С. В. Конявская-Счастная,
А. Ю. Ситников,
П. А. Галманов,
С. С. Буянов,
С. Г. Ищанова

ИНФОРМАЦИОННЫЕ ПРОЦЕССЫ В СИСТЕМАХ ДОВЕРЕННОГО ИИ

Москва
«Вишневый пирог»
2026

УДК 004.89
ББК 30ф

Рецензенты:
Геннадий Викторович Росс, доктор экономических наук, доктор технических наук, академик РАЕН, советник генерального директора АО «ОКБ САПР»
Шамиль Гасангусейнович Магомедов, доктор технических наук, директор института искусственного интеллекта РТУ МИРЭА

Конявская-Счастная С. В., Ситников А. Ю., Галманов П. А., Буянов С. С., Ищанова С. Г.
Информационные процессы в системах доверенного ИИ. М.: «Вишневый пирог», 2026. – 302 с.

ISBN 978-5-6056038-2-5

Книга представляет собой коллективную монографию, каждая глава которой касается одного из актуальных вопросов защиты информации в системах доверенного искусственного интеллекта, над которыми размышляют участники постоянного научного семинара кафедры «Защита информации» ФРКТ МФТИ. В каждой главе – вопросы и перспективы исследований, которые станут началом рационализации бурного, но пока стихийного процесса интеграции искусственного интеллекта в жизнь общества.

ISBN 978-5-6056038-2-5

ОГЛАВЛЕНИЕ

ПРЕДИСЛОВИЕ

Конявский В. А., д.т.н.

Новый инструмент всегда порождает новые проблемы.

Создание дорог породило разбойников, а борьба с ними – службу шерифов.

Флот – пиратов, ну и дальше – морской патруль.

Вслед за компьютерами появились хакеры, и как следствие – наша профессия: защита информации.

Новым мощным инструментом становится искусственный интеллект (ИИ), и он, в свою очередь, непременно создаст «что-то весьма неприличное»[1]. И чем же мы на это ответим? Ответом на очередной вызов станут системы доверенного ИИ – но что это такое? В условиях формирующейся терминологии нельзя точно определить даже базовые понятия – собственно искусственный интеллект, технологии искусственного интеллекта, системы искусственного интеллекта, доверие к системам искусственного интеллекта. Множество вопросов, и кому, как не самым юным и амбициозным молодым ученым отвечать на эти вопросы? Конечно, даже для «подхода к снаряду» нужно освоить огромный объем знаний, но с этим нормально – МФТИ дает одно из лучших в мире технических образований, а кафедра «Защита информации» пытается увязать теоретические знания с тематикой защиты информации и, в частности, с доверенностью, как принципиально новым феноменом.

Это непростая задача. Мы об этом много думаем, и со ссылкой на [1] подумаем еще раз.

[1] «Возле города Пекина», В. С. Высоцкий

Как только человечество сталкивается с чем-то новым, оно начинает испытывать в его отношении серьезные опасения и глубокую обеспокоенность, даже если это новое абсолютно рукотворно. Оставляя в стороне такую давнюю историю, как луддиты, даже в самом последнем времени, времени постоянного и очень быстрого технического прогресса, можно почерпнуть массу примеров. Мы (люди) были обеспокоены роботами еще до их появления, компьютерами в целом, компьютерными вирусами, 5G-вышками, ИНН. Нет ничего удивительного в том, что искусственный интеллект тоже вызывает глубокую обеспокоенность и желание что-то с этим сделать.

Опыт предыдущих обеспокоенностей подсказывает, что по-настоящему важно – определить, в какой мере то, с чем мы имеем дело:

1) действительно так уж качественно ново, и

2) насколько оно может на нас влиять.

Вечные опасения, присущие человеку как фундаментальные (что власть над людьми захватят какие-то «не люди», опасения потерять свою ценность для общества (или стоимость на рынке труда), а также опасения, что сознанием/мнением/чувствами человека будут манипулировать и обманывать) всколыхнули на этот раз достижения в области искусственного интеллекта. Отсюда, например, рассуждения о необходимости информирования человека о том, что он начинает взаимодействие не с человеком (в различных формулировках), с целью получения его информированного согласия на это. Хотя, когда нам в метро объявляют следующую станцию, это тоже делает не человек, и отсутствие предупреждения об этом пока не привело ни к какой катастрофе. К практике, кстати, такое требование зачастую мало применимо: например, в какой момент нужно предупредить о том, что трамвай, в который заходят люди на остановке, – беспилотный? Как будет выглядеть поездка при условии получения информированного согласия каждого пассажира? И почему этого не нужно делать, когда самолет сажает автопилот?

Манипулирование сознанием и чувствами людей – это, действительно, проблема, и то, что человечество задумывается о ней

при каждом удобном случае – хорошо, или по крайней мере, нормально. Однако вряд ли именно ИИ представляет в этом отношении самую главную угрозу.

Как правило, феномен становится менее пугающим, становясь более изученным. Так, наверняка будет и с ИИ.

Специалистам известно, что целое – есть единство формы и содержания. Числа – это форма. Содержание – это семантика. Алан Тьюринг этого еще не заметил – у него содержанием оперировал человек, изучающий последовательности символов на наличие семантики – если семантика появилась, то ключ шифрования найден. А вот Марвин Минский уже хорошо это знал, написав поистине замечательный труд «Фреймы для представления знаний» [2]. Кейслер, Тарский, Поспелов – известные ученые, изучавшие теорию моделей, семиотику, теорию языков и смежные аспекты. Главное положение в современной науке – знания фиксируются в виде семантических активов (СА) – словарей, справочников, классификаторов, онтологий и других, при этом не в виде цифр и чисел, а в виде связанных данных. СА содержат описания нашего представления об объектах и сущностях предметной области. Заметим – не описание объектов, а описание представления об объекте. Основа работы с семантикой – система управления знаниями (СУЗ). А «связанность данных» – это связь формы и содержания стандартизованной конструкцией – например, фреймами М. Минского или образами У. Гренандера [3]. С высоким уровнем уверенности можно сказать, что необходимой характеристикой наличия ИИ является наличие формализованных СА и СУЗ.

Знания – фиксируются в виде СА. На функции сознания похоже функционирование системы управления знаниями. Понятие «навыки» похоже на «алгоритм». Зачастую навыков достаточно для успешного функционирования в быту, особенно при выполнении относительно однообразных и достаточно несложных операций. А при наличии некоторых «умений» изменить алгоритм и модифицировать поведение, соответствующее «навыку» –

исполнителя зачастую можно идентифицировать уже как специалиста.

Принципиально важным является понимание, что только владение всеми составляющими триады позволяет говорить наличии у человека той или иной степени интеллекта. И, значит, говорить об ИИ можно лишь тогда, когда умения и навыки основаны на знаниях[2]. Знания должны быть формализованы для возможности их обработки в СИИ, и представляться в виде семантических активов.

Каждую из составляющих триады «знания-умения-навыки» можно считать некоторым проявлением интеллекта. А каждое сочетание элементов триады – вполне хорошее основание для классификации *алгоритмической составляющей* систем ИИ по глубине присущей ей интеллектуальности.

Есть и другое основание для классификации – функция, профессия в «человеческой» терминологии. Это будет классификация *моделей* (ведь речь тут идет об «умениях»). Здесь можно выделить:

- простые (например, большие языковые модели (БЯЗ или LLM), кредитный скоринг и др.);
- расширенные (например, БЯЗ с контекстом – RAG+LLM);
- гибридные (например, с использованием систем управления знаниями (СУЗ) – СУЗ+RAG+LLM).

С точки зрения безопасности – классификация будет уже другой, классификацией *систем ИИ*. Здесь уже мы должны говорить не об ИИ, моделях и технологиях, а именно о системах ИИ, СИИ.

Действительно, вряд ли технология может сама по себе чем-то навредить человеку. Используются системы, человек взаимодействует с системами, и поэтому требования по безопасности должны выдвигаться для систем.

В свою очередь, системы ИИ вполне можно разделить по степени их влияния на аналоговый мир на физическом уровне. Это в

[2] Благодарю А. И. Сурыгина, д.п.н., профессора за глубокое обсуждение этого вопроса

свою очередь детерминировано степенью их автономности от оператора.

Так, СИИ можно разделить на:

- виртуальные (функционируют под управлением человека, влияния на аналоговый мир на физическом уровне оказать не могут – результаты их работы виртуальные),

- кибер-физические (функционируют под контролем человека (например, автопилот), оказывают влияние на аналоговый мир на физическом уровне, но человек может в случае необходимости перехватить управление и предотвратить негативное влияние),

- автоматические (функционируют полностью автономно (например, робот-доставщик), оказывает влияние на аналоговый мир на физическом уровне, и оно не может быть скорректировано управляющим персоналом, так как этого персонала нет).

Для каждого из этих видов нужно формулировать свои требования по защите информации. При этом основы доверенности должны закладываться от сбора данных для обучения до принятия решений и исполнения их.

Доверенность же систем искусственного интеллекта, в частности, разговорных систем, робототехнических систем, беспилотного транспорта необходимо рассматривать в нескольких взаимосвязанных аспектах:

- доверенность методов и алгоритмов (при каких условиях их можно считать доверенными, как установить доверенное состояние системы),

- функциональная безопасность (безопасность функционирования при обеспечении условий доверенности),

- устойчивость системы в условиях агрессивной среды (при фиксированных уровнях нарушения условий доверенности),

- защищенность от стороннего вмешательства (например, в структуру и содержание базы знаний, в том числе в процессах дообучения, а также в последовательность сигналов управления).

На основе анализа публикаций в открытой печати можно считать, что эта тематика пока недостаточно раскрыта в научных исследованиях и практике. В частности, широко известны случаи перехвата хакерами управления беспилотным автотранспортом и

БПЛА, что в достаточной степени иллюстрирует опасность применения систем ИИ без использования механизмов обеспечения доверия.

Развитие направления обеспечения доверенного характера разрабатываемых математических моделей, алгоритмов искусственного интеллекта, программного или аппаратно-программного обеспечения связано с анализом и фиксацией обучающих выборок, анализа устойчивости и обеспечения целостности гибридных схем ИИ, выработкой требований, рекомендаций и набора контрольных процедур для установления соответствия требованиям доверенности при дообучении системы, обеспечения доверенного взаимодействия по каналам связи с помощью доверенных аппаратно-программных средств защиты информации.

Основная задача этого направления – обеспечить доверенное взаимодействие как в массовых применениях (кто-нибудь, да и попадется), так и при попытке реализации целевой атаки (цель – конкретный человек).

Значительная часть требований к защите информации вполне может быть позаимствована из уже давно сформулированных, описанных и освоенных требований. Так, для программных систем на этапе эксплуатации практически не видится принципиальных отличий СИИ от любой информационной системы (ИС) аналогичной области применения.

Область применения тут имеет принципиальное значение. Если это, к примеру, информационная система государственных органов, государственных унитарных предприятий, государственных учреждений, или, тем более, государственная информационная система (№216-ФЗ в редакции от 08.08.2024[3]), то необходимо как минимум принять все необходимые меры по защите обрабатываемой в этих ИС информации в соответствии с требованиями, устанавливаемым регуляторами.

[3] Федеральный закон от 08.08.2024 № 216-ФЗ «О внесении изменений в Федеральный закон "Об информации, информационных технологиях и о защите информации" и отдельные законодательные акты Российской Федерации»

Однако в СИИ, обученных на больших наборах данных, возникают и новые уязвимости. Так, по существу, значительная часть таких систем может быть описана как статистическая база данных с неконтролируемым потоком запросов. В этом случае, анализируя большое число результатов запросов, зачастую можно восстановить исходные наборы данных. Это является косвенной утечкой, требований к поиску которых пока нет. Возникает задача обеспечения дифференциальной конфиденциальности, включая блокирование целевых атак – особенно это может быть существенным для, например, задач кредитного и страхового скоринга, антифрода и других задач этого класса. На этом пути уже получен ряд фундаментальных результатов в анализе и блокировании вредоносных запросов при известных алгоритмах работы СИИ.

Существенно сложнее обстоят дела в кибер-физических системах. Примером может быть беспилотный транспорт в различных своих вариантах – например, взаимодействие Сапсана с БПЛА, летящим перед ним для контроля состояния пути. Ну, или БПЛА, используемые в качестве летающих светофоров для управления потоками перемещения людей и автотранспортных средств. И здесь тот же принцип – все известные меры должны быть применены, но при этом и расширены механизмами блокирования специфических угроз.

А вот в автоматических системах вряд ли есть существенные различия – безопасность человека, животных и растений регулируется техническими регламентами, что целесообразно зафиксировать как правильную меру.

Отдельным направлением исследования может быть применение систем ИИ для поиска уязвимостей в системах ИИ, а также в повышении уровня доверия – например, на основе мажоритарных механизмов (арбитраж) с близкими обучающими последовательностями и близкими семантическими средами. Необходимо изучить возможность применения подходов класса «слой доверия» из группы семантических стандартов W3C, присвоения набору данных некоторого индекса (уровня) доверия и

в соответствии с этим оценивать доверие к выводам, сделанным на основе этих данных.

Итак, интеллект человека – это знания, умения и навыки. Знания, а не сведения или сообщения. Соответственно, аналогия для ИИ – не данные, а семантические активы (СА). Инструмент работы с ними – система управления знаниями (СУЗ), аналог сознания человека (в отличие от рефлексов). Умения – это модели, навыки – алгоритмы. Наличие всей триады говорит о наличии ИИ.

Проявления интеллектуальности могут характеризовать алгоритмическую составляющую системы (и это может быть основой для их классификации как более или менее интеллектуальных), структурная сложность характеризует модели (и по этой характеристике их можно разделить на простые, сложные и гибридные), а системы искусственного интеллекта характеризуются степенью автономности и степенью влияния на аналоговый мир на физическом уровне – и именно это важно для предъявления требований по защите информации.

Мне известны десятки диссертаций и монографий (и моя докторская не стала исключением), в которых авторы анализируют десятки подходов к понятию «информация», пытаясь нащупать нечто, позволяющее строить продуктивные модели. Ох, нелегко идут эти исследования, что и понятно: нужно точно проследить варианты проявления (бытования) одной их базовых категорий и описать связанные с этим закономерности. Вряд ли это под силу одному человеку на современном этапе, но нельзя все же и оставлять без внимания бушующую терминологическую путаницу, мешающую пониманию основ.

Довольно большой, но не исчерпывающий перечень определений информации можно увидеть в [4], здесь ограничимся только самыми полезными для формирования понятийной базы.

В первую очередь, это определение, приведенное в статье Н. А. Кузнецова как одно из классических. Оно относится к алгоритмическому подходу к информации, в частности, А. Н. Колмогорова, и уже давно используется как народное, без ссылки на автора.

Информация есть сущность, сохраняющаяся при вычислимом изоморфизме [5].

Это определение – через атрибут, и не дает представления о самой природе этой «сущности». Большинство подходов к определению информации выделяют у информации следующее ключевое свойство: отражать движение объектов материального мира в некоторой системе, которой это необходимо для адаптации к изменениям окружающей действительности.

Здесь ключевыми моментами является, во-первых, *отражение* (отражение движения, а не само движение, является информацией), а во-вторых, необходимость этого отражения, его целесообразность для системы (бесполезный шум, даже членораздельный (понятный), информацией не является).

В соответствии с «органической концепцией информации» [6. С. 14–27] информация – это категория, присущая *только живой природе*. В первую очередь потому, что для нас на сегодняшний день не очевидно, каким образом в неживой природе осуществляется целеполагание (и осуществляется ли оно там вообще). А вот системы живой природы к изменениям окружающей среды целенаправленно адаптируются, и поэтому нуждаются в том, чтобы как-то эти изменения отслеживать и желательно – прогнозировать.

Таким образом, получаем, что **информация** – это отражение движения объектов материального мира в системах живой природы [6. С. 15].

