***ФГБОУ ВО «Кубанский государственный аграрный университет имени И.Т. Трубилина»,***

***Российская Федерация***

**96 Инженирия знаний и интеллектуалные системы. Лабораторная 3. Достоверность модели 3.0.3 + разбор**

**Резюме:**

Видео представляет собой запись лабораторного занятия №3 по дисциплине "Инженерия знаний и интеллектуальные системы" для группы БИ-1701 от 24 сентября 2020 года, проводимого профессором Луценко Евгением Вениаминовичем.

1. **Введение и Контекст:**

Занятие посвящено лабораторной работе №3 (3.03) с использованием системы "Эйдос". Обсуждается продолжение изучения темы, начатой на предыдущем занятии, а именно – переход к оценке достоверности модели.

**2. Демонстрация Работы с Системой "Эйдос":**

* **Загрузка данных:** Показан процесс запуска системы, сброса предыдущих приложений и загрузки данных для лабораторной работы №3 из Excel-файла. Указываются диапазоны для классификационных (2-3) и описательных (4-12) шкал.
* **Обработка шкал:** Система автоматически определяет типы шкал (текстовые, числовые) и количество градаций. Обсуждается возможность корректировки числа градаций для числовых шкал.
* **Создание модели:** Исходные данные кодируются с использованием созданных справочников шкал, формируется обучающая выборка (нормализованная база исходных данных) и матрица модели (14 колонок, 50 строк).

**3. Анализ Данных и Модели:**

* **Исходные данные:** Продемонстрирован Excel-файл с примерами объектов (компьютерные комплектующие и др.), их свойствами (цвет, материал, размер) и классами (конкретными и обобщенными).
* **Справочники шкал:** Показаны сгенерированные системой справочники классификационных и описательных шкал. Подчеркивается важность кратких и понятных названий шкал и проблема дублирования наименований, решаемая нормализацией (созданием справочников). Уникальные значения текстовых шкал становятся градациями. Числовые шкалы обрабатываются путем деления диапазона (min-max) на заданное число интервалов (равных по величине или по числу наблюдений).
* **Обучающая выборка:** Представлена в двух формах – кодированной (числовой) и в виде, похожем на исходный Excel-файл, но с кодами классов и признаков.

**4. Синтез и Верификация Модели:**

* **Процесс:** Запускается синтез моделей (3 статистических: АБС, ПРЦ1, ПРЦ2 и 7 моделей знаний). Верификация (проверка достоверности) выполняется путем решения задачи идентификации объектов обучающей выборки.
* **Выбор режима:** Объясняется возможность раздельного выполнения синтеза и верификации, что актуально для больших объемов данных, так как верификация может быть очень длительной. Упоминается возможность расчета на GPU.

**5. Оценка Достоверности (F-мера Ван Рейсбергена):**

* **Классическая F-мера:** Используется как стандартный критерий достоверности. Рассматриваются типы решений при идентификации: истинно положительные (True Positive, TP), истинно отрицательные (True Negative, TN), ложно положительные (False Positive, FP), ложно отрицательные (False Negative, FN).
* **Недостатки F-меры:**
	+ Моноклассовость: Предполагает принадлежность объекта только к одному классу, что редко соответствует реальности.
	+ Чувствительность к объему выборки: Сумматоры TP, TN, FP, FN являются абсолютными частотами и зависят от размера выборки, что искажает оценку достоверности.
	+ Бинарность оценки ошибки: Суммирует единицы при ошибках, не учитывая степень уверенности модели в неверном решении.
* **Предложенные улучшения (Луценко Е.В.):**
	+ Мультиклассовое обобщение F-меры.
	+ Нечеткое обобщение: Вместо суммирования единиц при ошибках, суммируется степень уверенности модели (значение интегрального критерия, нормированное или нет), что делает оценку более справедливой.
	+ Инвариантность к объему выборки: Использование относительных частот (вероятностей) TP, TN, FP, FN вместо абсолютных.

**6. Интерпретация Результатов и Исследование Модели:**

* **Идентификация как разложение в ряд:** Процесс идентификации объекта интерпретируется как разложение его функции (описывающей объект) в ряд по функциям классов (обобщенным образам), хранящимся в модели. Коэффициенты разложения – это уровни сходства.
* **Анализ факторов (SWOT-анализ):** Система позволяет определить, какие значения факторов (признаков) способствуют или препятствуют переходу объекта в определенное состояние (класс). Это автоматизированный SWOT-анализ.
* **Анализ сходства классов:** Визуализация сходства между классами с помощью когнитивных диаграмм и дендрограмм кластерного анализа. Показано, как классы группируются по сходству их описаний.

**7. Заключение:**
Занятие завершается демонстрацией инструментов системы "Эйдос" для исследования предметной области на основе построенной модели, включая анализ факторов и кластеризацию классов.

**Детальная Расшифровка Текста:**

**1. Введение и Контекст**

Ребята, сегодня 24 сентября 2020 года.

Вторая пара, 9:45-11:15. Лабораторная работа номер три в группе БИ-1701 по дисциплине Инженерия знаний и интеллектуальные системы.

Занятие ведет профессор Луценко Евгений Вениаминович. Здравствуйте.

Вы должны сказать: "Здравствуйте, здравствуйте". Но никто ничего не говорит.
Здравствуйте.
Здравствуйте.

Значит, на прошлом занятии... Здравствуйте. Да, здравствуйте, ребята. Значит, на прошлом занятии лабораторном, которое было вчера, мы начали изучать лабораторную работу 3.03 и дошли до момента, когда будем изучать достоверность модели. Да?

**2. Демонстрация Работы с Системой "Эйдос"**

* **Загрузка данных**

Значит, давайте сейчас я быстренько напомню вам, как это было сделано. Запускаем систему. Сбрасываем... Вы сейчас должны видеть экран. Видно экран, ребята? Сбрасываем все приложения. Заходим в диспетчер приложений, режим 1.3. Добавляем третью лабораторную работу 3.03. О'кей.

Это программный интерфейс ввода данных из внешнего файла экселевского. Здесь диапазон классификационных шкал 2-3 и диапазон описательных шкал 4-12. Файл скачивается из папочки лабораторных работ. Создается пустое приложение. В него записывается этот файл исходные данные и распаковывается. Данные вносятся в систему Эйдос.

* **Обработка шкал**

Было найдено две текстовых классификационных шкалы, ни одной числовой. И в описательных шкалах была одна найдена числовая описательная шкала и восемь текстовых. Здесь вот можно поменять число градаций, поскольку у нас есть числовая шкала, то вопрос возникает о числе числовых, о количестве числовых диапазонов в этой числовой шкале или или шкалах. Ну сейчас мы ничего менять не будем. А если если хотим поменять, то вот здесь вот надо пересчитывать шкалы и градации.

* **Создание модели**

Вверху написано, что у нас будет создана матрица: 14 колонок на 50 строк, матрица модели. Выйти на создание модели. Созданы справочники классификационных и описательных шкал и градаций, и потом они были использованы для того, чтобы закодировать исходные данные, в результате чего получилась обучающая выборка, которая представляет собой нормализованную базу исходных данных.

У кого-то микрофоны включены, ребята. Выключайте микрофоны.

Вот. Значит, вот здесь для студентов сообщения, которые они никогда не смотрят, сразу о'кей нажимают. Ну, по крайней мере, когда были в аудиториях. Здесь сказано: "Посмотреть на файлы исходных данных по этому пути". Смотрим.

**3. Анализ Данных и Модели**

* **Исходные данные**

Вот создана папочка приложения 01. Их может быть много там. Вот файл исходных данных. Микрофоны надо выключать, ребята. Выключайте микрофоны.

Вот. Смотрим на файл исходных данных. Что он собой представляет? Это примеры предметов, которые лежали на столе во время лабораторной работы, ну, связанные с компьютерами, конечно, и там ещё какие-то вещи студентов лежали. Есть две классификационных шкалы: конкретные классы и более общие классы. Это для двухслойной сети. И признаки предметов. То есть это их свойства и значения этих свойств: цвет, например, белый там и так далее. И шкалы есть текстовые и числовая вот одна шкала есть для примера.

* **Справочники шкал**

Смотрим теперь на сами эти шкалы, которые были сформированы. Режим 2.1. Классификационные шкалы. Мы видим, что названия этих классификационных шкал такие же, как названия в экселевском файле. Вот они. Отсюда вытекает такая рекомендация из этого. Эти названия писать как можно короче, но так, конечно, чтобы было понятно, что они означают.

