а ты её запишешь в папочку системы, хорошо?

И когда будешь… И когда будешь записывать, проконтролируй, э-э, появится ли сообщение, что такой файл уже есть, или нет? Или она запишется, никакого сообщения не будет?

Хорошо.

Вообще, по идее, там такой файл должен быть. У меня есть он в папке системы.

Что-то не вижу, чтоб ты записывала его.

Так, вы же сказали, что сейчас пришлёте.

Так я прислал в чат.

Нет, ничего в чате нету.

А, вот пришло теперь.

Скачай просто и попробуй записать его туда. И обрати внимание, есть там такой файл или нет? В папке, где система находится.

А что это за папочка?

Ну, EDSX. Где ты развернула систему? Где ты запускаешь систему? Там, где вот этот файл StartEDS, который ошибку выдал.

Нет, не сказала.

Арина, ты что, ты архивируешь прямо в эту папку в три?

А, нет, она рядом, да? Вот теперь открой её, которая разархивировалась. Всё из неё скопируй и положи именно в дату три, дата Д, а не в три.

Вот, видите, есть такое заменить файл? Замени.

Да. Ну у тебя, я вижу, файл… папочка EDSX есть. Вообще-то там должно всё находиться внутри.

И всё, да? Я теперь опять StartEDS.

Ну смотри, смотри, Арин. Вот я вижу, у тебя есть папка EDSX. А, Guard EDSX. Что это такое? Вот открой её.

Вот эту?

Ниже, а, вот эту.

Вот, вот эта папка и есть рабочая папка, где система находится. Вот отсюда…

Тут есть StartEDS.

Есть StartEDS. Вот запусти её там. Ну сейчас обновится, развернётся и будет работать.

А вот там остальные… То есть почему у тебя находится в дате это всё? Там очень много файлов, которые можно постирать все. Огромное количество связанных с системой файлов. Вот только оставь папочку EDSX, где система находится, и всё.

4.2. Ошибка доступа к интернет-ресурсу

А это что такое?

Вот именно с этим я первоначально и пришла.

Так, стоп. У тебя интернет отключен, что ли?

Нет, я с компьютера в Тимсе.

Ну, в общем, она вот пытается запустить сейчас вот это вот URL, она пытается сайт открыть. Вот. Непонятно. Web 1179. Подожди.

Брик, нажми, Брик. Ну это что-то не… скорее не система, а скорее вот с настройками безопасности, может, там BrandMauer не разрешает. Возьми вот этот вот EDSX запусти прямо сам. Исполняемый модуль.

Вот он вот внизу там у тебя появилась эта… Так, о'кей, нажми.

Ну у тебя программа не может обратиться к интернету, короче. 1179. Причём проверено, что интернет есть. Ну а могу ли я к нему обратиться к этому? Сейчас я посмотрю. FTP… Может, этот сервер сейчас накрылся? Нет, всё работает.

У меня такое впечатление, что в этой операционной системе запрещено обращаться из программы к сайтам. Ну почему ошибка возникает? Оно должно было написать там… BrandMauer должен был написать, что там разрешить, запретить, там вот такое вот. Не знаю, у меня, как ты понимаешь, работает это всё.

Может быть, давайте я попробую с телефона интернет раздать, проверить? Хотя, не знаю, может, ничего не поменяется.

Я пробовала, Арин, ничего не меняется, то же самое.

А у вас тоже такое же, да?

Да, у меня на четырёх компьютерах то же самое, вот абсолютно то же самое. Пишет, что нет интернета, при том, при всём, что я с этого компьютера скачивала только что файл.

Да, это понимаю, да. Это понятно.

Так, так, так, так, так. Вот такие чудеса, да? Причём режим не защищённый. А если защищённый поставить?

На некоторые вещи я не могу, как бы, увидеть у себя на компьютере, поэтому я про них узнаю вот так вот, когда вы рассказываете.

4.3. Отправка исполняемого модуля

Значит, сейчас, э-э, я вам сейчас запишу в чат исполнимый модуль, а вы его запустите, хорошо? Вот, в чат. Я не знаю, получится… даст она записать его или нет. Попробуем. Вроде получилось. Вот возьмите сейчас этот модуль исполняемый, запишите и попробуйте. Он такой же, только там в защищённом режиме обращение к сайту идёт.

5. Завершение

Ну ладно. Наверное, тогда пока мы продолжим. Разрешить надо, в смысле, спрашивает. Сейчас я… Да, у вас доступ есть сейчас, да? То есть вы видите то, что я делаю. Да, есть. Угу. Вот. Теперь будет каждый раз обновление скачиваться. Потому что я один файлик зря записал туда. Так.

Ну что ж, тогда до встречи на лекции.

Угу, приглашение я вам пришлю.

Да, спасибо.

Спасибо за помощь.

Ну… Может, получится в перспективе.