Нужно учитывать, что «движение» здесь понимается предельно широко – не только в физическом, но и химическом, биологическом и социальном смысле. Вне философского контекста, возможно, было бы точнее слово «изменение». В материалистической философии «движением» называют любые виды изменчивости, наблюдаемые в объективном (то есть существующем независимо от наблюдателя) мире. Именно в этом смысле слово «движение» используется в этом определении.

Слово «отражение» тоже используется как категория диалектического материализма. Это приобретение материальным объектом некоторых свойств под воздействием свойств объекта,

который с ним взаимодействует (**взаимодействие** – это взаимное изменение объектов материального мира).

Отражение в живой природе имеет особенности, связанные с тем, что она живая, то есть все ее системы являются организмами или состоят из организмов. «Отражение» в организмах состоит в изменении происходящих в этих организмах биохимических преобразований. Результаты этих изменений используются организмами, чтобы выбрать вариант поведения, наилучшим образом отвечающий его цели. Эти результаты являются формой существования информации в организме и называются «сведения».

Сведения – это запечатленные в организме результаты отражения движения объектов материального мира [6. С. 16].

Обратим внимание, что сведения не равны информации, их отличает «запечатление» и «результат».

Сведениями становятся только те результаты отражения движения объектов материального мира, которые запечатлелись в организме.

Смысл компонента «результат» просто понять на примере: одна и та же «входная» информация, воспринятая разными организмами (например, ребенком, собакой и рыбкой) породит в этих субъектах разные сведения.

Все сведения, которые есть в распоряжении организма, формируют так называемую «информационную модель мира», которая постоянно корректируется при поступлении новых сведений (в зависимости от особенностей организма, таких как большее или меньшее разнообразие вариантов адаптации к изменениям окружающей среды, и тому подобных). Организмы с высокоразвитой нервной системой и психикой способны при этом вырабатывать «знания».

Знание – это производные сведения о закономерностях изменения состояния отражаемых объектов материального мира [6. С. 17]. Ключевые пункты определения – «производные» и «о закономерностях». Сведения эти «производные» в том смысле, что это сведения не об объектах и не об их изменении, а о закономерностях их изменений. Они являются результатом не

отражения чего-либо, а операций с другими сведениями (обобщения, вывода). А то, что они касаются закономерностей, обеспечивает их целевую функцию – прогнозирования на основе поступающих сведений дальнейших изменений состояния окружающей среды.

Социальные организмы – люди – обладают способностью обмениваться сведениями. И осуществляется этот обмен через «сообщения».

Сообщение – это набор знаков, с помощью которого сведения могут быть переданы другому организму и восприняты им [6. С. 18].

Из этого определения очевидно, что сообщение – это форма существования сведений вне организма, при их передаче. Сообщение порождает сведения в том организме, который его получил, в этом смысле можно утверждать, что оно «содержит» сведения. Однако очевидно, что одно и то же сообщение порождает разные сведения в разных организмах (аналогично порождению разных сведений при получении одной и той же информации).

В своей работе [4. С. 179–191] я не опирался на выкладки А. А. Стрельцова, так как она написана раньше, чем [6], однако выделил именно эту точку отображения в сообщение – в материальную форму (все, что до этого происходит с информацией – не материально) как отправную точку для начала рассмотрения в качестве объекта защиты.

Объединив наши выкладки, можно сформулировать, что сведения могут «материализоваться» в предмет или в процесс. Предмет – это то, что зафиксировано в пространстве, процесс же – это то, что зафиксировано во времени.

Так, речь – не фиксируется в пространстве, но фиксируется во времени. Напечатанный текст фиксируется в пространстве, но не фиксируется во времени.

Предмет и процесс – это две формы существования сообщения. Сообщение в форме предмета называются *данными*, а в форме процесса – *процессами*.

Свойство материализации сведений, не рассмотренное А. А. Стрельцовым, заключается в упорядоченности знаков в

сообщении (будь оно в форме данных или процесса). Буквы, звуки, точки, насечки – любые знаки, с помощью которых субъект порождает сообщение, содержат сведения до тех пор, пока, они сохраняются в заданном порядке. Чтобы проверить это, достаточно переставить буквы в тексте или прокрутить аудиозапись от конца к началу.

Таким образом, определение сообщения целесообразно скорректировать следующим образом:

Сообщение – это упорядоченный набор знаков, с помощью которого сведения могут быть переданы другому субъекту и проинтерпретированы им.

Организм заменен на субъект, а восприятие на интерпретацию, так как в технических системах способностью порождать сообщения обладают не только организмы. Датчики могут отображать в форме упорядоченной последовательности знаков результаты измерений, процессы – результаты выполнения преобразований, все это передается в виде сообщений в той или иной знаковой системе. Аналогично «воспринимать» – интерпретировать сообщения, использовать их как входные данные могут тоже не только организмы, но и процессы, средства обработки, вывода и т. д.

Важным понятием, связанным с информационным взаимодействием, является среда, для существования в которой формируется сообщение. От нее зависит знаковая система сообщения и его форма (процесс или данные).

Одно и то же сообщение может существовать на определенном отрезке своего жизненного цикла в виде данных, а на другом – в виде процесса, может «перекодироваться» из одной знаковой системы в другую. Залогом сохранения тождества в этом случае будет именно сохранение заданного порядка знаков различной природы (сигналов) – вычислимый изоморфизм.

Сообщения (данные и процессы) как упорядоченный набор знаков, предназначенный для применения в определенной среде, формируются только с целью взаимодействия. Поэтому защита

информации – это всегда защита *информационного взаимодействия*.

Информационное взаимодействие называется защищенным, если обеспечивается вычислимый изоморфизм – сохраняется отношение упорядоченности множества сигналов, маркированных как «информация» [4. С. 181].

Подводя итог: информация является одним из проявлений категории «время», движение (в философском смысле) информации реализуется во времени, формы бытования информации – в человеческом обществе – это сведения и сообщения – в форме данных или процессов.

Процесс развивается во времени = информация бытует в виде процессов.

Технической защите подлежат данные, так как сведения (и знания) бытуют только в сознании человека. И это остается так до тех пор, пока мы не начинаем извлекать знания из данных! То есть – до начала опытов с ИИ. Начиная эти опыты, мы замечаем, что данные (и вместе с этим – сведения и информация в целом) могут характеризоваться ценностью (для оценки) в применении к анализу решаемой задачи (если хотите – движения материи). Ценность со временем падает (энтропия?), и возрастать может лишь при добавлении новых данных в процессах (разворачивающихся во времени) комплексирования наборов данных. Заметим: может, только может. Но это не обязательно. Это, теперь ясно, зависит от качества данных и от задачи.

Ну а раз ценность информации непостоянна, то нужно оценить, насколько ей можно доверять – именно «здесь и сейчас», учитывая движение времени, пространства, и цель наблюдаемого движения.

Впервые термин «Доверенная вычислительная система» (ДВС) появился в [7] в 1999 году, но тогда он касался корпоративных систем, состав и связи которых известны и зафиксированы (других не было). И, конечно, тогда говорилось о данных, семантика которых подразумевалась, но не определялась и прямо не учитывалась. Считалось, что традиционные механизмы управления доступом, такие как дискреционный и мандатный механизмы, вполне способны в

доверенной среде решить задачу блокировки утечек. Однако с тех пор многовато воды утекло, и если с традиционными корпоративными системами все более или менее понятно, то для стремительно растущих открытых систем, в которых нет зафиксированного перечня пользователей и технических средств, никаких специальных требований и рекомендаций до сих пор нет. Как минимум, гарантированный уровень защищенности в таких системах становится невозможным, риски остаются, и для их блокирования нужно применять не только организационно-технические, но и экономические меры, включая страхование остаточных рисков [8]. Этот подход был отражен уже в первой «Доктрине информационной безопасности Российской Федерации»[4] – как «создание системы страхования информационных рисков физических и юридических лиц». Пока эффективная система страхования информационных рисков так и не создана, но актуальность ее быстро растет, и не за горами ее реальная востребованность, так как в открытых системах гарантированная защищенность вряд ли достижима. Скорее, нужно оценить риски, и если они осознаны – то выбрать эффективный уровень защищенности и уровень экономических механизмов, компенсирующий возможные потери.

Следующим заметным шагом стало понимание необходимости учета семантики – как для обеспечения семантической интер-операбельности [9], так и для защиты информации [10]. Целое — это единство формы и содержания. Цифры — это форма. Содержание — семантика. Машина Тьюринга оперирует с формами, а содержание — результат трактовки, которая выполняется приложением. В этом — и ошибка великого Тьюринга, и беда современной “computer science”. Об этом я писал в предисловии к книге [9], для выпуска которой осуществлял научное редактирование, и думаю, будет не лишним разместить этот же текст и как второе предисловие к этой книге – с учетом важности семантики для СИИ и безопасности.

[4] Доктрина информационной безопасности Российской Федерации (утв. Президентом РФ от 9 сентября 2000 г. N Пр-1895)

И вот мы добрались до современных проблем с доверенностью систем искусственного интеллекта. Заметим – здесь мы не касаемся применения СИИ. Варианты применения практически неограниченны, вернее, ограничены только уровнем доверия. В ближайшее время пройдет эйфория от «поговорить», и потребуется «посоветоваться». А для этого даже простой «собеседник» должен быть не соседом, а экспертом. А уж тем более доверенность необходима при применении ИИ в системах беспилотного транспорта, киберфизических системах, и практически в любых on-line системах. Сосредоточимся на доверенности, сославшись на мысль, высказанную зам. министра Минцифразвития по информационной безопасности А. М. Шойтовым на встрече с учеными МФТИ 13.04.2021 года, а именно – в области обеспечения доверия к системам искусственного и интеллекта выделяются два вопроса: как сделать СИИ доверенной; и как использовать СИИ для обеспечения доверия. Из этой вполне справедливой дихотомии будем исходить и мы.

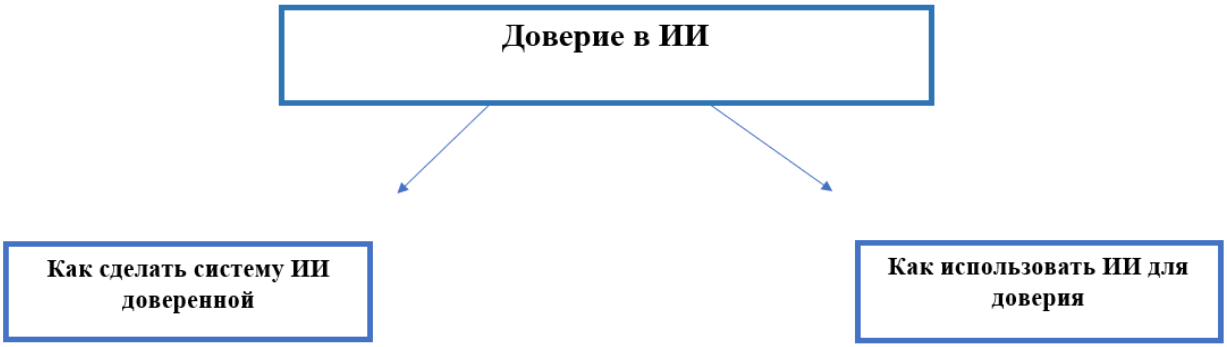


Рис. 1. Доверие в ИИ

КАК СДЕЛАТЬ СИИ ДОВЕРЕННОЙ, ИЛИ ЧЕМ ОБЕСПЕЧИВАЕТСЯ ДОВЕРИЕ К СИСТЕМАМ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Доверие к СИИ обеспечивается:

- *Доверием к коду*. Код корректен, не содержит вредоносного ПО (ВрПО), объясним и понятен. Это все – технологии безопасного программирования. Огромный объем исследований в этой области ведет ИСП РАН под руководством академика А. И. Аветисяна, результаты которых широко известны и легли в основу ряда государственных стандартов.

- *Доверием к данным*. Это сложная позиция – нужно учитывать источник данных, их актуальность и адекватность, цель сбора, преобразования при слиянии и выделении частей, да и семантику в целом. Нужно научиться учитывать семантику обрабатываемых данных – и в этом основная и важнейшая задача в движении к доверенным СИИ.
- *Доверием к среде функционирования*. Проверенные данные и проверенное ПО дают правильный результат только в правильной среде функционирования. Человек взаимодействует с СИИ через интерфейс среды. И самый простой способ исказить результаты — это исказить интерпретацию данных в среде. Универсальный вычислитель Тьюринга (а это все наши вычислительные средства) работает только с формой, и не может учесть семантику. Результат «2» — что это? 2 нанометра в микроэлектронике или 2 млн баррелей водоизмещение судна? Ответ на это до сих под дает приложение, а не вычисление. Конечно же, если трактовка результата возлагается на недоверенную среду – результат может трактоваться, как угодно. Среда функционирования – важнейшая составляющая доверенных СИИ.
- *Доверием к результату*, который не должен быть источником утечек. Достаточность результата – важнейшая характеристика. Здесь дело в том, что наборы данных, используемые для обучения модели ИИ, практически всегда содержат защищаемые данные. Естественно, что следы этих данных находятся и в составе параметров обученной модели. И, следовательно, из обученной модели защищаемые данные могут в принципе извлекаться.

Сохраняя данные (а это обеспечивается известными механизмами защиты информации), мы не обязательно обеспечим сохранность знаний. Это уже упомянутая новая категория утечек – косвенные утечки, возникающие при сопоставлении множества ответов на ряд вопросов. Классические подходы к защите информации никак не противостоят косвенным утечкам, и эта особенность ИИ вырастает в проблему. Классический пример косвенной утечки – модельный пример посещения психолога. Пусть известно, что в коллективе из 15 человек трое (неизвестно, кто именно) посещают психолога. В команду добавился еще один

сотрудник, и посещающих психолога стало четверо. Представляется несложным выяснить, посещает ли новый член команды психолога.

Ясно, что это упрощенный и модельный пример. Настоящие варианты косвенных утечек значительно сложнее для анализа и значительно сильнее могут повлиять на уровень приватности человека.

Умение выявлять и блокировать возможность косвенных утечек – важнейшая задача, решение которой совершенно необходимо для обеспечения доверенности СИИ.

Вот так выглядит классификация составляющих доверие элементов СИИ – рис. 2.

Рис. 2. Чем обеспечивается доверие ИИ

В этой книге основное внимание уделяется косвенным утечкам – их выявлению и блокированию. В следующих книгах этой серии мы рассмотрим и другие составляющие – среду (Доверенная интеграционная платформа – ДИП) и данные (Система управления знаниями – СУЗ).

КАК ИСПОЛЬЗОВАТЬ ИИ ДЛЯ ОБЕСПЕЧЕНИЯ ДОВЕРИЯ, ИЛИ ГДЕ И КАК ИИ МОЖЕТ БЫТЬ ПОЛЕЗЕН ДЛЯ ПОВЫШЕНИЯ УРОВНЯ ДОВЕРИЯ

Традиционно среди задач защиты информации выделяют три основные – обеспечение доступности, целостности и конфиденциальности. Это реализуется механизмами идентификации и аутентификации, управлением доступом, криптографическими механизмами. На наш взгляд – этого недостаточно. Для полноты совершенно необходимо говорить не только о доступе к данным, но и о технологии обработки данных.

Совершенно очевидно, что, применяя одни и тоже операции к одним и тем же данным, можно получить различные результаты, даже незначительно изменив порядок операций. Мы считаем, что технологии должны пониматься как последовательность операций. А раз последовательность – то становятся применимы хорошо изученные понятия изоморфизма и гомоморфизма, и все вытекающие из этого математические конструкции. Мы определяем защищенные информационные технологии (ЗИТ) как технологии, обладающие свойством сохранять последовательность операций. Основы построения таких технологий есть [11], но основы нужно развивать.