У вас была дисциплина Базы данных, да, ребят? Отвечайте, у кого была дисциплина Базы данных? КГУ, КубГАУ. Что-то вы никто не отвечаете или что? Была дисциплина. Значит, вот видите, здесь есть повторение наименований. Если есть повторение наименований, то это признак того, что надо делать справочник соответствующий. Для чего? Для того, чтобы это наименование хранилось в базах данных в одном месте, а в остальных местах ссылки были на это место, где оно хранится, текстовое наименование. И произвести для этого нормализацию баз данных, так сказать, на основе этого.

Так вот, уникальные текстовые значения являются градациями классификационной шкалы. Где-то там повторяются много раз, а в градациях они один раз повторяются. И рассортированы по алфавиту.

Описательные шкалы. Да, ребят, в аграрном университете, вы меня видите или нет? Или только слышите? Ответьте, вы меня видите или нет? Какие молчаливые. Может, их там нет никого? Всё время такое возникает вопрос. Есть они там или нет? Лена Леськова, есть? Меня видно или не видно? Вот вас не видно. Не видно, да. Вот потому что камера захвачена другим приложением. Ну неважно. А рабочий стол видно, да? И экран компьютера видно? Да? Экран видно? Ладно. Понятно, что видно.

Так вот, мы видим, что эти шкалы текстовые... Текстовые шкалы здесь номинального вида типа. Хотя цвет можно было бы сделать по спектру, тогда была бы порядковая шкала. Материал - это номинальная шкала, здесь никакого порядка нет. А вот размер один можно было бы сделать порядковой шкалой, но здесь она номинальная. Во-первых, мы что видим? Большая, большой. Это в экселевском файле надо привести к одному, скажем, роду, мужскому, например. Средний, маленький, большой. И другой раз ещё большая. Вот. То есть надо всё, чтобы отвечало на один вопрос. То есть у нас получится тогда четыре градации в этой шкале размер один. И надо их расположить в таком порядке, чтобы минимальный код был минимальный по смыслу величины, а максимальный код у максимальной. То есть, допустим, было бы так написано: 1 наклонная черта 4 - маленький, 2 наклонная черта 4 - средний, 3 наклонная черта 4 - под руку, и 4 наклонная черта 4 - большой. Тогда бы, если бы так в Экселе было сделано, а в Экселе для этого достаточно выделить колоночку вот так и заменять. Вот, допустим, под руку заменяем там на это 3 наклонная черта 4 под руку, например. И она во всей колоночке заменит система Excel. Понятно, да? То есть не руками там вбивать, а именно вот заменить заменой. И тогда при вводе данных в систему, здесь уже будет рассортировано в смысловом порядке.

Размер два - это числовая шкала. Значит, было для числовых шкал алгоритм другой, чем для текстовых. Для текстовых там просто выбирают уникальные значения и сортируются по алфавиту. А для числовых шкал ищется минимальное и максимальное значение и делится на то число диапазонов, которое вы задали в диалоге, когда вводили данные в систему. И можно задавать разное число этих диапазонов и разный их тип. Можно задавать равные диапазоны с разным числом наблюдений, а можно задавать разные по величине диапазоны с одинаковым числом наблюдений.

А потом идут такие вот шкалы бинарные, там булево да-нет. Наличие экрана, наличие там, форма разная, конечно, наличие ножек. Значит, обратите внимание. Значит, здесь э в качестве информации об объекте наблюдения является не только то, что у него есть какие-то признаки, но и то, что их нет. То есть есть и да, и нет, видите? И это всё несёт информацию. То, что есть какой-то признак, и то, что его нет, тоже несёт информацию.

* **Обучающая выборка**

В результате была закодирована исходные данные вот эти экселевские. Это первая колонка, которая содержит наименование объекта выборки. Здесь коды классов, согласно вот этим шкалам классификационным. А здесь коды признаков, согласно описательным шкалам. Есть другая форма этой обучающей выборки, очень похожая на экселевскую форму. Вот. Вот смотрите, прямо такая же точно форма. Вот, видите? Номер порядковый, наименование объекта выборки, шкалы классификационные и описательные шкалы. Именно все те же самые, в том же порядке. И название написано в колоночке, в верхней шапочке, да ещё и с кодом, какой код этой этого класса, этой классификационной шкалы, какая колоночка. Теперь смотрите, вот здесь вот у нас внизу есть ещё одна шапка, в которой номер порядковый - это нулевая колонка, потому что это колонка дополнительная. В Экселе тоже она не нумеруется ни А, ни Б, просто вот номер идёт. И потом наименование объекта - первое, классификационные шкалы - вторая, третья, описательные - 4-12. И так оно и есть. Когда мы вводили информацию в то в экранной форме, мы как раз такие диапазоны и вводили. То есть они точно соответствуют тому, что мы вводили.

**4. Синтез и Верификация Модели**

* **Процесс**

Вот. Теперь дальше мы осуществляем синтез модели. У нас есть таким образом всё необходимое для синтеза модели и для верификации. Верификация - это проверка модели на достоверность. Открываем экранную форму соответствующую. Видим, что создаётся, будет создано сейчас три статистических модели: АБС, ПРЦ1 и ПРЦ2, и семь моделей знаний. Можно считать на графическом процессоре, можно на центральном. Сейчас будем на центральном считать.

* **Выбор режима**

И здесь вот есть разные варианты синтеза и верификации: только синтез, только верификация. Для чего это сделано? Для того, чтобы, если очень большое число примеров обучающей выборки, ну, скажем, 700.000, то тогда верификация может занять неделю, к примеру. Если там 500 классов, там 11.000 признаков и 700.000 примеров обучающей выборки, это может занять значительное время. Поэтому я это разделил функции: только синтез модели, только верификация. Синтез модели всегда осуществляется довольно быстро, даже когда очень большие выборки, большое число классов и признаков. Всё равно это происходит, ну там 15-20 минут, ну час там, допустим. Ну то есть какое-то такое время, но более-менее приемлемое. А вот верификация - это каждый из этих 700.000 объектов сравнивается с каждым из 500 классов, а для сравнения нужно 11.000 признаков проверить его, какое количество информации в них содержится о принадлежности или непринадлежности к этому классу. Это огромный объём вычислений. Поэтому, когда не было графического процессора, это на центральном процессоре было нереально посчитать. Поэтому я делал так. Значит, э считал только верификацию для небольшого числа объектов, ну, допустим, 100 там или 1.000 вот выберу. Она, скажем, считает там 10 минут. После этого я составлял пропорцию, что 10 минут там 1.000 объектов. Вот. А сколько там минут там, ну, допустим, там 8 часов, ну, в минутах тоже, э вот, э X объектов. И вычислял, какое число объектов вот здесь вот можно написать, чтобы 8 часов шла верификация модели.

Евгений Вениаминович, прошу прощения. Алло. Да-да, я слышу. Вы спросите у студентов, экран ваш виден или нет? Я вас слышу, а экрана не вижу. Я спрашивал, они говорят, что виден. Виден, да? Да, и у меня идёт сообщение, здесь есть, что э закрыть доступ, скрыть. То есть э предоставлен доступ к вашему... Всё нормально, они видят, да? У меня почему-то я подключился и экрана нет, а только аудио идёт. Сейчас спросим. Ребят, вы меня мой экран видите, нет? Вас видно всё. Всё видно, да? Вот сейчас экранную форму... Да, всё видно. Ну хорошо. Это в аграрном? В аграрном вы отвечаете, да, что видно? Я в аграрном подключился, слышу... Я спрашиваю из аграрного, ребята, видно? Экрана не видно. Просмотр экрана у меня написано, экрана нет. Да вот сейчас попробуем переподключиться. Ну ладно, мы попробуем сейчас с Колей переподключиться. Я понял, привет, Коль. Извините пока. Да. Вот ребята из аграрного, тут тут под... Ну вот Марина, Марина Подгорная, экран виден или нет? Ответь, пожалуйста. Экран моего компьютера виден, Марина? Марина! Или Даниил там, или кто-нибудь, или Олег, ответьте. У вас плохой динамик, да, ваш экран видно, не переживайте, мы всё видим. И слышите, да? Да, и всё прекрасно слышим. Ну хорошо, хорошо.