Направление развития — это гибридные информационные инфраструктуры – среда которых состоит как из доверенных сегментов, так и из недоверенных. По сути, это единственная реальная инфраструктура. Мир как раз именно такой – в подлинности купюры, полученной на сдачу от незнакомого продавца, можно убедиться, попытавшись внести эти купюры на счет через банкомат, осуществляющий проверку подлинности. Здесь задача простая, так как банкомат верифицирован. А как быть, если информации о доверенности узлов инфраструктуры нет? Похоже, нужно собирать мнение пользователей о доверенности узла – пусть пользователи тоже не слишком доверенные, но обобщая мнение многих, можно выработать собственную оценку. Для больших систем, в которых действует множество независимых объектов, такой подход близок к известной [12] концепции мультиагентных систем (МАС), а множество узлов можно рассматривать как

множество капелек тумана – то есть свести задачу к задаче обеспечения безопасности в тумане – фогсекьюрити. Вот такое видится интересное применение методов ИИ для безопасности доступа и защищенности технологий в гибридных инфраструктурах.

Возвращаясь к базовой классификации, отметим, что наиболее часто говорят, что в системах искусственного интеллекта конфиденциальность будет обеспечиваться методами постквантовой криптографии. На мой взгляд, подход очень необычный с точки зрения привычной криптографии. В первую очередь – в постановке задачи. Речь идет о том, чтобы не точно восстановить передаваемые и зашифрованные символы, а примерно, так, чтобы результат мог быть «узнан» потребителем. То есть для каждого вида данных могут быть свои особые методы шифрования и расшифрованная – сохраняющие узнаваемость, а не точное соответствие. Для кошечек и собачек – разные. Для видео и аудио – разные. При моем относительно классическом образовании смириться с таким подходом нелегко. Остается наблюдать. Другое дело – идентификация и аутентификация. Здесь выявилось прямое, непосредственное применение методов ИИ для повышения качества этого этапа обеспечения защищенности. И самое главное – можно обеспечить доверенную идентификацию при использовании недовешенных клиентских терминалов (смартфонов).

Новый метод биометрической идентификации мы назвали рефлекторной биометрией. В следующей книге этой серии рефлекторной биометрии будет уделено много места, как и многим другим направлениям.

Здесь же в основном рассмотрим гибридные инфраструктуры. В целом же использование ИИ для улучшения доверенности показано на рис. 3.

Рис. 3. Как использовать ИИ для обеспечения доверия

Показанная на рисунках структура формирует представление о понятии доверия применительно к ИИ. По этой структуре мы планируем проводить исследования и публиковать их в серии книг, которую начинаем этой публикацией. Вначале мы рассмотрим инфраструктурные вопросы, а применение – уже после, тогда, когда будет оформлено базовое понимание доверия в тематике ИИ. Но как не структурируй изложение, совсем обойтись без применения ИИ совершенно невозможно. Ну и конечно – самое близкое любому человеку — это дистанционное банковское обслуживание (ДБО). А здесь самое болезненное — это фрод. Причем как фальсификация пользователя, так и неправомерное использование аккаунта. Традиционное трактование кредитными организациями (КО) положений 115-ФЗ[5] — это «закручивание гаек». Конечно, это крайне плохо влияет на конституционный «источник власти» — народ, пользующийся услугами банков. А ведь есть и другой подход — повысить точность идентификации клиента, и вот это вполне можно сделать методами ИИ. И об этом тоже есть раздел в этой книге.

[5] Федеральный закон от 7 августа 2001 г. N 115-ФЗ «О противодействии легализации (отмыванию) доходов, полученных преступным путем, и финансированию терроризма»

И наконец, представляю собственно книгу:

1. Предисловие – ваш покорный слуга.
2. Глава 1. Классификация аутентифицирующих признаков – С. В. Конявская-Счастная.
3. Глава 2. Оценка изменения доверенности наборов данных при их комплексировании и перспективы оценки доверия к отчуждённым обученным моделям – А. Ю. Ситников.
4. Глава 3. Оценка параметров систем слепой обработки данных, блокирующих косвенные утечки – П. А. Галманов.
5. Глава 4. О доверенности в гибридных системах искусственного интеллекта – С. С. Буянов.
6. Глава 5. Расширение пространства признаков в ИНС для задач антифрода в ДБО – С. Г. Ищанова.

Глава 1. КЛАССИФИКАЦИЯ АУТЕНТИФИЦИРУЮЩИХ ПРИЗНАКОВ

С. В. Конявская-Счастная

1.1. ЗАЧЕМ НУЖНО КЛАССИФИЦИРОВАТЬ АУТЕНТИФИЦИРУЮЩИЕ ПРИЗНАКИ

Идентификация, аутентификация и авторизация – это те функции защиты информации, с которыми сталкивается без преувеличения каждый. Хотя бы для того, чтобы отключить их на своем телефоне, планшете или компьютере.

Как правило, когда называются эти функции, речь идет об идентификации, аутентификации и авторизации *пользователя*. Однако это «не само собой разумеется». Идентифицированы и аутентифицированы защитными механизмами должны быть и объекты вычислительной среды, ведь до того, как объект идентифицирован и аутентифицирован, невозможно проверить его целостность [13]. И авторизация тоже осуществляется не только в отношении пользователя, ведь именно авторизация производится, когда подсистема разграничения доступа дает определенному процессу возможность получить доступ к тому или иному объекту.

Иными словами, идентификация, аутентификация и авторизация пронизывают так или иначе все функции защиты информации, а не только те, в которых пользователь непосредственно взаимодействует с системой.

Тема этой главы – классификация аутентифицирующих признаков – сама по себе нуждается в комментарии, поскольку классификация не является самоценной вещью – далеко не все и не в первую очередь нужно классифицировать. Более того, некоторая

принятая классификация аутентифицирующих признаков есть – это классификация по «факторам аутентификации».

Таким образом, начать необходимо с вопросов, зачем вообще классифицировать аутентифицирующие признаки, и почему не использовать классификацию по факторам.

В первую очередь, уточним, почему именно аутентифицирующие, а не идентифицирующие признаки представляется необходимым классифицировать. И то, и другое – некоторые отличительные признаки объекта. Признак может быть идентификационным или аутентификационным в зависимости от того, что сейчас выполняется – идентификация или аутентификация, в каком смысле признак предъявляется[1].

Для связности рассуждения позволю себе проговорить общеизвестные положения.

Идентификация (Identification – отождествление) – это отнесение объекта к определенной категории. При этом, заметим, категория, к которой мы относим объект, может представлять собой как множество (вижу нечто, идентифицирую это как сороку (какую-то, а не конкретную сороку)), так и единицу (вижу собаку, идентифицирую ее как свою собаку Жучку).

Идентифицировать пользователя – это определить, кто из пользователей перед нами. Идентифицировать компьютер, программу – аналогично – выяснить, какой это именно из компьютеров, какая именно из программ.

Признак, по которому идентифицируется объект – идентифицирующий признак (содержание), форма, в, которой этот признак предъявляется – идентифицирующие данные[2], или *идентификатор*.

Я идентифицирую сороку или Жучку по внешнему виду, это их идентификатор. Из двух одинаковых собак (предположим, что возможны одинаковые собаки) я могу идентифицировать свою, например, по ошейнику, тогда идентификатор – ошейник. Компьютер – по серийному номеру материнской платы, ПО – по

---

[1] Это не означает, что один и тот же признак будет (или должен быть) непременно принят в любом качестве. Но это совершенно другой вопрос.

[2] Заметим, что формой может быть и процесс, а не только данные, но для текущего рассуждения это не принципиально.

контрольной сумме кода (это не единственно возможные идентификаторы, а просто примеры).

Однако, обратим внимание на довольно очевидную вещь – если ошейник, который служит идентификатором моей собаки, на ней не надет, то он мне ничем не поможет (аналогично и, например, записанный на листочке UID). А процесс соотнесение ошейника с собакой – это аутентификация.

Строго говоря, если не производится аутентификация, то нельзя сказать, что идентификация в полной мере состоялась, потому что пока мы не обнаружили каких-то подтверждений тому, что это именно он (пользователь, компьютер, процесс, сорока), мы имеем лишь предположение. Подтверждение идентификации – привязка объекта к его идентификатору – и есть аутентификация.

Аутентификация (authenticity – подлинность) – это подтверждение связи между субъектом и признаком (идентификатором).

Ошейник, серийный номер, UID – могут быть и аутентифицирующими признаками.

Что такое нож – кухонная утварь или орудие преступления? Ответ зависит от того, что им в данный момент делают (или сделали).

Если я труп и рядом со мной обнаружен мой паспорт, то он (паспорт) – идентификатор. Если я пришла в присутственное место и представилась, а подтвердила свою самопрезентацию демонстрацией паспорта на такие же ФИО, то паспорт носитель аутентифицирующих данных. Если мы ищем устройство с серийным номером 1765390, то номер – идентификатор, если же мы проверяем серийный номер устройства, чтобы дать или не дать разрешение на его использование – то это аутентифицирующие данные (а устройство – их носитель).

Пример с паспортом выбран не случайно и не по ошибке, а потому, что он прекрасно иллюстрирует существенный дефект классификации аутентифицирующих признаков по «факторам». То, что не укладывается в «знать», «владеть» и т. д., в случае, если мы принимаем эту классификацию – не является аутентифицирующими признаками. Как правило, такие данные называют «свидетельствами». Так, паспорт – это официальное свидетельство.

Действительно, в парадигме «знать-владеть-биометрия» классификация паспорта как того, с помощью чего человек аутентифицируется, затруднительна хотя бы потому, что некоторые из паспортов – биометрические, а некоторые – нет. А главное, в отношении паспорта каким-то сомнительным представляется установление связи «владение». Вместе с тем, кроме этого затруднения, нет никаких причин отказывать паспорту в том, что он является носителем аутентифицирующих данных.

Но свидетельства согласно стандартам должны применяться только в процессах идентификации, а не аутентификации, а значит, паспорт не может применяться для аутентификации, а может применяться только для идентификации[3]. Хотя наблюдение над практикой аутентификации показывает обратное: как правило мы показываем паспорт именно для того, чтобы подтвердить декларацию «я такой-то».

Очевидно, что, если классификация вызывает необходимость необоснованного усложнения системы, она, скорее всего неверна, или, как минимум, нецелесообразна.

Признаки, которые могут быть и идентифицирующими, и аутентифицирующими, целесообразно классифицировать именно как аутентифицирующие потому, что именно аутентификация дает основания для принятия решений (для авторизации, если аутентифицируется субъект, какой-то действующий агент, которому могут быть разрешены или не разрешены какие-то действия).

Однако это не дает ответа на вопрос о том, для чего вообще нужно эти признаки классифицировать. Для этого еще пара слов об общеизвестных вещах.

Классификация – это разделение класса на категории по некоторому категориальному признаку. Это недвусмысленно свидетельствует, что классификаций объектов одного класса может быть несколько по *разным* категориальным признакам. Кроме чисто теоретических задач исследования с целью лучшего понимания, группировка по категориальному признаку полезна для того, чтобы выбирать объект с нужными характеристиками. И в этом смысле,

[3] Этот вывод не является вымыслом автора, он получен от А. Г. Сабанова, который является в нашей стране апологетом теории факторов, в личной беседе о том, к какому фактору аутентификации относить паспорт.

естественно, полезна та классификация, которая группирует объекты по тому признаку, который существенен для задачи, для которой необходимо сделать выбор.

Породы кошек можно классифицировать по средней температуре тела, и эта классификация вряд ли будет полезна для того, чтобы выбрать себе питомца, но может быть полезна при проверке гипотезы о том, что «кошка с хладнокровным характером» – это не совсем метафора.

С классификациями аутентифицирующих данных, в принципе, так же. Должна быть какая-то задача, которой отвечает классификация, иначе она нецелесообразна. Подробный анализ понятия «фактор аутентификации» и классификации на основании «факторов» приведен в [14 и др.], здесь остановимся только на целесообразности этой классификации.

1.2. КЛАССИФИКАЦИЯ НА ОСНОВАНИИ «ФАКТОРОВ АУТЕНТИФИКАЦИИ»

Согласно определению в ГОСТ и ряде научных работ [например, 15], фактор аутентификации – это «вид (форма) существования аутентификационной информации, предъявляемой субъектом доступа или объектом доступа при аутентификации»[4].

Классификация информации по форме ее существования – крайне продуктивна.

По системе определений, сформулированной в [6. С. 10–33] А. А. Стрельцовым, информация существует *в форме сведений и сообщений*, при этом и та, и другая форма существования информации – определены и четко отделены одна от другой. По системе определений В. А. Конявского [4. С. 179–191], в цифровой среде следует говорить не об информации как таковой, а о ее отображении (если провести параллель с системой определений Стрельцова, то в ней сведения отображаются в сообщениях). Отображается информация или в статической форме – *в форме*

[4] ГОСТ Р 58833-2020 «Идентификация и аутентификация. Общие положения». С. 7.

данных (чисел, представляющих собой упорядоченное множество символов) или *в форме процессов* – динамической форме.

В обеих системах дефиниций понятна природа определяемых феноменов, понятны их взаимосвязи и следствия из них – «форма существования» информации в виде сведений и сообщений позволяет выделить и определить предмет правоотношений, форма существования в виде данных и процессов – построить модель электронного документа и его защиты.

Что позволяет прояснить «существование аутентифицирующей информации в форме факторов аутентификации» – установить невозможно.

«Расположение в пространстве» – это не форма и не вид существования геолокационных данных, даже метафорически вряд ли можно сказать, что геолокационные данные существуют в форме (или в виде) расположения в пространстве. Это высказывание просто не имеет никакого смысла. То же самое будет, если попытаться подставить в эту конструкцию другие факторы: пароль существует в форме знания, ТМ-идентификатор существует в форме владения и так далее.

Скорее можно говорить о разделении всех возможных для применения в процессе аутентификации признаков на классы по тому, *чем* в реальном физическом мире является *их источник,* откуда они берутся – из знания, из свойств некоторого предмета, из физиологической особенности субъекта, из его расположения в пространстве или образа его действия.

Это не характеристика самих признаков, и это принципиально, ведь компьютерная система не может работать ни с предметом, ни со знанием, ни с расположением, ни с чем другим из реального мира, она может работать только с цифровыми данными и процессами, причем через строго определенные интерфейсы (допустим, если пароль надо ввести на клавиатуре, то бесполезно его называть вслух или показывать на бумажке – хотя пароль будет и верный, пока он не будет введен с клавиатуры, аутентификация успешно не завершится).

Признаки (и в целом информацию) можно, вероятно, без ошибок классифицировать по их источнику. Но целесообразно ли это для задач защиты информации?

Классификационный признак должен быть связан с целью, для которой проводится классификация, он должен быть объясним.

Не удается обнаружить явных причин тому, что классифицировать аутентифицирующие признаки целесообразно именно *по источнику*. Более того, если на основании «факторов» аутентификации строятся дальнейшие рассуждения, то оценивается не то, откуда аутентифицирующая информация берется, а признаки, характеризующие степень ее связанности с субъектом. Чаще всего – *насколько необходимо* вступать в контакт с субъектом, и *насколько тесным* должен контакт для того, чтобы злоумышленник смог получить эту информацию в свое распоряжение. То есть разделяются феномены на группы по одному признаку, а используются далее – другие признаки, свойственные этим же группам (возможно, эти группы признаков связаны, но эта связь не описана).

Именно поэтому представляется верным не пытаться исправлять ошибки в классификации по «факторам аутентификации», а отказаться от нее совсем как от нецелесообразной.

1.3. КЛАССИФИКАЦИЯ БЕЗ ФАКТОРОВ, НО С КОНСТРУКТИВНОЙ ЦЕЛЬЮ

Прикладная цель, для достижения которой может понадобиться классификация аутентифицирующих признаков – реализовать в системе идентификацию/аутентификацию *субъекта* адекватно самой *системе*.

Отсюда вытекает исследовательская цель – классифицировать аутентифицирующие признаки так, чтобы это было информативно для выбора при конкретном проектировании. А это, в свою очередь, значит, что категориальный признак должен быть связан с *субъектом* и с *системой*, а классы должны накладываться на известные характеристики систем, значимые для защиты информации.

Если категориальный признак не будет связан с субъектом и с системой, то классификация не будет релевантна цели, а если классы не будут органично накладываться на известные характеристики

систем, то классификацию будет невозможно (или, как минимум, сложно) применять на практике.

Применение классификации на практике заключается в том, чтобы оценить, не что может или не может быть аутентифицирующим признаком[5], а какие аутентифицирующие признаки подходят для той или иной конкретной системы при каких сопутствующих условиях, которые потребуется создать и поддерживать для доверия результатам аутентификации.

1.4.1. Предпосылки классификации аутентифицирующих признаков

1.4.1.1. Предмет классификации

В первую очередь, необходимо зафиксировать, что в информационную систему мы в любом случае представляем только *данные или процессы*, а не знания, не собственность или что-то еще из аналогового или духовного мира. Поэтому речь должна идти о *данных или процессах*, характеризуемых какими-то *признаками* или наборами признаков.

В абсолютном большинстве случаев аутентифицирующие признаки представлены в форме данных, а не процессов, поэтому для упрощения конструкций будем называть их данными, имея в виду в то же время, это на самом деле это может быть и процесс[6].

---

[5] Строго говоря, это может быть практически что-угодно, например, рост больше или равный 120 см может подтверждать, что субъект входит в множество тех, кому уже можно кататься на определенной карусели. Заметим, что для авторизации в этом случае понадобится еще оплата катания, успешной аутентификации недостаточно.

[6] В настоящее время известен только один способ аутентификации, использующий аутентифицирующий признак в форме процесса – это интерактивная рефлекторная биометрия [16, 17 и др.]. Аутентифицирующий признак в этом случае – реакция рефлекторной дуги, то есть процесс развития реакции на стимул во времени. Важно не путать этот случай с биометрической идентификацией, например, по голосу – голос тоже имеет протяженность во времени, но он передается, обрабатывается и сравнивается с эталоном как набор данных, в котором выделяются и постоянные характеристики. В случае же с интерактивной реакцией на уникальный стимул предметом анализа является процесс и его характеристики.

Способы аутентификации и *аутентифицирующие данные* – это разные вещи, и они не должны смешиваться. Является общим местом, что важно не только и не столько то, какие данные используются для аутентификации, сколько то, *каким образом реализован механизм аутентификации* в системе. Например, реализация парольной защиты может быть признана слабой в двух совершенно разных случаях:

1) если она не включает в себя проверку на слабые пароли (или какие-то механизмы блокировки угроз, связанных с их применением),

2) если слабые пароли в ней исключены, но сильные хорошие пароли – хранятся и передаются в открытом виде и в форме, делающей их доступными для нелегального применения.

То есть так или иначе, слабость пароля как аутентифицирующего признака и слабость реализации – это две разные слабости.

Ниже речь пойдет не о способах и не о механизмах аутентификации, а именно об аутентифицирующих данных и их признаках.

1.4.1.2. Классификационные признаки

У аутентифицирующих данных можно выделять множество особенностей, и значительная часть описывающих эти особенности признаков будет важна с точки зрения защиты информации. Например, может требоваться точное совпадение сравниваемых данных с эталоном или совпадение в рамках некоторого диапазона: пример первого – пароль, второго – биометрический эталон.

Данные могут быть постоянными / условно постоянными / изменяющимися.

Постоянные – статические биометрические данные (такие, как, например, папиллярный узор), они никогда не меняются. Однако, постоянным является так же и пароль, например, так как один и тот же пароль никогда не станет другим, его можно только поменять на другой – тоже постоянный.

Условно постоянные – это данные, которые могут меняться, но не целенаправленно, а из-за каких-либо неконтролируемых обстоятельств. Таким признаком, например, является сосудистое русло ладони. Его принято считать неизменяемым биологическим

признаком, но уже доказано, что приблизительно раз в полгода следует обновлять их эталон, так как рисунок сосудистого русла все же меняется.

Изменяющиеся данные – это данные, которые изменяются «сами по себе», независимо от воли человека, например, динамические биометрические данные[7] (они каждый раз разные, даже если будет предъявлен тот же самый стимул, и какие бы усилия ни приложил человек).

Другой признак – характеризующий не столько данные, сколько их применение – это кратность. Данные, использующиеся для аутентификации могут представлять собой «кортеж», содержащий кроме собственно представляемых в систему данных (например, пароля), еще данные о том, сколько раз можно использовать эти данные.

Эти – и многие другие – признаки важны, и они характеризуют именно те особенности аутентифицирующих данных, которые определяют их принципиальную применимость и ограничивают способы их применения в механизмах аутентификации. Но для классификации нужны такие признаки, которые характеризуют одновременно *все* возможные данные, причем характеризуют информативно, а не формально (так, наличие признака «частота» не дает оснований классифицировать звуки и цвета по длине волны в одну сплошную классификацию).

Это первое принципиальное требование: категориальный признак должен быть релевантен всем объектам класса.

Признаки, по которым данные будут классифицированы, должны позволять строить на основе этой классификации рассуждения о защищенности механизмов аутентификации, использующих эти признаки аутентифицирующих данных. Значит, они *должны быть связаны с ключевыми элементами процесса аутентификации*.

Рассмотрим, что это за элементы. Аутентификация – понятие относительное. Что-то одно можно аутентифицировать относительно чего-то другого, нельзя аутентифицировать что бы то ни было само по себе. Более того, даже объективно существующая

[7] Например, траектория слежения взглядом за стимулом.

связь между идентификатором и субъектом может в одном случае обеспечить успешную аутентификацию, а в другом случае решительно никак не повлиять на результат аутентификации. Например, совершенно бессмысленно показывать паспорт вместо предъявления карты СКУД или прикладывать палец к считывателю рисунка радужной оболочки глаза. И даже если технически предъявление возможно – это еще ничего не гарантирует, попробуйте предъявить не тот палец, который был зарегистрирован – несмотря на то, что он определенно ваш, аутентификация закончится неуспешно. Важно, что в процессе аутентификации производится подтверждение объективной связи субъекта *именно с тем* идентификатором, который является его признаком для второй участвующей в процессе аутентификации стороны.

В процессе аутентификации участвует не менее двух сторон. Если это стороны А и Б, то:

1. А можно аутентифицировать относительно Б;
2. Б можно аутентифицировать относительно А;
3. Стороны А и Б можно аутентифицировать одну относительно другой – взаимно аутентифицировать.

Назовем участвующие в аутентификации стороны субъектом и объектом аутентификации, так как даже при *взаимной* аутентификации все равно каждый конкретный случай аутентификации какая-то сторона инициирует, и в этом смысле разделение *субъекта* и *объекта* вполне уместно и, как кажется, не порождает никаких проблем.

Важно учитывать разницу между объектом доступа и объектом аутентификации. Когда администратор включает сервер, этот сервер может не быть объектом доступа, так как никаких действий на нем администратор не выполняет, он может даже не иметь на нем учетной записи пользователя ОС (а быть, например, только пользователем СДЗ). Однако объектом аутентификации сервер определенно будет. В зависимости от реализации подсистемы защиты информации объектом аутентификации при процедуре включения сервера может быть не сервер, а СДЗ на сервере, однако, общей картины это не меняет, так как объектом *доступа* он все равно не будет: пользователю не станет доступно выполнение никаких операций ни с сервером, ни с СДЗ.

Объектом доступа может быть некоторый целевой ресурс, а не информационная система целиком или даже СВТ. Аутентификацию в идентификации/аутентификации банкомата клиент банка проходит для того, чтобы получить доступ к своему счету, а не к банкомату. И *объектом аутентификации* будет банкомат как информационно-вычислительный ресурс, предоставляющий доступ к целевому *объекту доступа*.

Условимся, что

Объект доступа – объект, на выполнение операций над которым претендует субъект доступа. Успешный результат процедуры доступа – возможность выполнения операций субъектом над объектом (например, чтения, записи и так далее). Условием для успешности процедуры доступа может быть *авторизация*.

Объект аутентификации – объект, относительно которого аутентифицируется субъект. Успешный результат процедуры аутентификации – подтверждение связи пользователя и предъявленных им идентификационных данных пользователя.

Это разделение существует и в аналоговом мире. Мы аутентифицируемся в ЗАГСе по паспорту, но чтобы получить доступ к услуге (регистрация вступления в брак или расторжения брака, например), нужно оплатить гос.пошлину, то есть успешной аутентификации для доступа не всегда достаточно.

А в целом ряде других случаев – и в цифровом, и в аналоговом мире, наоборот, для доступа к чему-либо не понадобится вовсе никакая аутентификация.

Говоря о типах аутентифицирующих данных, останемся в рамках процесса аутентификации.

Итак, аутентифицирующие данные используются для того, чтобы подтвердить для объекта аутентификации связь между субъектом аутентификации и его идентификатором.

Процесс этот состоит из предъявления данных субъектом и проверки данных объектом (см. рис. 1.1).

Рис. 1.1. Предъявление субъектом аутентификации аутентифицирующих данных для проверки объектом аутентификации

Различные аутентифицирующие данные могут быть охарактеризованы с точки зрения и других процессов – порождения этих данных, их хранения, обработки, передачи. В этих случаях могут рассматриваться такие параметры, как носитель и канал передачи данных объекту[8], источник и характер порождения данных, организация хранения и другие особенности, однако для того чтобы было возможно рассматривать эти аспекты системно, целесообразно классифицировать аутентифицирующие данные на самом высоком уровне – с точки зрения их *связанности* с субъектом и объектом аутентификации.

Это второе принципиальное требование – связанность категориального признака с субъектом и объектом аутентификации.

1.4.1.2.1. Связанность с объектом аутентификации

Не зависеть от объекта аутентификации совсем аутентифицирующие данные не могут, так как иначе они не смогут сыграть целевую для них роль – доказательства. Объект должен располагать какими-то данными для того, чтобы принять решение о корректности представленного подтверждения.

Для того чтобы проверить предъявленные данные, объект аутентификации может использовать всего два варианта:

[8] Канал передачи данных от субъекта в подсистему аутентификации объекта зачастую определяется применяемым носителем данных (USB-токен не передаст данные через сканер отпечатка пальца и наоборот), поэтому их целесообразно рассматривать в связке как интерфейс передачи данных от субъекта объекту.

- именно те данные, которые и предъявит субъект,
- какие-то другие данные, с помощью которых можно проверить предъявленные данные.

Если данные *именно те*, которые субъект будет предъявлять, то проверяться будет точное совпадение, это простая проверка путем сравнения.

Если это какие-то *косвенные данные*, позволяющими определить корректность данных, не располагая ими «в лоб», то это будет функциональная проверка. Например, так производится проверка ключа подписи атрибутного сертификата – ключом подписи субъекта объект не располагает, но располагает возможностью проверить его корректность.

Соответственно, по этому признаку аутентифицирующие данные можно разделить на те, что требуют *простой* проверки (рис. 1.2), и те, что требуют проверки *функциональной* (рис. 1.3). В первом случае производится сравнение предъявленных данных с эталонными, во втором – вычисляется некоторая функция.

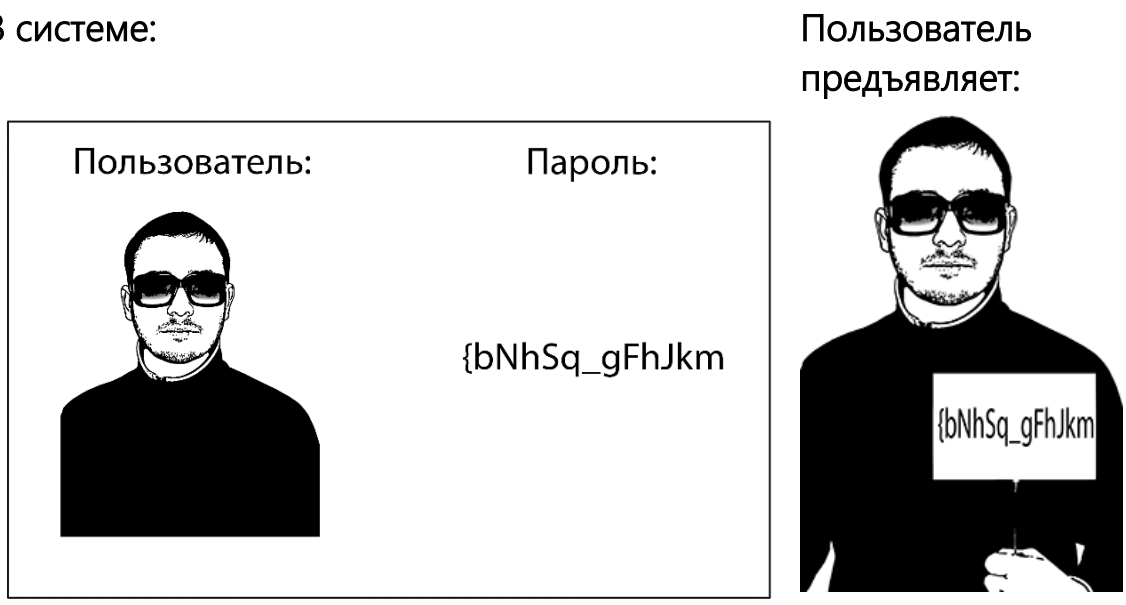


Рис. 1.2. В системе пользователю сопоставлен назначенный ему пароль, пользователь при аутентификации должен предъявить тот же самый пароль

В системе: Пользователь предъявляет:

Рис. 1.3. В системе хранятся данные и реализованы механизмы для проверки предъявленных аутентифицирующих данных (в данном случае данные – это половинка разрезанной банкноты, а механизм – сложить имеющуюся половинку с предъявленной и оценить, от той же банкноты предъявленная половинка или нет). Пользователь предъявляет другие данные (другую половинку банкноты). В системе производится оценка корректности данных (сложилась ли банкнота из предъявленной и имеющейся половинок).

Проиллюстрируем свойства и различия таких данных таблицей 1.1.

Таблица 1.1. Свойства сравниваемых и проверяемых аутентифицирующих данных

	Сравниваемые	Проверяемые
Хранение аутентифицирующих данных на стороне объекта	да	нет
Что требуется для успешной аутентификации	совпадение	корректность
Контрольная процедура	сравнение	проверка

Предъявляемые данные и данные, используемые контрольной процедурой	одинаковые	разные
Зависимость от объекта аутентификации	прямая	косвенная
Примеры в аналоговом мире	Пароль, проверка собственноручной подписи	Узнавание по наличию подходящей ответной части (разделенный на части предмет, ключ от замка, способность вынуть меч из камня)
Примеры в информационном взаимодействии, в котором субъект – человек	Пароль, PIN- и PUK-код	Атрибутные сертификаты, криптографические ключи в процедуре «рукопожатия»
Примеры в информационном взаимодействии, в котором субъект – техническое средство или процесс	Имя компьютера	Контрольные суммы

1.4.1.2.2. Связанность с субъектом аутентификации

Охарактеризовать связанность аутентифицирующих данных с субъектом аутентификации можно по признаку *отделимости* от него.

Данные или присущи, имманентны субъекту – и тогда они от него неотделимы, или ассоциированы с ним (упрощенно можно сказать, что они ему выданы, назначены). Ассоциированные данные

от субъекта отделимы. Примеры аутентифицирующих данных этого типа визуализированы на рисунке 1.4 и 1.5.