Вот. Вот для этой цели сделана, значит, такая возможность использовать какое-то подмножество объектов для верификации модели, когда очень много. Но на центральном, на графическом процессоре получается всё, даже с очень большими выборками, в реальное разумное время. Э, ну сейчас я в этом на расчётах на графическом процессоре в одной модели есть неточность, не влияющая на её достоверность, результаты её использования, но когда показываешь модели, то как-то не хочется, чтобы эта неточность была. Сейчас нажимаю о'кей, ребята. Происходит синтез моделей и проверка их на достоверность. Проверка на достоверность происходит путём э решения задачи идентификации объектов обучающей выборки. То есть обучающая выборка копируется в распознаваемую. Шаг первый, видите? Потом осуществляется синтез модели абсолютных частот, потом синтез модели относительных частот, потом синтез модели знаний осуществляется, а потом верификация. Вот. В разных э в разных моделях и с разными интегральными критериями. И потом формирование выходных форм по результатам верификации.

Значит, на прошлом занятии вчерашнем, которое, похоже, что сегодня повторно проводится, потому что там у нас расписание постоянно то добавляют, то удаляют пары, я не пойму уже, какие пары там э я провёл, какие не провёл. Уже один раз как-то провёл два раза, и похоже, что сейчас я тоже два раза провожу. Но я двигаюсь немножко дальше, чтобы вы, как бы, напоминаю, на чём ход мысли, как говорится, что мы изучали на прошлом занятии, и потом продолжаю.

**5. Оценка Достоверности (F-мера Ван Рейсбергена)**

* **Классическая F-мера**

Вот сейчас мы начинаем изучать новое, то, что на прошлом занятии только мы запланировали изучить. Это анализ достоверности модели с разными интегральными критериями. Значит, у нас есть такая форма, где слева у нас модели перечислены, а потом колоночка идёт интегральные критерии. В системе Эйдос два интегральных критерия используются. Один э, ну это я сейчас немножко пропустил, не показал сами модели. Вот модель АБС, модель ПРЦ2, это уже в процентах модель АБС. А это вот модель, где уже сравнения проводятся этих относительных частот друг с другом и с э условных относительных частот, то есть в колонках по классам, сравнивается частота относительной частоты с безусловной относительной частотой по всей выборке. А вот здесь вот помощь у нас. Здесь вот у нас описано, как это рассчитываются эти модели. P(i|j) - это вероятность встречи i-того признака в j-той группе, а P(i) - это вероятность наличия этого признака по всей выборке. Кстати, ребят, я в самом начале вам упустил сказать, надо было это сказать, что занятие не только я провожу, а ещё профессор Аршинов Георгий Александрович, с которым мы разговаривали.

И э сейчас мы посмотрим на интегральные критерии. В системе два интегральных критерия. Один называется э сумма знаний. На самом деле у нас вот в матрице информативности у нас написано, какое количество информации мы получаем о том, что объект с этим признаком относится к житому классу. Это мы посчитали уже. Ну я показал это быстро, может быть, слишком быстро. Сейчас ещё раз покажу. Вот. Инф1. Какое количество информации в битах мы получаем о принадлежности объекта с каким-то признаком к какому-то классу. И это количество информации может быть положительным и отрицательным. Если признак встречается в группе, в соответствующем классе реже, чем в среднем, тогда это наличие этого признака говорит о том, что это не к этому классу относится объект. А если чаще, чем в среднем, тогда информация о том, что к этому классу он относится. И вот о каждом признаке мы знаем, какое количество информации в нём содержится. А вопрос возникает такой. Это вот хи-квадрат критерий. Возникает вопрос о том, а если у нас об объекте известно, что у него не один признак, а 10, например, там 15, какое-то количество признаков, то как тогда мы можем оценить, к какому классу, по-видимому, он принадлежит? Предлагается использовать для этого суммарное количество информации во всех признаках объекта о том, что он принадлежит тому или иному классу. Это и есть интегральный критерий сумма знаний.

Теперь я хотел бы ещё вот в двух словах сказать, что когда мы сравниваем два числа, то есть два способа это сделать. Можно сделать способом вычитания, а можно сделать способом деления. Если мы, допустим, сравниваем фактическую частоту наблюдения этого признака в житой категории, в житом классе, с теоретической частотой путём вычитания, то это хи-квадрат, критерий Пирсона. А если мы это же самое делаем путём деления, тогда вот видите, здесь n(i|j), а это вот n \* n(i) \* n(j) - это перевёрнутая теоретическая частота. Видите? Тогда получается количество информации по Хракевичу. Я просто хочу вам сказать, что я когда-то это заметил, но это, по-моему, очень интересно, что критерий классический хи-квадрат оказывается теснейшим образом математически связан с количеством информации по Хракевичу, с мерой количества информации.

И вот мы смотрим на интегральный критерий первый. L(i) - это массив, описывающий объект идентифицируемый. У него, если признак есть, то единичка в этом массиве, а если нет, тогда нолик. А таким образом получается, что если мы вот такую сумму вычисляем, то мы практически что делаем? Мы суммируем количество информации во всех признаках объекта о том, что он принадлежит к житому классу. При этом, если признака нет, то умножается на ноль, ну, считая, что мы ничего не делаем. Ну можно было сделать либо обход суммирования, логику там if then else там это написать, а можно было вот такой массив ввести. Но мне больше понравился массив. Почему? Потому что вот это выражение - это широко известное выражение, которое встречается в очень многих областях науки. Ну, допустим, в теории позиционных систем счисления такое же, в нейронных сетях там, ну, в общем, масса есть областей, где встречается вот это выражение, очень похожее. Но чем оно интересно? Оно представляет собой скалярное произведение двух векторов: вектора класса в системно-когнитивной модели и вектора объекта, описывающего объект. И это скалярное произведение, то есть косинус угла между векторами. И чем оно интересно? Это неметрическая мера сходства, то есть она не связана с предположением о том, что пространство ортонормировано. Это предположение обычно о нём никто не знает, что такое, как бы, даже, вернее, скажем так, никто не знает, что оно может быть не ортонормировано. Предполагаю, что все оси взаимно перпендикулярны пространства модели, и можно использовать расстояние Евклида, ну, вариант теоремы Пифагора для определения расстояния. Но на самом деле, если мы посчитаем взаимосвязи между классами, между признаками, то оказывается, что они взаимосвязаны. И реально никогда пространство не бывает ортонормированным. Поэтому применение такой меры расстояния, как евклидово расстояние, она является некорректным. А вот это скалярное произведение не связано с этим предположением, называется информационное расстояние, я его называл или межвекторное расстояние - это мера корректная. Кроме того, есть определение шума. Белый шум определяется как такой сигнал, у которого каждый элемент этого сигнала имеет корреляцию с любым другим элементом этого сигнала, равную нулю. Ну, вернее так, там говорится не так, что она равна нулю прямо, а вот она стремится к нулю при увеличении размеров этих отрезков этих сигналов, фрагментов этих сигналов. Вот. То есть при увеличении этих фрагментов, э корреляция стремится между ними к нулю. Что это значит? Это значит, что если у нас в исходных данных есть шум, и мы используем именно вот такой белый шум, который так вот определяется, то и мы используем эту меру расстояния, то мы практически, фактически мы подавляем этот шум, не учитываем этот шум при принятии решений.

* **Недостатки F-меры и предложенные улучшения**

И второй есть критерий интегральный в системе, который очень похож на первый, но единственное, что там вектора, вот этот вектор класса и вектор объекта, они стандартизированы. Видите, вычитается среднее из них и делится на среднеквадратичное отклонение. Можно было бы, конечно, и по-другому стандартизировать. Допустим, можно было из значения вычесть минимальное значение и разделить на максимальное минус минимальное. Тоже была бы стандартизация. Ну, этих вариантов стандартизации их много, но я вот выбрал такой классический, как я бы сказал. Если это сделать, тогда получается, что скалярное произведение получается нормированных векторов. Вы тоже эту формулу знаем, Пирсон её вывел. Вообще он вывел почти всё, если честно. Вот всю статистику, ну я не знаю, не считал сколько именно, но, наверное, процентов 90 того, что есть в классической статистике, в параметрических методах статистики - это вот работы Пирсона. И там ещё Стьюдент, там Фишер вы слышали. Ну а вот Фишер, то есть Пирсон - это такой гигант просто. Вот он, ну, практически он внёс огромный вклад в разработку параметрической статистики. И вот, в частности, это выражение, оно тоже является интегральным критерием в системе Эйдос.