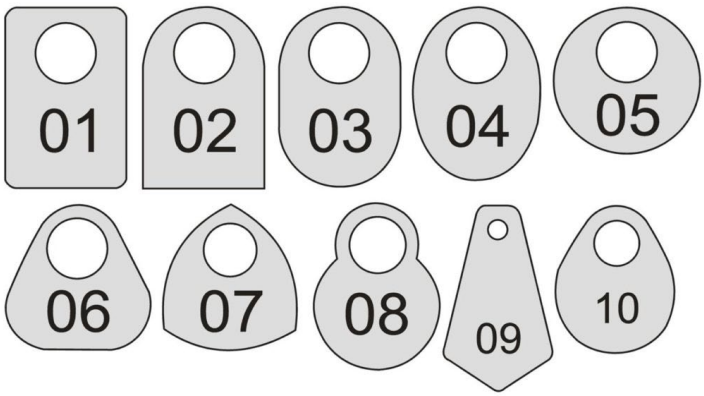


Рис. 1.4. Номерки в гардеробе не имеют никакого отношения ни к человеку, ни к его одежде, однако обеспечивают некоторую вероятность получить свою одежду назад.

Рис. 1.5. Комикс о том, что пароль должен быть сложным для подбора компьютером, а не для запоминания человеком (хотя как правило получается наоборот) с сайта https://xkcd.com/.

Не следует смешивать понятия неотделимости и неизвлекаемости. Неизвлекаемость – характеристика, означающая, что тот или иной объект нельзя штатными средствами *извлечь* из чего-либо (например, ключ из носителя) и, например, скопировать. С неотделимостью дело обстоит иначе – доминантой является не невозможность *доступа* к той или иной сущности и манипуляций с нею (аутентифицирующие данные должны быть непременно доступны для предъявления!), а скорее то, что субъект является целым, штатно функционирующим, или, проще говоря – является самим собой – только тогда, когда этот признак при нем, не отделен

от него. Строго говоря, такие данные как папиллярный узор (или иная биометрия) являются в такой же степени условно неотделимыми от субъекта, как условно некопируемым является аппаратный идентификатор. При желании и настойчивости от человека можно многое отделить и извлечь – но он определенно потеряет свою целостность и будет уже не тем, что прежде. Ассоциированные же данные можно отделить от человека вполне безболезненно (прошу прощения за неуместный каламбур).

Итак, к неизвлекаемым данным невозможен прямой доступ, а к неотделимым доступ может быть вполне возможен, но они являются частью субъекта, без которых он теряет целостность.

Определенная сложность с признаком неотделимости данных заключается в том, что этот признак может характеризовать только те данные, которые *в непреобразованном виде* в систему (и подсистему аутентификации, в частности) передаваться *не могут*. От чего бы неотделимыми ни были эти данные (от человека ли, от устройства ли), они неотделимы от чего-то, что находится за *рамками* системы, значит в систему *никогда не попадают*. Однако это влияет только на описание процесса порождения данных и не создает никакой проблемы для их классификации.

В то же время сам по себе признак неотделимости данных очень важен, это очевидно хотя бы по иррациональному доверию людей к биометрии. На рисунке 1.6 в качестве примера приведено изображение глаза человека. Радужная оболочка глаза – это неотделимый от человека признак.

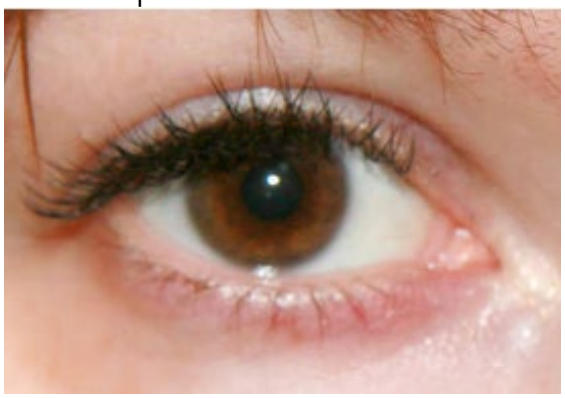


Рис. 1.6. Глаз человека

Необходимо ответить только на один существенный вопрос – отделимость/неотделимость *от чего* мы будем считать классифицирующим признаком. Неотделимость биологических

характеристик от человека – это одно, а неотделимость, допустим, ключа от носителя – это несколько иное, хотя бы потому, что носитель от человека вполне легко отделяется.

Ответ получается довольно очевидным – мы характеризуем зависимость от *субъекта аутентификации*, стало быть отделимость или неотделимость мы должны установить именно от него. Если субъект аутентификации (напомню, это тот, кто аутентифицируется) – человек, то нас интересует отделимость от него, если аутентифицируется носитель каких-то данных или любое другое устройство (или программа) – то отделимость от носителя / устройства / программы соответственно.

Проиллюстрируем свойства и различия ассоциированных и неотделимых данных таблицей 1.2.

Таблица 1.2. Свойства ассоциированных и неотделимых аутентифицирующих данных

	Ассоциированные	Неотделимые
Связь с субъектом	нет	есть
Возможность передачи другому субъекту	+	-
Возможность утраты	+	- (с оговорками, которые в данном контексте значения не имеют)
Возможность замены в случае компрометации	+	-
Примеры в аналоговом мире	Номерок в гардеробе, билет на концерт, ключ от замка, разделенный на части предмет	Особые приметы, способность вынуть меч из камня
Примеры в информационном взаимодействии,	Пароль, PIN- и PUK-код, аппаратный идентификатор,	Биометрические характеристики различного рода

где субъект – человек	сертификат ключа подписи	
Примеры в информационном взаимодействии, где субъект техническое средство или процесс	Имя компьютера, Электронная подпись	Серийный номер устройства, фрагмент кода программы

Важно также то, один или много субъектов и одна или много систем могут использовать признак.

Поэтому еще одна группа признаков аутентифицирующих данных – их *тиражируемость*. На первый взгляд этот признак близок к ассоциированности/неотделимости до полного смешения – операция копирования является базовой в цифровой среде, значит, любые данные, попадающие в цифровую среду, могут быть скопированы сколько угодно раз.

Разумеется, такой подход непродуктивен. Тиражируемостью предлагается называть возможность *параллельного использования* данных разными субъектами или разными объектами аутентификации.

1.4.1.2.3. Тиражируемость между субъектами аутентификации

Так, аппаратный идентификатор мы условно считаем некопируемым, то есть одновременно он может находиться только у одного пользователя. Поэтому, хотя он может быть передан легальным субъектом постороннему лицу, он все же является *нетиражируемым*, так как одновременно им может воспользоваться один субъект.

Пароль же может быть назначен любому числу субъектов (именно так обычно происходит с паролями user, student, admin и аналогичными) – поэтому он *тиражируемый*.

Эти примеры иллюстрируют тиражируемость данных с точки зрения субъектов аутентификации: возможность использования данных разными субъектами в одно время.

1.4.1.2.4. Тиражируемость между объектами аутентификации

С точки зрения объекта аутентификации тиражируемость аутентифицирующих данных тоже может быть значимой. От того, тиражируемы ли данные между объектами, зависит то, в одном или нескольких объектах аутентификации этому субъекту могут быть сопоставлены эти данные.

Сразу приходит в голову старое, как мир, но бесконечно нарушаемое правило – не регистрировать в разных ресурсах один и тот же пароль. Однако, когда в разных ресурсах регистрируется одно и то же лицо (в прямом физическом смысле – face), принципиально – это то же самое. Причем ситуация даже хуже, чем с паролем – пароль поменять существенно легче, чем лицо. В то же время именно одно и то же лицо вы регистрируете в метро и в Сбере. Несмотря на то, что это, конечно, очень ответственные операторы, кто может быть до конца уверен, какая еще система располагает этими данными?

Этот пример показывает крайне важное обстоятельство: *тиражируемыми между объектами* аутентификации могут быть данные, *нетиражируемые между субъектами*, и даже *неотделимые от субъекта* аутентификации.

Надо заметить, что тиражируемые данные – это данные, которые *могут быть* растиражированы, а не обязательно заведомо растиражированные. Но с точки зрения защиты информации необходимо исходить из того, что то, что может быть растиражировано, будет растиражировано, если это хоть кому-нибудь нужно. Значит, для всех тиражируемых данных в системах должны приниматься какие-то меры, препятствующие их тиражированию.

Нетиражируемыми между объектами аутентификации могут быть, по-видимому, только динамические аутентифицирующие данные (те, что представляют собой интерактивный ответ на уникальный стимул, и, соответственно, являются уникальными). В отношении любых статических данных должны применяться какие-то механизмы, ограничивающие возможность использования одних и тех же данных в разных системах. Так, ограничивая пользователя в праве зарегистрировать в той или иной системе свой личный

идентификатор (или идентификатор из другой системы) – мы создаем ему определенные неудобства, но несколько снижаем вероятность компрометации аутентифицирующих данных. Но насколько это снижение существенно – нужно анализировать на основании моделей угроз и нарушителя.

Может ли пользователь зарегистрироваться со своим идентификатором и паролем, или только получить специальные идентификатор и пароль именно для этой системы, будет зависеть от реализации системы аутентификации[9].

1.4.2. Субъектно-объектная классификация аутентифицирующих данных

Итак, получается, что по отношению к объекту аутентификации данные могут быть

1) сравниваемые / функционально проверяемые,
2) тиражирумые между объектами / нетиражируемые между объектами.

Наложением этих признаков, получаем следующие варианты пересечений:

1) сравниваемые тиражируемые,
2) проверяемые тиражируемые,
3) сравниваемые нетиражируемые,
4) проверяемые нетиражируемые.

Довольно очевидно, что «проверяемых тиражируемых между объектами» данных быть не может, так как если в системе не хранятся данные (проверяемые данные – значит, в системе хранится

[9] От реализации системы аутентификации будет зависеть еще много параметров. Например, постоянными или временными будут данные, ассоциированные с субъектом:
- система может использовать заводской номер идентификатора (он постоянный) или вычисление от каких-либо данных, записанное в память идентификатора, или даже им осуществляемое (и это будут временные данные);
- система может требовать или не требовать смены пароля по регламенту.
Однако в любом случае и пароль, и аппаратный идентификатор останутся ассоциированными и тиражируемыми между объектами аутентификации данными.

функция для проверки, а не сами данные), то они и не могут быть растиражированы между системами. Система не растиражирует данные, которых у нее нет.

Остальные классы не пустые, для них находятся примеры:

1) пароль,
2) -
3) одноразовый пароль, талончик в очереди[10],
4) атрибутный сертификат, интерактивная биометрия.

По отношению к субъекту аутентификации получаются следующие пары признаков:

1) ассоциированные / неотделимые,
2) тиражируемые между субъектами / нетиражируемые между субъектами.

Наложением признаков получаем следующие классы:

1) ассоциированные тиражируемые,
2) неотделимые тиражируемые,
3) ассоциированные нетиражируемые,
4) неотделимые нетиражируемые.

Проверим на предмет пустых классов:

1) пароль, имя компьютера;

[10] Очевидно, что система не хранит список номеров талончиков, из которых выдает по одному, а формирует номера по порядку с учетом каких-то дополнительных входных данных (например, за какой государственной услугой пришел в многофункциональный центр посетитель, или к какому врачу в поликлинике); также очевидно, что талончик не связан с субъектом, а ему назначается; о зависимости от носителя тут тоже говорить некорректно, так как то, что на талончике напечатано, от носителя не зависит, оно полностью формируется системой. Можно возразить, что такой же номерок, но напечатанный на чем-то другом, не будет принят как подтверждение очереди, однако тут опять же связь с системой, а не с носителем, ибо талончики именно в таком как есть виде целиком выпускаются системой, можно сказать, являются ее продуктом. При этом доступ дается на основании сравнения данных талончика с номером, который высветился над соответствующим окошком. Суммируя все это, можно утверждать, что это ни что иное, как вариант реализации одноразового пароля. Важной особенностью этого случая является то, что подтвердить с помощью таких аутентифицирующих данных субъект может только одно – «бронирование» права на взаимодействие с системой.

2) биологические признаки, свойственные классу субъектов (такие как возраст, пол, цвет глаз или волос), ФИО, фрагмент кода программы;
3) аппаратные идентификаторы,
4) личные биометрические данные (как статические, так и динамические).

Здесь стоит лишний раз оговориться, что мы определяем аутентифицирующие данные, а не реализацию подсистем, поэтому возможность кражи данных *после предъявления их в систему* к тиражированию не относим.

Представить себе данные, которые были бы неотделимы от субъекта, но в то же время тиражируемы между субъектами – на первый взгляд, довольно сложно, и ситуация с этим классом кажется аналогичной «проверяемым тиражируемым». Но через субъект – техническое средство или программу – это сделать возможно. Фрагмент кода программы является от нее явно неотделимым, без него, если его отделить – программа станет другой. Однако тот же кусок кода может встречаться в других программах, и тоже являться их неотъемлемой частью. Имя человека ему имманентно, но как правило, есть еще довольно много людей с таким же самым.

Любопытно, что с именем компьютера дело обстоит иначе, чем с именем человека – человеку имя назначается раз и навсегда (кроме исключительных случаев), компьютер же может быть переименован сколь угодно часто, поэтому имя компьютера целесообразно считать ассоциированным, а не неотделимым признаком.

Это рассуждение делает довольно очевидным, что использовать такие данные как аутентифицирующие целесообразно только тогда, когда требуется аутентификация на уровне отношения субъекта к некоторому классу. В аналоговом мире такая задача иногда встречается (проход в женскую раздевалку (женский пол) или на аттракцион (рост не менее 120 см) и тому подобные), но в области технической защиты информации это маловероятно.

В целом формулирование этих свойств и сопоставление им фактически используемых на практике аутентифицирующих данных позволяет лучше понять, в каких случаях какие данные адекватны целям аутентификации, и принятия каких защитных мер требует

применение именно их в той или иной системе. Например, если модель угроз и модель нарушителя допускают, что субъект аутентификации может быть недобросовестным (например, в сговоре с нарушителем, или просто неблагонадежным в плане дисциплины), то использование для аутентификации данных, ассоциированных с субъектом, а в еще большей степени – тиражируемых между субъектами, должно сопровождаться комплексом мер, каким-то образом снижающих риск передачи другим лицам и/или тиражирования аутентифицирующих данных. Особенно это касается аутентификации при доступе к учитываемым или просто платным услугам – здесь передача и/или тиражирование открывает возможности личной наживы, что может быть сильным стимулом для нарушения политики безопасности.

Ассоциированные с субъектом аутентификации и тиражируемые между субъектами аутентификации данные должны считаться потенциально скомпрометированными во всех случаях, если субъект аутентификации признается потенциальным злоумышленником.

Разница между ассоциированными нетиражируемыми и тиражируемыми заключается только в том, что о компрометации тиражируемых данных субъект данных может и не знать. То есть строго говоря, ассоциированные данные следует считать потенциально скомпрометированными в любом случае.

Что же касается возможности тиражирования между объектами аутентификации – если важно, чтобы аутентифицирующие данные, использующиеся в той или иной системе, не использовались в других системах, то необходимо либо использовать данные, не тиражируемые между объектами аутентификации, либо разрабатывать какие-либо защитные меры, препятствующие тиражированию.

И наоборот, если субъект аутентификации имеет основания предполагать, что объект аутентификации передает аутентифицирующие данные в смежные системы или просто не чист на руку и, например, продает их, то ему целесообразно избегать предоставления таких данных, которые система может

тиражировать. Хотя, надо признать, пользователь не всегда может влиять на то, какие данные предоставлять, если взаимодействовать с системой он вынужден в любом случае.

Тиражируемые между объектами аутентификации аутентифицирующие данные, которые мы предоставляем системе, следует всегда считать уже потенциально скомпрометированными.

Классификация

Одновременно данные характеризуются какой-то зависимостью и от субъекта, и от объекта. Это можно визуализировать в виде таблицы (таблица 1.3) и попробовать сопоставить получившимся ячейкам фактически использующиеся данные. Так станет очевидно, каких классов быть не может, а далее будет возможно проанализировать, какие данные использовать решительно нецелесообразно.