Поэтому мы оцениваем достоверность моделей по двум интегральным критериям: сумма знаний и семантический резонанс. Ну я их так назвал. Вот. Как мы это делаем? Значит, смотрим Help этого режима. И здесь я вам скажу, что для оценки достоверности моделей используется F-мера Ван Рейсбергена в системе Эйдос. И, значит, несколько она, и сама она используется, и несколько модифицированная мера Ван Рейсбергена используется. Почему? Потому что мера F-мера Ван Рейсбергена является классической мерой достоверности модели. Я сам пошлю ссылочку в чат на статью свою по этому вопросу. Вот. И вы видите, как я это делаю. То есть я нахожу статью по ключевым словам в списке статей и в чат посылаю. И в другой чат тоже посылаю. Во все чаты. Вот. Видно, да, ребята? Видно.

Так вот, что это за F-мера Ван Рейсбергена? Значит, что должна делать система, ребят? Простейшая задача, которую должна решать система на основе модели - это задача идентификации. То есть она должна правильно относить объекты обучающей выборки, которая скопирована в распознаваемую, тестовую выборку. Правильно должна относить их к тем категориям, я их называю классы, к которым эти объекты относятся, и правильно не относить к тем категориям, к которым они не относятся. Значит, когда объект относится к какому-то классу, то уровень сходства выше нуля с этим классом у этого вектора, описывающего объект, уровень сходства с вектором, описывающим класс, больше нуля. А если меньше нуля, значит, система считает, что он не относится к этому классу, в соответствии с моделью. Значит, когда относится, значит, это называется позитивное решение, не относится - негативное. Зависит от того, какой критерий интегральный, больше нуля или меньше нуля. И эти решения могут быть истинными. Тогда их называют True Positive, True Negative. Вот. И могут быть ложными: False Positive, False Negative. Истина, ложь. Истинное решение: объект относится к классу, и система его и к этому классу и относит. Объект не относится к какому-то классу, и система к этому классу его и не относит. Вот. Если такие решения все, а ложных нет, тогда критерий Ван Рейсбергена равен единице. Вот. Значит, если возникают ошибки, то есть объект не относится к классу, а система его относит, это называется False Positive, то есть ложноположительное решение. Тогда Рейсберген суммирует единичку к этому сумматору. Да, кстати, вот что он сделал? Он ввёл четыре сумматора: True Positive, True Negative, False Positive, False Negative. Вот. И просто суммировал, каких сколько решений было на основе модели о принадлежности, непринадлежности. Ну, естественно, для того, чтобы определить истинное или ложное решение, для этого нужно знать, к какой категории относятся и не относятся объекты. Поэтому распознаётся обучающая выборка. То есть просто обучающая выборка копируется в распознаваемую или тренировочная выборка копируется в тестовую выборку, это синонимы. Распознаваемая - это тренировочная, на основе неё создаётся модель. А распознаваемая - это тестовая, проверяется модель. Ну мне больше нравится распознаваемая, потому что мы ж не можем не только тестировать модель, а просто можем ей пользоваться для решения задач практических уже, которые перед нами стоят. Это ж не тестирование, это просто уже применение. То есть распознавание, идентификация, прогнозирование.

Вот. И вот получается что? Что если объект, это про ложное решение, система относит его к какому-то классу, а фактически он к нему не относится, тогда это ложноположительное решение, ложное срабатывание. А если объект на самом деле не относится к какому-то классу, а система его относит к этому классу. Представляете? Вот. Нет, это про ложное отрицательное решение хочу сказать. Если объект на самом деле относится к какому-то классу, а система его к этому классу не относит, это называется ложное отрицательное решение. И вот можно, используя эти сумматоры True Positive, True Negative, False Positive, False Negative, посчитать меру Ван Рейсбергена. Вот здесь вот у нас написано. У неё, здесь написано очень коротко, в интернете можно найти гораздо подробнее. Там есть очень много разных интересных моментов.

Значит, я эту меру Ван Рейсбергена использовал, но я могу вам сказать, я обнаружил в ней недостатки некоторые. Какие в ней недостатки я вижу? Первое - эта мера моноклассовая, то есть считается, что объект может относиться только к одному классу. Но не может относиться к нескольким классам. А реально все объекты относятся к нескольким классам. Такого вообще не бывает, чтобы он только к одному классу относился объект. Ну мы это можем искусственно сделать в модели, но вообще-то всегда ко многим классам. Ну, допустим, я к каким классам отношусь? Муж, дедушка, отец, вот, профессор, преподаватель вуза, мужчина, и так далее, и так далее. То есть я ко многим классам отношусь, понимаете, к категориям обобщающим. Вот. И также точно можно придумать много классов, к которым любой объект относится. Вот. Ну поэтому мера Ван Рейсбергена, она нуждается в мультиклассовом обобщении. Этот вопрос обсуждается, я предложил математическое решение этого вопроса. То есть предложил мультиклассовое обобщение меры Ван Рейсбергена.

Второе очень важное момент, связанный с мерой Ван Рейсбергена, состоит в том, что он всегда суммирует ко всем сумматорам единички. И когда у нас... Сейчас я вам покажу, как выглядит конкретная форма, где есть различные решения: положительные, отрицательные. Ну где-то не в самой лучшей модели. Это я просто показываю, что вот мышка один в модели... Ну это даже не... Текущая инф1, а распознавание было в седьмой. Ну давай сейчас возьмём самую лучшую модель, выберем. Это хи-квадрат, она обычно наиболее хорошая. И в ней проведём распознавание на центральном процессоре. А теперь посмотрим на результаты... То есть сейчас я действую согласно вот этому алгоритму преобразования данных в информацию, а информации в знания. То есть когда мы создали классификационные, описательные шкалы и градации, с помощью них закодировали исходные данные, создали модели, вот, статистические модели и модели знаний. Потом выберем сейчас наиболее достоверную из них, уже даже выбрали, и в ней провели распознавание. И начинаем решать вот эти задачи. То есть мы действуем согласно логике и схеме преобразования данных в информацию, её в знания.

**6. Интерпретация Результатов и Исследование Модели**

* **Идентификация как разложение в ряд**

И смотрим результат решения задачи идентификации самой обучающей выборки. Смотрите, ребята, вот у нас мышка один. Она похожа больше всего на мышку (конкретный класс) и на элемент компьютера. И так оно и есть, она действительно относится к этим классам, птичка стоит. Ещё она похожа на телефон, но в меньшей степени. А почему она похожа на телефон? А потому что она пластмассовая, и на ней есть кнопочки. Вот. А ещё на средство связи, потому что телефон - это средство связи, однозначно является. На клавиатуру похоже. То есть это ложноположительное решение. А теперь смотрим дальше. У нас, когда мы вот так вот двигаем, то мы видим, что всегда на первых позициях наивысший уровень сходства у тех классов, к которым какой-то объект, который мы здесь вот распознаём обучающей выборки, действительно к этим классам и относится.

* **Анализ факторов (SWOT-анализ)**

А теперь смотрите, вот сейчас я вам здесь найду... Вот, пинг-понговый мячик. Вот система подумала, что он похож на мышку. Видите, пинг-понговый мячик? И на средство связи похож. Но я скажу, что сходство с мышкой весьма отдалённое. Но мы, в общем, можем догадаться, почему модель отнесла этот пинг-понговый мячик к этому классу мышка. Потому что у них одинаковый материал - пластмасса. А теперь смотрите, она-то отнесла, это ложноположительное решение. Ещё к телефону, и к средству связи по этой же причине отнесла. Но смотрите, ребята, она-то отнесла всего на с уровнем сходства 5%. Если мы 100% не проценты не будем использовать, а разделим на 100, то это, считай, 5 сотых. Это не единичку надо суммировать к ложноположительным решениям. То, что Ван Рейсберген суммировал единичку к ложноположительным решениям, то есть это False Positive. А я говорю: "Да нет, единичку - это как-то очень уж прямо жестоко он наказывает модель". Надо суммировать 5 сотых, ребята, а не единичку. Это я назвал нечёткое обобщение меры Ван Рейсбергена. То есть я предложил мультиклассовое нечёткое обобщение меры Ван Рейсбергена. В этой статье, которую я сейчас вот вам привёл. Вот. И считаю, что это справедливо. Система сама сомневается, что это мышка. Она понимает, что сходство весьма такое отдалённое, всего 5% сходства, понимаете? Что ж мы её наказываем, добавляя единичку к сумматору ложноположительное решение? Мы должны адекватно, она сама оценивает достоверность своего решения. Надо вот эту достоверность и использовать. Вот я в этом уверен, это очень убедительное обоснование, такие разумные. Таким образом получается мультиклассовое нечёткое обобщение меры Ван Рейсбергена.