Необходимо оговориться, что по данным, передаваемым объекту аутентификации как таковым, в отрыве от каких-либо сопутствующих условий или обстоятельств, определить, какого они типа – не представляется возможным. Как принято говорить – «это просто нули и единицы». Различия проступают наглядно тогда, когда мы смотрим на систему на уровне «аутентификаторов» или сценариев использования данных. Поэтому выделенные типы аутентифицирующих данных ниже иллюстрируются именно через эти два проявления: аутентификаторы и сценарии использования. Предпосылкой, делающей возможным такое сведение друг к другу разных понятий, является их детерминированность. Одному типу данных может соответствовать (и, как правило, соответствует) несколько типов аутентификаторов и сценариев аутентификации, но не наоборот. Аутентификаторы и сценарии использования не могут работать сразу с разными типами аутентифицирующих данных.

Таблица 1.3. Пересечение классов аутентифицирующих данных по отношению к объекту и к субъекту аутентификации: примеры данных

		Сравниваемые тиражируемые между О	Сравниваемые не тиражируемые между О	Проверяемые не тиражируемые между О
Асс. тир. между S	человек	**Пароль, контрольные вопросы**	**Одноразовый пароль, талончик в очереди, гостевая карта СКУД**	**нет**
	Тех. ср.	**Имя** файла, компьютера, пр.	**лицензионный ключ на ПО**, зависящий от данных лицензируемого ПО и системы, для которой оно лицензируется	**нет**
Асс. не тир. между S	человек	**Аппаратный идентификатор, номерок в гардеробе,**	**токен аутентификации (Token-Based Authentication)**	**Атрибутный сертификат, ЭП, OTP** (устройство, генерирующее одноразовые пароли)
	Тех. ср.	**Серийный номер наложенного СЗИ** (при условии, что S аутентификации является не СЗИ, а СВТ)	**лицензионный ключ на ПО**, зависящий от данных лицензируемого ПО и СВТ в рамках этой системы	**Криптографическое рукопожатие**
Неотд. тир.	человек	**ФИО**, биологические параметры, характеризующие класс, а не индивидуума (**цвет кожи, рост, вес**, пр.)	**нет**	**нет**
	Тех. ср.	**Фрагмент кода программы, сигнатура**	**нет**	**нет**
Неотд. не тир.	человек	**Статические биометрические данные**	**нет**	**Динамическая (интерактивная) биометрия**

	Тех. ср.	**заводской серийный номер устройства** (того, которое является S аут-ции, или неотделимого от него), **контрольные суммы**	нет	ЭП поставщика

Необходимо оговорить ячейки, в которых стоят «нет».

Так, неотделимые от субъекта и в то же время тиражируемые между субъектами данные могут быть только сравниваемыми с эталоном и тиражируемыми между объектами аутентификации. Это логично, так как это самый простой в реализации системы тип связи аутентифицирующих данных и объекта аутентификации, и просто не имеет никакого смысла реализовывать более сложные варианты для использования данных, позволяющих аутентифицировать не конкретного субъекта, а лишь класс, к которому он относится. Трудно придумать какую-то конструктивную цель такой аутентификации в информационной системе, поэтому данные этого типа предлагается считать нецелесообразными к применению в защите информации.

Ассоциированные с субъектом тиражируемые между субъектами данные, как выше уже упоминалось, логично в любом случае считать потенциально скомпрометированными, поэтому строить более сложную систему их проверки (функциональная проверка – более сложная, чем простое сравнение) – явно нецелесообразно.

Если данные неотделимы от субъекта, но их эталоны для прямого сравнения хранятся в системе, нет никаких причин к тому, чтобы их было невозможно поместить и в другую систему, поэтому примеров неотделимых от субъекта сравниваемых и нетиражируемых между объектами данных – нет.

Обращает на себя внимание тот факт, что все ячейки заполнены только в строке ассоциированных с субъектом нетиражируемых между субъектами данных. Именно этот тип аутентифицирующих данных фактически наиболее популярен, так как в условиях защищенной корпоративной системы позволяет без значительных

сложностей реализовать подсистему аутентификации, соответствующую всем распространенным моделям угроз.

И второе наблюдение, которое можно почерпнуть из полученной таблицы – единственный тип данных, которые нет причин заранее считать потенциально скомпрометированными без применения дополнительных защитных мер – это неотделимые от субъекта нетиражируемые между субъектами данные, функционально проверяемые объектом аутентификации. Для аутентификации пользователя (человека) – это динамические биометрические данные.

Примера данных, которые бы не попадали ни в одну из получившихся категорий, пока не обнаружено, что, безусловно не говорит само по себе о том, что классификация верна, однако усиливает ее правдоподобность.

Классы данных расположены по возрастанию сложности нелегального получения данных этих классов в свое распоряжение.

В то же время нельзя не заметить, что так же возрастает и сложность реализации подсистемы идентификации \ аутентификации, использующей данные разных типов.

Однако рост сложности реализации в некотором смысле компенсируется тем, что снижаются требования к обеспечению доверия тем или иным компонентам системы, которые могут в разных случаях быть легче или сложнее доступны для контроля. Иногда проще сделать доверенным клиентское устройство, а иногда – использовать намного более сложную систему аутентификации, но не контролировать клиентские устройства совсем. Очевидно, что в условиях, когда взаимодействие с системой производится с произвольных устройств – такие системы называются открытыми – контроль этих клиентских устройств обеспечить невозможно, а значит, в них должны использоваться проверяемые имманентные аутентифицирующие данные.

Понимание того, контроль какого компонента должен быть усилен, а какого – может быть ослаблен, тоже можно получить как следствие из предложенной классификации.

Теперь сделаем заявленное в начале главы наложение типов аутентифицирующих данных на характеристики систем и сделаем

выводы об областях применения данных разных типов, приемлемых с точки зрения защиты информации.

Системы, которые будем рассматривать как объект аутентификации, относительно которого аутентифицируется субъект аутентификации, будем считать защищенными, иначе нет смысла оценивать, какие именно аутентифицирующие данные в них следует или не следует применять.

Значит, остается 2 параметра, показатели по которым могут быть учтены:

1) доверенное или нет устройство, с помощью которого субъект предъявляет свои данные, и
2) доверенная или нет среда передачи этих данных.

Область применения и/а в целом в таком случае можно визуализировать так, как показано на рис. 1.7, и для данных разных типов разметим, где их можно, а где нельзя применять.

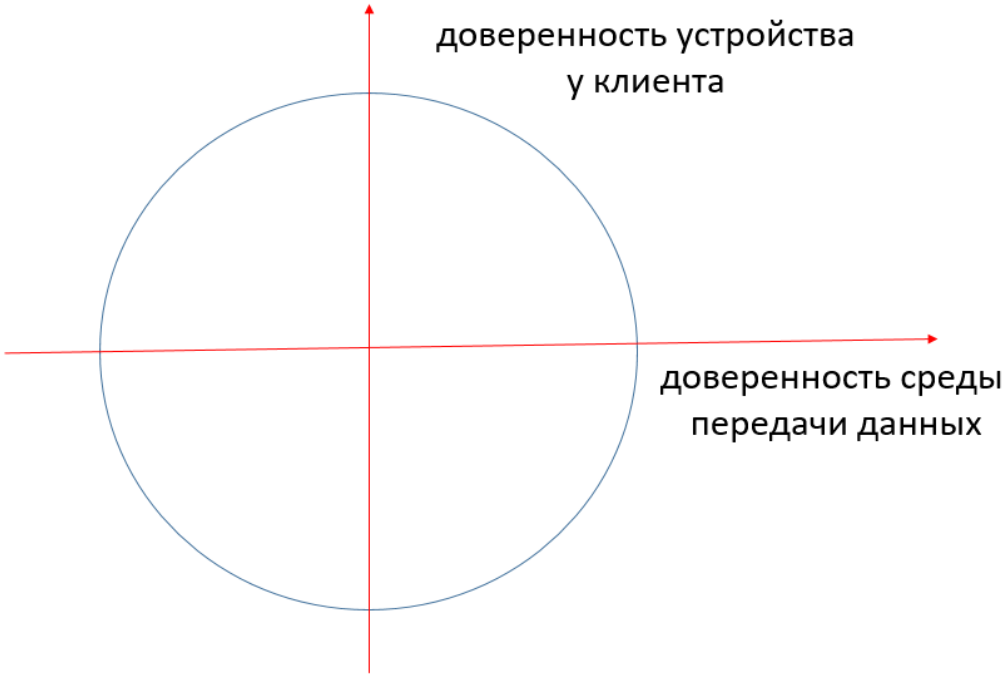


Рис. 1.7. Область применения и/а

Первыми рассмотрим сравниваемые, тиражируемые между объектами аутентификации данные, вне зависимости от того, каковы их признаки по отношению к субъектам аутентификации, то есть весь первый столбец таблицы 1.3.

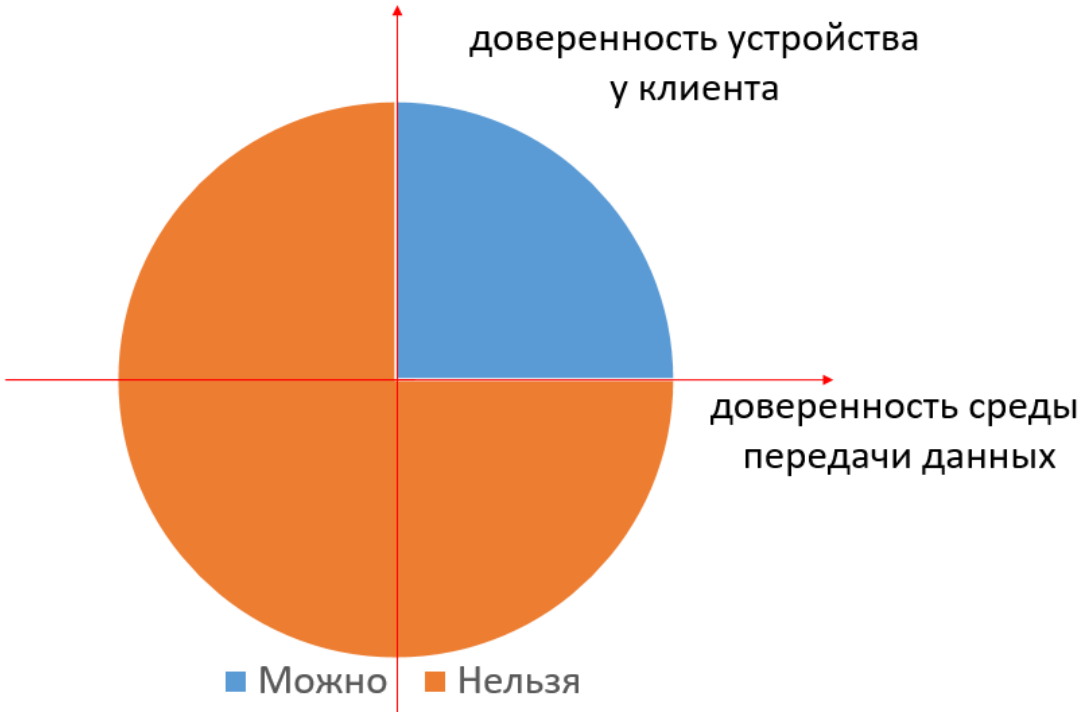


Рис. 1.8. Область применения сравниваемых, тиражируемых между объектами аутентификации данных

Это такие данные, как пароли, аппаратные идентификаторы, ФИО (аналоговые документы), статические биометрические признаки.

Такая узкая область применения определяется тем, что такие данные наиболее уязвимы к компрометации, и только в предельно защищенных условиях доверенной среды их применение может быть безопасным.

Следующие данные – второй столбец таблицы 1.3, сравниваемые, нетиражируемые между системами (объектами аутентификации) данные, вне зависимости от того, каковы их признаки по отношению к субъектам (они бывают только ассоциированными, но могут быть и тиражируемыми, и нет). Их область применения показана на рис. 1. 9. Это гостевая карта СКУД, талончик на очередь, высылаемые одноразовые пароли, токен аутентификации.

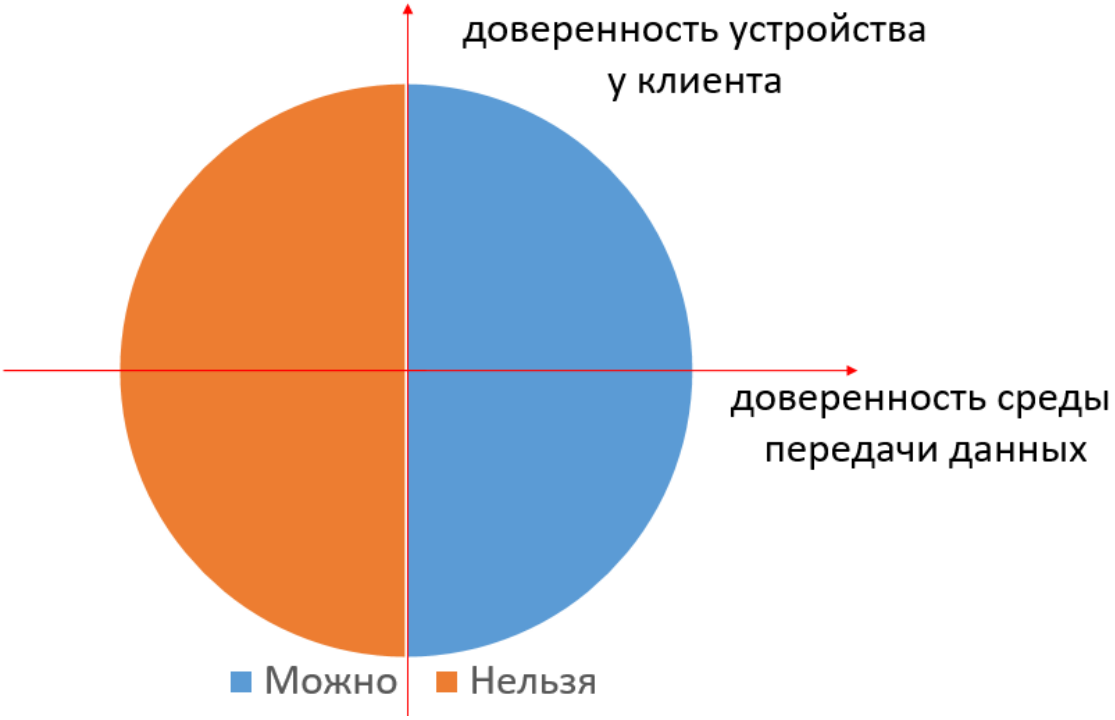


Рис. 1.9. Область применения сравниваемых, не тиражируемых между объектами аутентификации данных

Такое расширение области применения связано с тем, что их компрометация именно при передаче не имеет смысла, поскольку они релевантны только для целевой системы.

А вот для не тиражируемых между объектами проверяемых аутентифицирующих данных (3-й столбец таблицы) область применения не будет одинаковой.

Для ассоциированных признаков (ЭП, OTP, атрибутные сертификаты) область применения будет выглядеть так, как показано на рис. 1.10, а для неотделимых (динамические биометрические признаки) – так, как показано на рис. 1.11.