* **Анализ сходства классов**

Но и это не всё ещё. Ребята, вот эти вот сумматоры, которые предложил Рейсберген: True Positive, True Negative, False Positive, False Negative - это же абсолютная частота, абсолютные частоты. И они зависят от объёма выборки. Если у нас выборка была маленькая, то эти частоты в сумме будут давать маленькую величину. Сумма этих всех частот, она равна объёму распознаваемой выборки или обучающей. Все решения, то есть какое-либо из этих решений выпадает. Если все их просуммировать, то как раз получится число объектов обучающей или там распознаваемой выборки в данном случае. Ну и получается, что эти сумматоры, ребята, слушайте дальше внимательно, их значения линейно зависят от объёма выборки. Чем больше объём выборки, тем больше значения этих сумматоров. Возникает вопрос такой: если они зависят от объёма выборки, будет ли зависеть от объёма выборки сам критерий достоверности Ван Рейсбергена? Я вот в этой статье привёл графики, в которых я исследовал зависимость критерия Ван Рейсбергена от объёма модели. Худшие мои опасения оправдались, ребята. Оказалось, что критерий Ван Рейсбергена зависит от объёма обучающей выборки. Ну это просто, ну я не знаю, это плохо просто вообще, не годится так. Если мы берём большие какие-то задачи решаем, то у нас достоверность получается ниже по критерию. Если маленькие, то выше. Ну что за ерунда такая? Ведь модель-то одна и та же. Вот. И выборки эти являются подмножествами одной и той же выборки, какой-то генеральной совокупности там. Короче говоря, я предложил, что сделать? Я думаю, что правильно сделал, что предложил. Я предложил заменить вот эти относительные частоты, то есть абсолютные частоты, счётчики вот сами эти, относительными частотами, то есть посчитать вероятность того, что у нас будет истинно положительное, истинно отрицательное, ложно положительное, ложно отрицательное решение. Ну, точнее, не вероятность, а относительную частоту. Вот. Ну, при увеличении объёма выборки эта относительная частота стремится к вероятности. И потом уже эти вероятности использовать для расчёта меры Ван Рейсбергена. Я это сделал, соответствующие математические выражения вывел, посчитал на примерчике всё это начисленном. И у меня получилось, ребята, такая картина, здесь вот формулы приводятся где-то там пораньше немножко. Вот они для этого случая. Это нечёткое мультиклассовое обобщение меры Ван Рейсбергена. И получилось то, что и ожидалось. Оказалось, что вот эта вот инвариантная относительно объёма данных, нечёткая мультиклассовая мера Ван Рейсбергена, она очень быстро стабилизируется. Значит, я предложил, собственно, две меры. Одна мера мультиклассовая сразу же, то есть нечёткую мультиклассовую меру предложил. Вот, это L1 назвал её, а L2 - это ещё и инвариантная относительно объёма выборки. Она быстро стабилизируется. Эти меры по величине достоверности выше, чем классическая мера Ван Рейсбергена, F-мера. Почему? Потому что они более справедливые. Суммируются к ошибочным решениям, к сумматорам, учитывающим ошибочные решения, суммируются не единички, а степень ошибки. Вот эта величина ошибки и суммируется. Получается, что на самом деле достоверность модели гораздо выше, чем по мере Ван Рейсбергена, но значительно выше, я бы сказал.

Как это можно интерпретировать, если мы посмотрим на выходные формы по результатам идентификации? Смотрим 4.1 3.1. Мы видели, что чем выше уровень достоверности, тем чаще это соответствует фактической ситуации. Значит, давайте сейчас посмотрим, как это выглядит графически. Потом мне пришло в голову, что это можно посчитать, а нарисовать. Так, что-то такое слишком много систем, наверное, разных. Сейчас попробуем. Извините. Что-то такое произошло, какой-то конфликт между всеми этими многочисленными системами. Я замечал, что Teams хорошо притормаживает компьютер. А там ещё Webex. То есть сейчас два канала трансляции на моём компьютере в разные стороны идёт. Ну он что-то не хочет показывать. К сожалению. Короче говоря, я просто взял и посчитал, сколько истинно положительных, истинно отрицательных решений, ложно положительных и ложно отрицательных решений при различных интегральных критериях и различных уровнях сходства. Вот сейчас окошко, ребят, появилось, что не было папочки вот такой для частотных распределений числа решений. Вот это система, когда первый раз создаёт папочку, она всегда вот так пишет. Это надо внимательно читать путь на эту папочку. Вот. Что мы видим здесь? Значит, мы видим ось Y - это сколько таких решений было. И вправо - это положительные решения, влево - отрицательные решения. Здесь всё внизу поясняется, кстати. И когда эта форма создана, и всё тут написано. И вот, посмотрите, ребята, значит, сглаженные центральное линейное сглаживание. Вот. Что мы видим? Что решения отрицательные, когда у нас уровень сходства отрицательный, всегда истинные, ложных вообще нету в этой маленькой модельке учебной, которую мы сейчас исследуем. А положительные решения есть ложные, вот эта вот синяя линия. Но они все при очень низких уровнях сходства. При уровне сходства выше 40%, ложных решений вообще нет. Это надо запомнить. Ниже 30% нет истинных положительных решений. То есть ниже 30% все решения ложные положительные. В диапазоне от 30 до 40% уровня сходства есть и истинные, и ложные решения, причём при увеличении уровня сходства доля истинных решений увеличивается. Выше 40% вообще не наблюдаются ложных решений. О чём это говорит? Это даёт нам критерий, ребят, для интерпретации вот этой вот формы выходной по результатам решения задачи идентификации. Ну вот опять надо задать модель, для которой мы смотрели эту кривульку. Мы же пересчитали сейчас модели, вы же заметили, наверное. Пересчитали 4.1. Почему-то там что-то произошло. Само распознавание провели и смотрим результаты в этой модели 4.1 3.1.

Так как распознавание - это очень длительный процесс, то я сделал его для текущей модели. А очень многие формы для любой модели выводятся и считаются сразу для всех. Значит, ребята, вот мы видим здесь ложные решения, видите, ложноположительные решения. 2,8 уровень, 22, 9. Да, они есть. Есть ложные решения. Но ни разу, ни разу нет случая в этом, в этой задаче, когда ложное решение, ложноположительное решение имело уровень сходства выше 40%. Такого у нас вообще ни разу не было. Только ниже 40%. А если от 30 до 40, вот, допустим, 37, видите? То там есть и истинные, и ложные решения от 30 до 40 уровня сходства. Но выше 40 только истинные. Это даёт нам способ интерпретации вот этих форм. То есть мы можем спокойно не обращать никакого внимания на результаты идентификации с низкими уровнями сходства объекта с классом. То есть мы знаем хорошо, что с низким уровнем сходства до 30% вообще нет истинных решений, все решения ложные. Мы это видели уже. Вот этим мы можем руководствоваться при интерпретации вот этих результатов решения задачи идентификации.

Есть ещё другой, много других вариантов вывода этой информации, результатов идентификации. В той форме у нас слева были классы, то есть объекты, а справа классы. Вот. То есть распознавались объекты. Результат распознавания - это список классов, ранжированный в порядке убывания интегрального критерия сходства. Вот здесь у нас интегральный критерий сходства - сумма знаний, а вот здесь у нас интегральный критерий сходства - резонанс знаний. Вот. И мы видим, какой результат идентификации каждого объекта обучающей выборки. Ещё что здесь интересно, ребят, можно сказать? Дело в том, что у нас-то в этих матрицах модели, в матрицах модели... Вот берём модель, например, да? Вот эта колоночка, что она собой представляет? Она представляет собой вектор, который можно представить себе графически в виде некоторой кривулечки, профиля этого класса. И что представляет собой результат идентификации объекта конкретного? Объект тоже представляет собой кривулечку, меандр. То есть там есть признак - единичка, нет признака - нолик. А признаки у нас все перечислены вот здесь. Вот они, видите? И есть формы в базе данных в самой системе, в которых прямо вот эти признаки идут: шкала, градация, шкала, градация. Там всё предназначено для того, чтобы как можно удобнее было использовать при написании работ различных научных. К чему я это говорю? Потому что объект конкретный, он где-то против каких-то вот этих вот значений свойств, признаков будет единичка, а против каких-то нолики. То есть это тоже кривулечка. А что представляет собой вот такая вот форма? Такая форма показывает нам, ребята, что кривулечка, которая изображает мышка один, вот этот массив L с нулями единичками, она, эта кривулечка, может быть представлена как сумма кривулечек класса мышка умножить на 0,92, класса элемент компьютера умноженное на 0,7, класса телефон умноженное на 0,28. И вот так вот все они. Что это такое? У вас была теория рядов, ребят? Про теорию рядов вы знаете, что можно разложить функцию в ряд Тейлора, Маклорена, Лагранжа, Лежандра, Бесселя. Слышали про это, нет, ребят? Ответьте, да, нет? Никто не отвечает. Ну, видимо, природная скромность, наверное, скорее всего. Вот. Но, ребята, я могу вам сказать, что результат решения задачи идентификации представляет собой набор функций классов с уровнями сходства. Вот каждая функция каждого класса, она там есть в базе данных, в базе знаний, системно-когнитивной модели, с каким-то уровнем сходства. Если мы умножим эти функции на эти уровни сходства и сложим, то у нас получится исходный объект. Как в разложении в ряд, любой из рядов. Чебышева ряды. И других учёных. Чебышев вообще классные ряды, кстати. Вот, имеющие ряд достоинств.

Так вот, ребята, я надеюсь, вы поняли, что операция идентификации или распознавания, по сути дела, является операцией разложения в ряд функции, описывающей объект или его состояние, по функциям классов, которые берутся из системно-когнитивной модели. То есть это разложение спектр по классам. Вот насколько мышка похожа на мышка первая, на обобщённый образ мышки вообще, на элемент компьютера и на другие классы. А вот настолько. Вот если их просуммировать, то тогда получится исходный объект.

Есть обратная операция. Обратное преобразование Фурье, например. А вот форма 4.1 3.2, она тоже в системе реализована. Здесь слева видим не объекты распознаваемые, а прямо сами классы. И мы видим, что у нас по каждому классу справа объекты есть, которые больше всего похожи на этот класс. И по некоторым классам у нас есть один объект, например, в качестве примера. По некоторым два, по некоторым там, а вот по классам обобщающим у нас много объектов есть. Вот обобщающие классы. Видите? Элемент компьютера - это обобщающий класс. Много разных есть элементов: мышь 1, монитор 2, мышь 3, монитор 1. И они в разной степени похожи на этот обобщающий класс. Меньше всего похож на обобщающий класс элемент компьютера клавиатура 1, 57%. А больше всего мышка 1. Вот. А вот на средство связи похожи все телефоны. И почти что одинаково, 91-92%. А на спортинвентарь больше всего баскетбольный мяч похож, и футбольный, и теннисный. И меньше всего пинг-понговый мячик похож на спортинвентарь.

О чём это говорит, ребята? Какой вывод из этого можно сделать? Такой, что образ класса отражает статистику объектов, которые к этому классу относятся, их обобщённые свойства этих объектов. И всегда есть и объекты типичные для класса, и всегда есть исключения. Вот. Ну, допустим, берём мы создаём образ студента и образ студентки на основе студентов вот ваших групп. И есть вероятность, что у одного студента будут длинные волосы. Ну есть такой прикид там определённых подмножеств культурологических групп: музыканты, художники - это как бы у них профессионально. И вот, значит, что мы можем сказать? Что этот студент, когда мы посмотрим здесь образ студента, обобщённый образ студента, то он получится похожим на студента, ребят, он получится похожим на студента, не на студентку. Но всё равно будет уровень сходства с классом студент минимальный из всех студентов у него будет. То есть эти длинные волосы, они несут большой объём информации о том, что это не студент, а студентка. У него есть и другие признаки есть, которые больше суммарный объём информации, чем эти длинные волосы. То есть он всё равно будет похож на студента, а не на студентку. Но гораздо в меньшей степени будет похож, чем остальные студенты. Вот. И борода, наличие бороды - это признак очень несущий очень большой объём информации о том, что это студент. Вот. А вы помните, было Евровидение, где там была Кончита Вурст? Первое место заняла. Такая не поймёшь, что она такое. Ну, вроде как девушка, но с бородой. Такой коротенькой, но густой. Помните, нет, Кончиту Вурст? Да, была такая. Ужас какой-то. Ну, в общем, короче говоря, так. Бывает, конечно, и девушки с бородой, но но редко, понимаете, слава Богу. То есть этот признак, он у девушек встречается, но очень редко, понимаете? То есть он будет нести большой объём информации о том, что это парень, и большой объём информации о том, что это не девушка. Ну так же, как вот длинные волосы у парня. То есть, соответственно, Кончита Вурст, она будет на девушек не очень похожа получится. Хотя по остальным признакам она, возможно, и похожа на девушку. Я там не разглядывал особенно, мне прямо что-то дурно стало, когда я её увидел. Вот. Но я не исключаю, что она по остальным признакам похожа, но этот признак он просто вот, я не знаю, делает её очень похожей не на девушку, а на парня. И она будет на девушек похожа в минимальном, с минимальным уровнем сходства из всех девушек. Вот. Ну вот так вот.