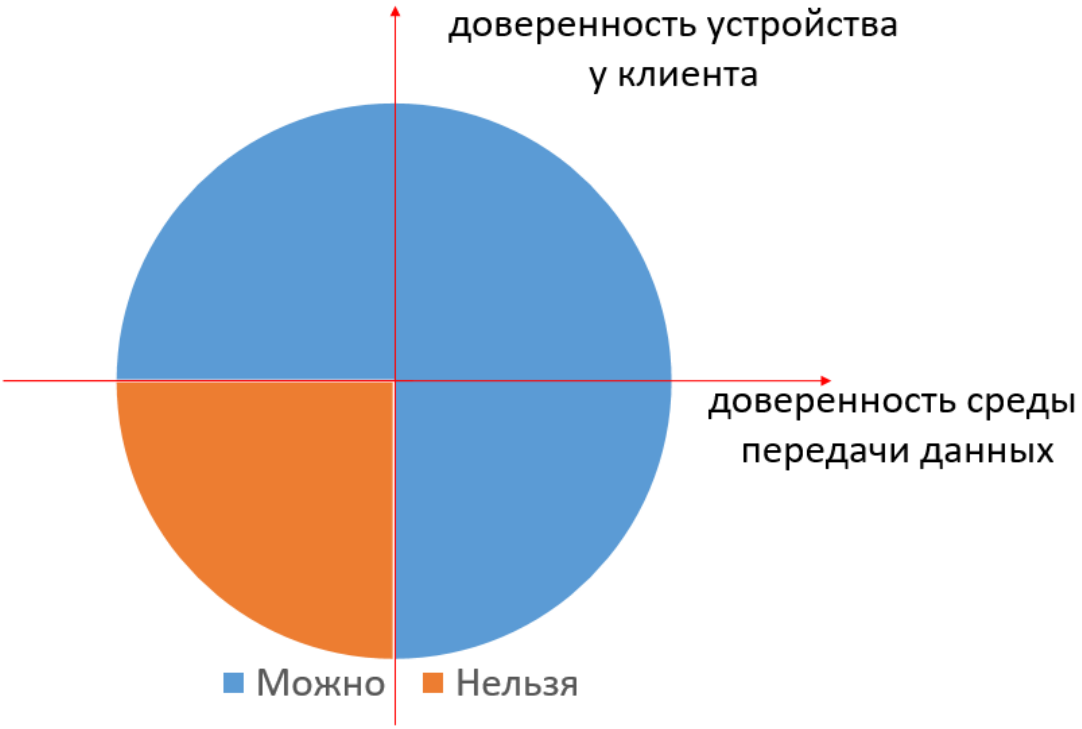


Рис. 1.10. Область применения проверяемых, не тиражируемых между системами, ассоциированных с субъектами аутентифицирующих данных

Для таких данных критичным становится сочетание недоверенной среды передачи данных и недоверенного клиентского компьютера, так как в этих условиях возможен перехват отправленных аутентифицирующих данных и отправка их в другой сессии. Поскольку доставка легального пакета не состоится, то даже одноразовые данные корректно сработают.

А для проверяемых, нетиражируемых и неотделимых от субъекта данных – динамических биометрических данных – не критична недоверенная среда передачи данных, то есть их можно применять в условиях, когда недоверенным является все, кроме серверной части системы, относительно которой аутентифицируется субъект.

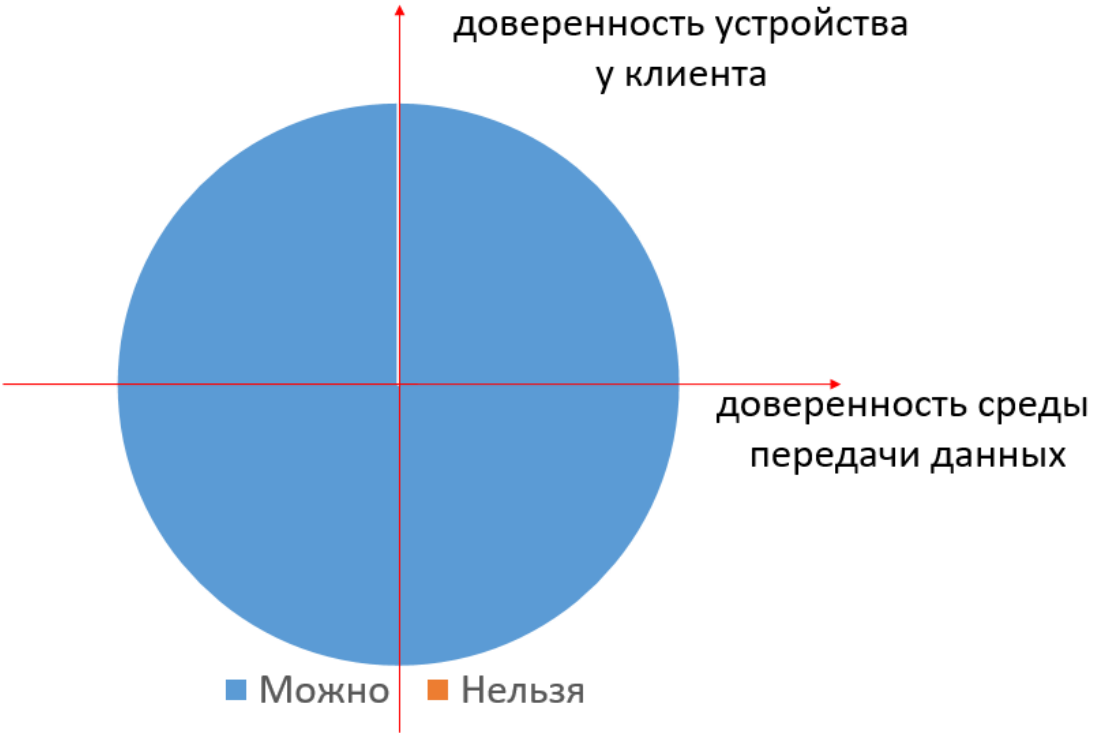


Рис. 1.11. Область применения проверяемых, не тиражируемых между системами, неотделимых от субъекта аутентифицирующих данных

Однако необходимо признать, что эта особенность динамических биометрических признаков не связана с их отличием от предыдущей группы (неотделимостью от субъекта в отличие от ассоциированности с ним). А связана она с тем, что генерация и демонстрация стимула, а также считывание и передача реакции производятся в режиме реального времени: сервер однократно генерирует уникальный стимул в каждой конкретной сессии, и в ней же получает реакцию на него. Разрыв сессии приведет к тому, что процесс аутентификации не состоится. То есть залогом нечувствительности к недоверенности среды передачи данных и клиентского устройства является то, что форма существования динамических биометрических признаков – процесс, а не данные.

Это подтверждается проверкой на неотделимом не тиражируемом между объектами аутентифицирующим признаке ПО – ЭП поставщика. Этот признак отличается от ЭП человека ровно тем, что для человека ЭП – это ассоциированный признак, а для файла – неотделимый (отделим ЭП и файл станет другим). При этом

область применения будет такой же, как у ЭП человека, с учетом того обстоятельства, что использование для подписи своего ПО поставщик и самому вряд ли придет в голову использовать недоверенный компьютер.

Поскольку других аутентифицирующих данных пользователя, не связанных с интерактивной динамической биометрией, с аналогичным набором характеристик пока не выявлено, предположение о том, что данная область применения релевантна для всего типа – остается предположением, убедительно обоснованным только для динамической (рефлекторной) биометрии. Весьма вероятно, что этот набор характеристик для аутентифицирующих признаков человека находится в дополнительной дистрибуции с формой существования в виде процесса, то есть невозможно, чтобы признак, неотделимый от субъекта-человека и не тиражируемый между объектами аутентификации существовал в форме данных. Однако это предположение еще нуждается в проверке.

Отдельно необходимо напомнить, при взаимной аутентификации, как правило, стороны используют аутентифицирующие данные разных типов (за исключением отдельных механизмов, таких как криптографическое рукопожатие), и области применения тех или иных данных в этом случае еще предстоит смоделировать.

Глава 2. ОЦЕНКА ИЗМЕНЕНИЯ ДОВЕРЕННОСТИ НАБОРОВ ДАННЫХ ПРИ ИХ КОМПЛЕКСИРОВАНИИ И ПЕРСПЕКТИВЫ ОЦЕНКИ ДОВЕРИЯ К ОТЧУЖДЁННЫМ ОБУЧЕННЫМ МОДЕЛЯМ

А. Ю. Ситников

Современный этап развития искусственного интеллекта характеризуется парадоксальной ситуацией: математические основания обработки информации, разработанные ещё во второй половине XX века, остаются для большинства классов задач прежними, тогда как практическая эффективность интеллектуальных систем за последние двадцать лет выросла на порядки. Это противоречие разрешается, как только мы перестаём рассматривать информационный процесс как замкнутый процесс, и обращаем внимание на тот объект, над которым он совершается, — данные. Информационные процессы хорошо изучены: классические работы по теории информации К. Шеннона [18. Гл. 6–17], по статистической оценке [19. Гл. 2] и проверке гипотез [20. Гл. 3], по теории случайных процессов и теории алгоритмов сформировали тот аппарат, в рамках которого работают практически все современные методы машинного обучения. Однако сами данные, к которым этот аппарат применяется, до сих пор остаются объектом, изученным существенно слабее, и именно качество данных, а не качество алгоритма, всё чаще оказывается лимитирующим фактором интеллектуальных систем.

Информационный процесс в широком смысле представляет собой целенаправленное преобразование одного набора сведений в другой [4, 21, 22]. Под этим понятием объединяются операции сбора, хранения, передачи, обработки и представления информации. Каждая из этих операций имеет за собой длительную традицию формального изучения: теория кодирования описывает, как информация может быть представлена в виде дискретных сигналов; теория передачи информации устанавливает фундаментальные пределы пропускной способности канала и зависимость количества передаваемой информации от уровня шума; теория алгоритмов задаёт ограничения на сложность вычислительных процедур; теория вероятностей и математическая статистика формируют аппарат, позволяющий извлекать выводы из выборки; машинное обучение и интеллектуальный анализ данных объединяют эти инструменты в законченные методологии. Информационный процесс при таком взгляде выступает как формальная структура, обладающая собственными свойствами: устойчивостью, сходимостью, скоростью, погрешностью.

Иное положение занимают данные. Несмотря на то, что без них не существует ни одного информационного процесса, понятие данных сохраняет преимущественно прикладной характер и зависит от контекста конкретной системы. В разных сферах жизни одни и те же операции и преобразования применяются к данным максимально различным между собой: к сигналам датчиков, к результатам опросов, к текстовым корпусам, к графам взаимосвязей, к параметрам поведения пользователей. Несмотря на это, формальные характеристики данных — то, что отличает один набор данных от другого с точки зрения пригодности его использования описаны в литературе фрагментарно. Существуют отдельные модели качества данных, [23. Гл. 3] отдельные модели провенанса [24], отдельные модели доверия в сетях источников [25], но единого аппарата, который позволял бы характеризовать наборы данных как объекты, изменяющиеся во времени и вступающие в отношения друг с другом, ещё не сформировано.

Данная глава посвящена одной из таких отсутствующих характеристик — доверенности набора данных. Под доверенностью в дальнейшем понимается интегральная оценка пригодности данных

для использования в задаче в условиях неопределённости, противоречий и неполноты исходной информации. Доверенность не равносильна истинности — последняя относится к соответствию данных реальным фактам и в общем случае является ненаблюдаемой; не тождественна и надёжности, которая описывает свойства источника, а не самого набора. Доверенность, в отличие от этих понятий, оказывается удобным объектом для количественной оценки, поскольку допускает значения на непрерывной шкале, может меняться постепенно и реагирует на типовые операции с данными — на их объединение, на пополнение, на удаление. Данная операционность делает доверенность ключевой характеристикой данных в системах доверенного искусственного интеллекта.

Необходимость в количественной оценке доверия к данным приобретает дополнительную актуальность в связи с тенденцией к объединению наборов данных, поступающих из различных источников. Современные системы машинного обучения предъявляют постоянно растущие требования к объёму обучающей выборки. Эмпирические наблюдения показывают, что для большинства семейств моделей рост качества тесно связан с ростом объёма обучающих данных, и нередко имеют место степенные зависимости с положительной асимптотой. Однако объём не является самостоятельной характеристикой, влияющей на качество обучения: рост объёма за счёт добавления слабо подтверждаемой или неконтролируемой информации может, напротив, снижать качество модели и приводить к появлению нежелательных артефактов — от смещения распределений до запоминания шумовых паттернов. Объединение наборов данных, или, как иногда говорят, их комплексирование, оказывается в этом смысле не индифферентной операцией: оно может как повышать, так и понижать пригодность результирующего набора, и важно понимать, по каким правилам это происходит.

Существующие методологии в области управления данными — Data Governance, Data Quality, Data Provenance — в значительной степени описывают организационные и инженерные стороны работы с данными. Стандарты ISO 8000 и ISO/IEC 25012 формализуют размерности качества данных: полноту, точность, согласованность, актуальность; модели провенанса PROV-DM и связанные с ними

рекомендации W3C задают язык описания цепочки преобразований; работы по qualified data в области научных исследований акцентируют внимание на воспроизводимости результатов. В то же время математический аппарат, описывающий, как численная оценка доверия к набору данных изменяется при операциях с ним, в этих стандартах развит слабо. Авторы преимущественно работают на уровне категоризации и контрольных вопросов, оставляя количественные оценки на усмотрение конкретных приложений. При этом именно количественные оценки необходимы для того, чтобы интегрировать представления о доверенности в архитектуру информационных систем — в системы метаданных, в правила маршрутизации данных, в политики допуска данных в обучающую выборку.

В последние годы появились отдельные исследования, посвящённые количественной оценке доверия к данным. Одни авторы [26] рассматривают доверие в контексте сетей источников и формализуют его через алгебру субъективной логики; другие [25] — через метрики качества и индексы согласованности [27]; третьи — через модели риска, связанные с использованием данных в принимающих решения системах. [28] Эти подходы, при всём их разнообразии, имеют общую черту: они фокусируются на оценке текущего состояния данных, но не описывают динамику доверенности при операциях, типичных для систем машинного обучения. При этом именно динамика — то, как доверенность изменяется при последовательном добавлении новых наборов данных, при их слиянии, при их фильтрации — представляет наибольший интерес для практики. Без модели этой динамики невозможно проектировать политики управления данными, ориентированные на сохранение или повышение их доверенности.

В этой главе развивается подход, в рамках которого доверенность набора данных рассматривается как состояние системы, а операция комплексирования — как переход между состояниями. Такой взгляд позволяет содержательным образом задействовать аппарат марковских процессов, дающий математически строгое описание систем, эволюция которых определяется текущим состоянием и не зависит от полной истории прошлых переходов. Марковское свойство в данной модели не

является произвольным предположением: оно отражает фундаментальное свойство операции комплексирования, при которой итоговый набор данных характеризуется собственной доверенностью, и для дальнейшего объединения с другими наборами не имеет значения, каким именно путём он был получен. Любая релевантная история уже отражена в текущем значении доверенности. Принятие этого допущения позволяет ввести функциональное описание изменения доверенности, построить матрицу переходов, провести вычислительные эксперименты и проанализировать асимптотическое поведение модели.

Само понятие доверенного искусственного интеллекта (trusted AI) развивается в нескольких параллельных направлениях. С одной стороны, оно ассоциируется с интерпретируемостью моделей и возможностью объяснения их решений [29]; с другой — с устойчивостью к состязательным воздействиям и к смещениям распределенийи [30]; с третьей — с соответствием этическим, правовым и регуляторным требованиям[1]. Менее заметным, но не менее важным является четвертое направление, связанное с доверием к самим данным, на которых система обучается и принимает решения. Если данные, лежащие в основе работы системы, не вызывают доверия, то и решения, получаемые на их основе, не могут считаться доверенными независимо от выбранной модели и качества её интерпретируемости. Именно в этом смысле оценка изменения доверенности данных при их комплексировании становится составной частью построения систем доверенного искусственного интеллекта: она поставляет ту информацию, без которой невозможно ни осмысленное управление обучающей выборкой, ни последующая сертификация работы модели.