**7. Заключение**

Это я всё вам рассказал про решение задачи идентификации, про достоверность модели, решение задачи идентификации. Ну теперь давайте другие задачи решать. Вот. То есть мы изучили, как осуществляется формализация предметной области, это полностью автоматизировано всё в системе. Вот, создали справочники классификационно-описательных шкал и градаций, закодировали исходные данные, создали модели, вот, статистические модели и модели знаний. Потом выберем сейчас наиболее достоверную из них, уже даже выбрали, и в ней провели распознавание. И начинаем решать вот эти задачи. То есть мы действуем согласно логике и схеме преобразования данных в информацию, её в знания. И смотрим результат решения задачи принятия решений, управления, исследования предметной области. Сейчас вот давайте дальше двинемся. Значит, прежде всего про прогнозирование давайте в двух словах скажу. Значит, когда мы смотрим вот на эту таблицу исходных данных, на базу исходных данных, то у нас сейчас статическая интерпретация. Я вам рассказывал, что есть две интерпретации: статическая и динамическая. И ещё есть универсальная терминология. Вот эту форму я показывал вам хелпы программного интерфейса. Вам объяснял, что шкалы могут быть текстовые и числовые. Текстовые могут быть номинальные и порядковые. Что текстовые сортируются с признаком уникальные значения из них вытаскиваются. Числовые находятся минимальное, максимальное значение и делится на заданное число диапазонов. Интерпретация такая: это описательные шкалы и градации, это классификационные. Описательные описывают объект с его свойствами, а классификационные - принадлежность к обобщающим категориям. Это, значит, у нас статическая интерпретация. Динамическая интерпретация: описательные шкалы описывают факторы, действующие на объект управления, а классификационные шкалы описывают будущие состояния объекта управления, в которые он переходит под действием этих факторов. Это динамическая интерпретация. И универсально я просто говорю, что это описательные шкалы и градации, это классификационные. Это такая терминология, которая не поймёшь там, динамическая интерпретация или статическая. Вот. Так вот, ребята, значит, раз у нас задача идентификации решается, то точно так же математически решается задача прогнозирования. Почему? А потому что ничем они друг от друга не отличаются, кроме способа интерпретации э свойств объекта и значений или факторов действующих и будущих состояний. Кстати, вот эта папочка, которая была создана, смотрите, а вот папка приложения, вот, здесь все базы, папка приложения. И в этой папке приложения создана папочка, и там графическая форма. Ребята, все формы записываются графические в папочке со своими именами, всё сохраняется. Но хотя можно, конечно, Alt Print Screen сделать и поместить её там в документ. Но всё есть. Если на компьютере экран, разрешение монитора ниже, чем 1920 на 1080, то система это определяет и тогда на экране показывает масштабированную форму. Ну, допустим, 1360 точек там или 1000, сколько там, 1000, 1000, ну да, 1333, я не помню сейчас точно. Вот. И тогда графическая форма масштабируется до размеров фактических размеров экрана и показывается масштабированном виде. Ну как вот примерно с помощью просмотрщиков изображений. А вот сюда записывается в оригинальном виде с полным разрешением. И здесь ещё с этим же именем будет форма скалет, то есть шкалированная, масштабированная. Вот. Поэтому, значит, если мы интерпретируем свойства объектов как значение факторов вот эти колоночки, а эти как результат действия этих факторов, тогда это будет динамическая интерпретация. В этом случае решение задачи идентификации уже можно рассматривать не как решение задачи идентификации, а как решение задачи прогнозирования. То есть задача идентификации и задача прогнозирования - это одна и та же задача математически, но разный способ интерпретации причин и следствий. При статической интерпретации причины - это значение свойств, а результатом этих причин является принадлежность объекта определённым классам. И причины и следствия по времени они примерно, ну, скажем так, строго говоря, не к одному, но вообще практически к одному моменту времени. Ну, скажем, симптоматика, анализы делали раньше, а потом врач поставил диагноз. Ну считается, что этот диагноз соответствует сейчас этому пациенту. То есть относится к одному моменту времени. А при прогнозировании там осознанно прямо понимают чётко, что причины действовали в прошлом, а результаты их действия наблюдаются в будущем. Вот, например, вчера на бирже межбанковской Московской валютной бирже очень резко увеличилось число банков, которые хотят купить доллары или евро. Как вы думаете, что сегодня будет с курсом евро по отношению к рублю? Ну то, что обычно происходит, то и будет, увеличится курс евро по отношению к рублю. Потому что спрос-предложение определяют цену и так далее, равновесная цена. Короче говоря, получается очень интересная ситуация, что задача идентификации, задача прогнозирования - это одна и та же задача математически, но разный способ интерпретации причин и следствий. Это я хотел сказать. Теперь давайте, есть, конечно, различная терминология в различных областях. Вот в одной области задача идентификации называется идентификация, в другой - распознавание, в третьей - диагностика, в четвёртой - прогнозирование. То есть это называется по-разному. Сама задача идентификации называется по-разному: распознавание, идентификация, классификация, диагностика, прогнозирование. Здесь я не написал варианты. Вот. Смотрим теперь решение задачи принятия решений. Предположим, что у нас, я так надеюсь, у вас фантазия развита, ребят. Представим себе, что эта задача у нас задача прогнозирования решалась. То есть это не признаки объектов, а значения факторов. И мы хотим решить обратную задачу, э обратную задачу прогнозирования. Почему я говорю, что принятие решения является обратной задачей прогнозирования, ребят? Потому что... Вот опять же написано, где будет создана папочка для выходных форм графических. Вот, допустим, берём мы элемент компьютера, модель инф3, то, что я вам говорил. Видим, вот такие вот значения факторов обуславливают переход объекта моделирования, объекта управления в состояние, соответствующее вот этому классу. А эти препятствуют. То есть это классический SWOT-анализ. Ребята из КГУ, вы изучали SWOT-анализ? У вас было это? Вы ж там экономисты, должны были изучать. Скажите, изучали вы, нет? Ну вообще... Ребят, вы изучали SWOT-анализ? Изучали? Молодцы. Да или нет, я не понял только. Изучали. Короче говоря, значит, обычно строится SWOT-матрица, где слева располагаются все факторы, способствующие достижению некоторого результата, справа - препятствующие. Если спросить э дипломников, э магистрантов, которые защищают диссертацию, э соискателей учёных степеней, которые показывают на слайдах SWOT-матрицы, спросить их: "А где вы взяли вот эти вот коэффициенты нагрузки? Вот насколько этот фактор способствует этому переходу объекта вот в это состояние? И насколько этот фактор препятствует переходу в это состояние? Откуда вы взяли вот эти числа?" Обычно вообще там чисел нет, там просто перечислены факторы. Но иногда бывает там что-то вроде баллов там, пять баллов, там, три балла. Да? Ответ одинаковый всегда, ребята: экспертным путём. А кто был экспертом? Да, если так между нами, то сами же и были экспертами. Правильно? Вот. Ну, может, там кто-нибудь и был экспертом. Вот. Система Эйдос является единственной на данный момент системой, которую вообще можно найти в интернете, которая обеспечивает автоматизированный SWOT-анализ. То есть она сама вычисляет количество информации, содержащееся в том или ином значении фактора о переходе объекта под действием этого фактора в определённое будущее состояние. И можно это графически тоже представить себе. И туда это вот всё записывается. Смотрите. SWOT-диаграмма классов модели инф3, приложения. Ну это стандартное учебное приложение, лабораторная работа 3.03, локальная, которая прямо в инсталляции есть. Шкала обобщающий класс, сам класс, градация шкалы, элемент компьютера. Наиболее характерным для этого класса является наличие проводов, наличие кнопок, пластмасса, материал, форма округлая, квадратная, средний размер, цвет чёрный. А наиболее нехарактерными является, что наличие проводов нету их, кнопок нету, форма круглая. Вот, размер один. Смотрите, шкала. Большого размера - это не элемент компьютера, а среднего размера - это элемент компьютера. Ножки есть у него. Если ножки есть, то это не характерно. Сложная форма не характерна. Вот. Вот так мы можем про каждый класс узнать. Значит, почему задача принятия решения является обратной по отношению к задаче прогнозирования? Потому что при прогнозировании мы по значениям факторов определяем, в какому будущему состоянию, какое будущее состояние перейдёт объект моделирования. А при принятии решения мы наоборот, по заданному целевому состоянию определяем, какая необходима система значений факторов, чтобы объект перешёл в это целевое состояние. То есть это чистая классическая обратная задача. Все ли модели интеллектуальных систем обеспечивают решение этой обратной задачи? Я вам скажу, далеко не все. Далеко не все. Почему? Значит, есть одна особенность этих моделей, которая должна быть обязательно, чтобы эта задача в принципе могла решаться. Может быть, система её не решает, но модель позволяет. Просто не сделали режим. А может быть, модель вообще в принципе не позволяет её решить. Значит, вот есть такой метод распознавания - метод К ближайших соседей, где обобщённые образы классов вообще не создаются. А просто результатом распознавания является сравнение конкретного объекта с другими конкретными объектами обучающей выборки. Вот, допустим, берём вашу группу, распознаём, кто у нас вот этот вот конкретный студент? Кто вот он у нас? Результат распознавания. Вот. Вот я сейчас беру и смотрю. Здесь это начинает немножко... Так. Вот Марина есть, да, староста, да, Подгорная? Вот я смотрю и думаю, э, какой результат распознавания Марины по её имиджу? И вижу, что она похожа на Екатерину, на Кристину, на Ольгу, на Викторию. А вот на Олега там, на Даниила, э, вот, на Артёма не похожа. Это называется метод К ближайших соседей. То есть просто перечисляется какое-то число объектов обучающей выборки, на которые она больше всего похожа и больше всего не похожа. Или или даже только те, на кого она похожа, перечисляются. Это вот и есть результат распознавания. А как мы поступаем, когда модель позволяет создавать классы? Создаются обобщённые образы классов на основе ряда примеров. И объект сравнивается не с другими объектами распознаваемыми, а он сравнивается с обобщёнными образами классов. Так вот, если обобщённые образы классов есть, как вот вы видели в этих формах экранных, которые я показывал, то тогда можно решить обратную задачу. То есть можно взять э модель... Сейчас я вам покажу, как это делается. Вот берём модель какую-то, в которой мы решаем задачу. Вот, допустим, модель инф3. Вот нас интересует, какие значения факторов необходимо использовать, чтобы объект перешёл в состояние, соответствующее вот этому классу четырнадцатому, обобщающий класс, элемент компьютера. Что для этого нужно сделать, ребят? Для этого мы просто сортируем вот эту базу данных, соответствующую этой модели. Ну это таблица просто, база данных, матрица. Сортируем по этой колоночке вот, соответствующей этому классу, в порядке убывания. Ну, естественно, все целиком строчки, целиком записи сортируются. Это можно делать не физически, а логически, просто индексный массив формируется. Это происходит гораздо быстрее, чем физическая сортировка, особенно когда база большие. И вот э результат этой сортировки у нас такой, что больше всего способствует переходу объекта в состояние, соответствующее этому классу, вот такое значение фактора. Поменьше такое. А в самом конце будут те, которые просто препятствуют. В самом в самом конце будут те, которые препятствуют в наибольшей степени, потом те, которые препятствуют несколько в меньшей степени. Понятно, да, ребят? Как вы оцениваете? Ну, в общем, понятно. Так вот я могу вам сказать, что эта задача решается мгновенно. Вот всё, она решена уже. И даже на больших моделях она решается мгновенно. Почему, знаете? Ну, индексный массив создаётся оптимальным методом двоичного поиска. Ну оптимальные методы математические используются. И получается это очень быстро, очень хорошо. То есть задача принятия решения решается мгновенно фактически. Вот. Если же мы будем пытаться её решать путём прогнозирования, то есть будем многократно осуществлять прогнозирование при разных значениях факторов, при разных сочетаниях значений всех факторов. У нас будет L в степени n по М число сочетаний значений факторов. У нас могут быть там сотни тысяч раз нужно спрогнозировать, чтобы узнать, что будет завтра. А мы можем это сделать такое число прогнозирования, только если неделю потратим. То есть мы не сможем за необходимое время спрогнозировать на необходимый период. То есть нужно большее время. А вот таким образом, как решение обратной задачи, это получается быстро очень. Смотрим теперь следующее, что можно ещё узнать интересного. Мы можем узнать, как влияет любое значение фактора на результат. Опять нет папочки. Это называется инвертированные SWOT-диаграммы. Такого в науке не было, это я предложил инвертированные SWOT-диаграммы. Ну, это сходно, знаете, с чем? Семантический потенциал фактора. То есть это что значит? Какой смысл этого вот признака? Вот признак, допустим, материал пластмасса. О чём этот признак говорит? О том, что скорее всего, это элемент компьютера, возможно, мышка, телефон, средство связи, клавиатура или монитор. И ни в коем случае это не мебель, не аксессуар, не сумка, не спортинвентарь, не мячи, не стулья, не вешалки. То есть это смысл этого признака. Ребята, система сама это вычисляет этот смысл на основе эмпирических данных. Как и SWOT-диаграмму. И знает это по всем признакам. Значит, фактически, когда эта задача решается, просто сортируется строка матрицы модели. Берётся какая-то модель, вот строка, допустим, там материал там или что там, пластмасса, да, вот, допустим, семнадцатая. Вот. Просто берётся эта строка и сортируется в порядке убывания количества информации в этом признаке о принадлежности классов к этим вот, то есть объектов к этим классам. И классы располагаются в порядке убывания количества информации о принадлежности к ним в этом значении фактора. Понятно, да? То есть это делается всё просто, ребята, просто, понимаете? Что вот, собственно говоря, интересно тоже. Следующее. Мы можем посмотреть, а что у нас за элемент компьютера с точки зрения нейронных сетей? Мы можем увидеть э в интерпретации нелокального нейрона это. Нелокальные нейронные сети я предложил в 2003 году в статье. Это один слой нелокальной нейронной сети. Система Эйдос позволяет многослойные модели строить. Несложно, причём, это делается совершенно. Я это делал ещё там в девяностых годах. Так вот, ребята, смотрите. Значит, здесь у нас вверху классы, и они в этой интерпретации нейросетевой - это нейроны. А здесь у нас значения свойств или значения факторов. В этой интерпретации нейросетевой - это рецепторы и нагрузки на этих рецепторах. И мы что делаем? Мы вычисляем все связи каждого рецептора со каждым из нейронов, все, абсолютно все, какие только есть. А потом сортируем в порядке убывания модуля этих связей. И рисуем те, которые помещаются на экране при данных параметрах. Здесь вот у нас эти параметры перечислены. Значит, я могу эти параметры поменять. Вот я, допустим, могу сказать, что вот у нас нейронов 16, а всего их 14, хватит. Максимальное число связей - 1000. Диапазон отображаемых рецепторов - 50. Максимальное число отображаемых рецепторов - 16. Вот. Я сейчас возьму и задам 50. Вот. И посмотрим, что получится. Получится, что там ничего не поймёшь. То есть уже наползают наименования, но коды ещё видны. И я могу вам сказать, что оно так, в общем-то, симпатично выглядит, но на самом деле уже нечитабельно и нет смысла особого в этом, потому что вот эти вот связи, которые здесь справа - это связи, которые практически ни на что не влияют. То есть это можно такое вот для того, чтобы пугать детей показывать такие диаграммы. Вот здесь видите, 100% отображено информации, которая есть в сети на данный момент. А если здесь мы зададим стандартное это значение, когда хорошо всё прорисовывается, там получится не 100%, а 60%. А бывают задачи, когда вот рисуешь такую вот подмножество, а там получается 3% там или 5%. Вот. Смотрим дальше. Интегральные когнитивные карты. Сразу я просто беру и... Нейронов беру пять, вот, а рецепторов беру семь. Потому что иначе мы не поймём, что там. Ага. Видите, значит, когда нарушаешь последовательность обработки, преобразования данных в информацию, её в знания, она нам высказывает всё, что она нас думает система. Она говорит: "Перед этим, чтобы вы хотите сейчас посмотреть, надо вот эти режимы выполнить". Ну давай их и выполним. Значит, э режим 4.2 2.1 - это расчёт матрицы сходства в разных моделях. Посчитали. Теперь мы можем посмотреть эту матрицу сходства в виде базы данных, она открывается в Экселе. Вот. Можем посмотреть в виде когнитивной диаграммы. Тоже куда она будет записана. Здесь мы видим уже решение задачи, ребята, другой. Не сравнение конкретного объекта с обобщёнными образами классов, а сравнение самих обобщённых образов классов друг с другом. И можем это же сравнение провести в форме агломеративной дендрограммы кластерного анализа провести. Так, ребята, у нас сейчас, похоже, что конец занятия, да? 11:10. Значит, ребята, из КГУ, вы можете уже... Так, вот дендрограмма. Дендрограмма. Из КГУ, ребята, у вас конец занятия, поэтому вы можете, в принципе, покидать совещание, но можете посмотреть ещё несколько минут. 5 минут будет. Вот мы здесь видим, ребята, вот эта шкала различий, а это перечислены классы. И мы видим, насколько какие классы друг на друга похожи и не похожи. И видим, что эти классы все образуют кластеры. Вот стул и мебель, допустим, кластер. И ещё туда стул и вешалка входят. Вот это кластер, видите, вот он образуется. Вот. И это образует большие кластеры, большие, на самом верхнем уровне различий, сходств и различий. С одной стороны всё, что связано с компьютерами, а также с телефоном, а на другой всё, что связано с мебелью и всякими сумками там и спортинвентарём. Вот так вот. Это называется конструкт. Про конструкт я вам, кажется, рассказывал на предыдущих занятиях. Всё, ребята, для КГУ конец занятия. Надеюсь, вам понравилось. И сейчас, собственно, и в аграрном тоже конец занятия. Значит, пожалуйста, если какие-то есть вопросы, можете спросить, ребята. Давайте. Какие есть вопросы? Пожалуйста, спрашивайте. Ну, у студентов всегда два вопроса. Один спрашивает, когда мы когда закончится занятие, а второй - где взять там денег, я не знаю, так что-нибудь такое, да? А ещё вопросы есть вот по теме самой занятия? Ну кто-нибудь отвечает, а то что вы все так сидите там? Вы это там вообще есть вот Марина Подгорная, староста? Вопросов не имеем, всё понятно. Спасибо большое. Пожалуйста. Значит, тогда мы в следующем занятии продолжим решение задач исследования моделируемой предметной области путём исследования её модели. Как вы догадались, мы уже решили задачу идентификации, прогнозирования, принятия решений. И сейчас уже занялись исследованием моделируемой предметной области путём исследования её модели. Вот эти формы, которые я последние вам показывал, они к этому именно относятся. Но я показал не все. Сейчас мы в следующем занятии мы продолжим эту тему. Именно вот исследование объекта путём исследования её модели, не объекта, а моделируемой предметной области путём исследования её модели. Всё, ребята, всего самого хорошего, до свидания. До свидания. До свидания. Спасибо, всего доброго. До свидания.