Актуальность такого подхода обусловлена, помимо прочего, ростом значения систем, в которых данные поступают из множества источников с неподтверждённым уровнем доверия. К этому классу относятся современные платформы федеративного обучения, в которых модель обучается на распределённых наборах данных, не покидающих своих локальных хранилищ; системы интеграции данных в крупных корпоративных информационных контурах,

[1] NIST: AI Risk Management Framework (AI RMF 1.0)

объединяющие сведения из десятков унаследованных систем; системы поддержки принятия решений в государственном секторе, опирающиеся на ведомственные базы с различной степенью валидации; научные исследовательские конвейеры, агрегирующие данные открытых научных репозиториев. Во всех этих случаях упрощённое предположение о равной доверенности всех источников оказывается необоснованным, и возникает необходимость в формальной процедуре, позволяющей оценить, как меняется доверенность результирующего набора по мере его пополнения данными из этих источников.

В настоящей главе рассматриваются наборы данных, используемые в системах искусственного интеллекта, на предмет изменения доверенности этих наборов данных при их последовательном комплексировании, строится формальная модель, описывающая эту динамику, и рассматриваются её свойства на типовых сценариях, отражающих практические условия эксплуатации систем интеграции и обработки данных. Для этого последовательно решаются задачи: определение параметров, влияющих на доверенность; формализация перехода от одного набора данных к другому; построение марковской модели изменения доверенности с обоснованной матрицей переходов; вычислительное моделирование сценариев комплексирования; анализ полученных результатов и формулирование выводов о применимости модели.

Научная новизна излагаемого материала состоит в систематическом применении аппарата марковских процессов к задаче оценки динамики доверенности наборов данных. В отличие от ранее предлагавшихся одношаговых формул, описывающих изменение доверенности при однократном слиянии, в настоящей работе доверенность рассматривается как состояние системы, эволюционирующей во времени, а переходы между её значениями выводятся из детерминированной функции изменения доверенности и распределения характеристик добавляемых данных. Это позволяет получить замкнутое описание процесса комплексирования и проводить количественный анализ его длительной динамики. Кроме того, впервые формулируется представление о доверенности данных как о наблюдаемой

величине, для которой может быть выписана матрица переходов марковской цепи, и предлагается процедура её вычисления, опирающаяся на функцию обновления доверенности и априорное распределение доверенности добавляемых данных.

Вне рассмотрения остаются такие важные вопросы, как формализация провенанса в виде направленных ациклических графов; оценка влияния данных на качество конкретной модели через теорию влияния (influence functions); правовые и регуляторные аспекты использования данных; вопросы конфиденциальности и сохранения приватности субъектов данных. Тем не менее формальный аппарат, развиваемый в главе, образует одну из опорных элементов, необходимых для последующей разработки полноценной теории доверия в системах обработки данных.

Таким образом, центральный тезис настоящей главы может быть сформулирован следующим образом: переход от формального изучения информационных процессов к формальному изучению самих данных требует введения количественных характеристик данных, изменяющихся при операциях с ними; среди этих характеристик доверенность занимает особое место, поскольку выражает интегральную пригодность данных для использования и поддаётся вероятностному моделированию; марковское описание изменения доверенности при комплексировании оказывается первым шагом, открывающим путь к более широкому классу задач, связанных с доверием в системах искусственного интеллекта, включая оценку доверия к обученным моделям и к их выводам.

2.1. ДОВЕРЕННОСТЬ ДАННЫХ КАК ФУНДАМЕНТАЛЬНАЯ ХАРАКТЕРИСТИКА

2.1.1. Понятие доверенности данных

Понятие доверенности данных, используемое в настоящей главе, относится к категории интегральных характеристик, описывающих пригодность данных для применения в условиях неопределённости. Под доверенностью набора данных понимается

численно выражаемая степень обоснованности использования этого набора в качестве источника информации для последующего анализа, обучения моделей и принятия решений. В отличие от истинности или фактической достоверности, которые предполагают сравнение содержимого данных с реальными событиями или фактами, доверенность не требует такого сопоставления и формируется на основе совокупности характеристик самого набора и обстоятельств его получения. Доверенность всегда относительна: набор данных может быть высоко доверен в одних задачах и существенно менее доверен в других, поскольку пригодность зависит не только от качества самих данных, но и от характера решений, опирающихся на них.

В научной и инженерной литературе понятие, близкое по смыслу к доверенности данных, появляется в работах по управлению качеством данных, по информационной безопасности и по теории доверия в распределённых системах. В работах по управлению качеством рассматриваются такие размерности качества данных, как точность, полнота, согласованность, актуальность, и доверенность близка к интегральному показателю, объединяющему эти размерности. В работах по доверию в распределённых системах доверие к данным формализуется через алгебру субъективной логики или через индексы репутации источника. В работах по информационной безопасности доверенность связывается с цепочкой контроля и с провенансом данных. Несмотря на разнообразие подходов, общая черта всех этих направлений — стремление выразить пригодность данных некоторой количественной величиной, поведение которой можно изучать формальными методами.

Доверенность не является бинарным критерием. Если задача требует, чтобы данные обладали или не обладали неким качеством — например, прошли проверку на синтаксическую корректность, — для этого используется фильтрация, а не оценка доверия. Доверенность отражает ту часть характеристик, которые принципиально не сводятся к проверке за конечное число шагов: возможные противоречия с другими источниками, степень согласованности с уже накопленными данными, потенциальный риск использования. Доверенность по своей природе непрерывна,

и удобно считать, что она принимает значения на отрезке от нуля до единицы. Это допущение делает её удобным объектом математического моделирования и позволяет применить аппарат вероятностных переходов для описания её изменений во времени.

$$t(D) \in [0,1]$$

При работе с реальными системами данных, как правило, не предполагается, что значение доверенности набора может быть установлено с абсолютной точностью. Напротив, доверенность является оценкой, опирающейся на доступную информацию: на знание о происхождении данных, на согласованность с другими источниками, на репутацию владельца данных, на ход предыдущих операций с ними. Эта оценка может уточняться по мере получения новой информации, в том числе по результатам взаимодействия данного набора с другими наборами. Данная возможность уточнения и составляет внутреннюю содержательную сторону доверенности, отличающую её от прочих характеристик качества.

2.1.2. Соотношение доверенности с истинностью и надёжностью

Истинность данных — характеристика, отражающая соответствие данных реальным объектам, событиям или явлениям. В тех случаях, когда возможна непосредственная сверка данных с реальностью, истинность является наиболее естественным критерием их оценки. Однако такая сверка во многих практических задачах невозможна: либо реальная ситуация, описываемая данными, более не наблюдаема (исторические данные); либо она доступна только частично (данные о поведении пользователей в распределённых сервисах); либо самой природой задачи предполагается, что данные описывают не наблюдаемые непосредственно сущности (метаданные, оценки экспертов). В этих случаях понятие истинности теряет операциональный смысл, и его место в практическом анализе занимает понятие доверия — выраженное в числовой форме предположение о том, насколько данные пригодны для использования.

Надёжность данных, в свою очередь, обычно связывается со свойствами источника и процесса получения данных, а не с самим полученным набором. Под надёжностью источника понимается его

способность стабильно выдавать данные с известными статистическими характеристиками — калиброванные датчики, контролируемые лабораторные протоколы, сертифицированные информационные системы. Соответственно, когда говорят, что надёжность характеризует процесс, речь идёт о процессе приобретения данных от источника: последовательности измерения, регистрации и передачи, повторное выполнение которой при заданных условиях даёт согласованные результаты.

Существенно, что от надёжного источника могут поступать как высоко, так и слабо доверенные наборы. Характерный пример — сырые показания датчиков. Метрологически датчик может быть надёжен: известна его калибровка, специфицирована точность, протокол выдачи данных воспроизводим. Однако соответствующий набор сырых показаний, не прошедший процедуру согласования с другими наблюдениями — сопоставления с эталоном, устранения дрейфа, синхронизации временных меток, выявления аномалий — оказывается слабо доверенным с точки зрения последующего использования: его прямое применение в задаче принятия решений сопровождается значительным риском ошибочного вывода. Доверенность здесь ниже надёжности именно потому, что надёжный процесс получения данных не гарантирует пригодности результата к конкретной задаче.

Таким образом, надёжность и доверенность не совпадают и характеризуют разные стороны работы с данными: надёжность относится к процессу получения данных от источника, тогда как доверенность — к набору как к объекту последующего анализа и зависит от того, насколько этот набор пригоден к решению поставленной задачи в условиях неопределённости и возможных противоречий с другими источниками.

Доверенность поэтому удобнее рассматривать как отдельное измерение, существующее наряду с истинностью и надёжностью и не сводящееся к ним.

Существенное преимущество доверенности как характеристики заключается в том, что она открыта для математического моделирования. В отличие от истинности, которая фактически является скрытой переменной, доступной лишь в идеализированных условиях, доверенность задаётся как функция от

наблюдаемых характеристик набора и потому может быть вычислена. В отличие от надёжности, относящейся к источнику, доверенность относится к набору и потому может быть исследована на предмет того, как она изменяется при операциях с этим набором. Эта последняя возможность является ключевой для предлагаемого далее моделирования: изменение доверенности при комплексировании наборов данных составляет основной предмет настоящей работы.

2.1.3. Провенанс данных и его влияние на доверенность

Под провенансом данных понимается история формирования набора данных: его источники, последовательность преобразований, операторы, через руки которых он прошёл, контекст принятия решений о включении в него тех или иных записей. Провенанс является важной частью информации, лежащей в основе оценки доверенности: набор данных с прозрачной историей, как правило, заслуживает большего доверия, чем набор, происхождение которого неизвестно или восстанавливается лишь частично. Стандарты PROV-DM и PROV-O, принятые W3C, описывают универсальный язык для записи провенанса в виде направленных ациклических графов сущностей, действий и агентов; этот язык широко используется в системах с автоматизированной отчётностью.

Несмотря на это, провенанс не заменяет собой доверенности и не позволяет автоматически вывести её значение. Подробная история набора данных лишь поставляет исходный материал для оценки, но окончательное решение о доверенности остаётся за тем, кто оценивает данные. В предлагаемом подходе предполагается, что провенанс уже учтён в текущем значении доверенности набора. Это означает, что после того, как доверенность установлена, для целей дальнейшего анализа значение имеет именно она, а не вся история, через которую набор прошёл. Такая редукция оправданна, поскольку дальнейшие операции с набором — в первую очередь его комплексирование с другими наборами — описываются на уровне доверенности и не нуждаются в полной информации о провенансе.

Связь провенанса с доверенностью особенно важна в контексте систем, в которых данные циркулируют между множеством участников. В федеративном обучении, в обмене данными между организациями, в публикации открытых научных датасетов цепочка преобразований нередко настолько длинна, что её полное хранение становится практически нецелесообразным. В этих случаях разумно агрегировать информацию о провенансе в значение доверенности и работать с ним как с компактной сводкой. Преимущество такого подхода — независимость дальнейших операций от объёма исторических данных; ограничение — необходимость аккуратно проектировать функцию, по которой сводка формируется. В настоящей главе рассмотрена одна из возможных функций такого рода — функция изменения доверенности при объединении двух наборов данных.

В результате уточнения понятия доверенности данных проведено его сопоставление с понятиями истинности и надёжности, а также рассмотрена связь с провенансом. Установлено, что доверенность представляет собой интегральную характеристику, отражающую пригодность набора данных для использования, поддающуюся численной оценке и изменяющуюся при операциях с набором. Это позволяет в дальнейшем рассматривать доверенность как состояние системы, эволюционирующей во времени, и применять к ней формальные средства анализа динамики, в первую очередь — аппарат марковских процессов.

2.2. МАРКОВСКИЕ ЦЕПИ КАК ИНСТРУМЕНТ МОДЕЛИРОВАНИЯ

2.2.1. Основные определения марковских процессов

Для формализации процесса изменения доверенности набора данных при его последовательном содержательном комплексировании аппарат марковских процессов. Этот аппарат описывает системы, эволюция которых во времени носит стохастический характер, при этом будущее системы зависит исключительно от её текущего состояния и не зависит от полной истории прохождения через предыдущие состояния. Подобная редукция сложных историй к компактному состоянию делает

марковские процессы одним из основных инструментов моделирования широкого круга задач — от теории очередей до моделирования речи, от моделирования биологических популяций до моделирования финансовых временных рядов.

Дискретная марковская цепь определяется тройкой объектов. Во-первых, фиксируется конечное или счётное множество состояний S = {s_1, s_2, …, sN}, в которых может находиться система; в каждый момент времени система пребывает ровно в одном состоянии. Во-вторых, задаётся матрица переходов P = [p_{ij}], элементы которой описывают вероятность перехода системы из состояния s_i в состояние s_j за один шаг. В-третьих, фиксируется начальное распределение π_0, описывающее вероятности нахождения системы в каждом из состояний в начальный момент. Сочетание этих трёх объектов определяет всю динамику цепи.

Центральным является марковское свойство, формализующее предположение об отсутствии «памяти» у процесса:

$$P(X_{n+1} = j | X_n = i, \dots) = P(X_{n+1} = j | X_n = i)$$

Марковское свойство имеет важное методологическое значение: оно означает, что в состоянии системы уже сосредоточена вся релевантная для дальнейшего поведения информация о прошлом. Никакие дополнительные сведения о том, каким путём система пришла в текущее состояние, не нужны для предсказания её следующего шага. Это допущение лежит в основе всей теории марковских цепей и является той точкой, в которой обычно проверяется применимость аппарата к конкретной задаче.

Вероятности переходов формально определяются как условные вероятности

$$p_{ij} = P(X_{n+1} = j | X_n = i)$$

и образуют матрицу переходов P. Каждая строка этой матрицы представляет собой распределение вероятностей перехода из соответствующего состояния и потому удовлетворяет условиям

$$\sum_j \quad p_{ij} = 1, p_{ij} \geq 0$$

В большинстве практических задач предполагается, что правила перехода неизменны во времени: вероятности p_{ij} не зависят от номера шага. Такая цепь называется однородной. В

однородном случае поведение системы на нескольких шагах может быть описано степенью матрицы переходов:

$$P(X_n = j) = (P^n)_{ij}$$

В сочетании с начальным распределением π_0 это даёт замкнутое описание динамики системы:

$$\pi_n = \pi_0 P^n$$

Помимо чисто формального описания, для марковских цепей развит обширный аппарат анализа предельного поведения: при определённых условиях существует стационарное распределение, к которому система стремится с ростом числа шагов независимо от начального состояния; характеристические времена сходимости связаны со спектром матрицы переходов; топологические свойства цепи (неприводимость, апериодичность) определяют, существует ли единственное предельное распределение и существует ли оно вообще. Эти средства будут использованы при анализе результатов моделирования в четвёртой главе.

2.2.2. Обоснование применения марковских цепей к задаче доверенности

Применение аппарата марковских процессов к моделированию изменения доверенности набора данных требует обоснования, поскольку оно опирается на содержательные предположения о природе операции комплексирования. Первое предположение заключается в том, что доверенность является единственной характеристикой набора данных, с которой придется работать. Это означает, что для целей последующего анализа значение имеет именно интегральная оценка пригодности, а не подробные характеристики содержимого набора — его объём, тематика, структура полей. Допустимость такой редукции связана с предположением, что подробные характеристики уже учтены в текущем значении доверенности и не должны учитываться повторно. То есть доверенность принимается за состояние системы по построению.

Второе предположение — марковость переходов — связано с природой операции комплексирования. При объединении двух наборов данных возникает новый набор, обладающий собственной

доверенностью. С точки зрения дальнейших операций с этим объединённым набором значение играет именно его текущая доверенность, и для последующего объединения с другими наборами не имеет значения, каким именно образом он был получен из исходных. Это обусловлено тем, что содержательная история объединения уже отражена в полученном значении, и для последующего этапа достаточно его одного. Таким образом, при принятом допущении о достаточности текущего значения доверенности процесс последовательного комплексирования может быть описан как марковский: переходы между состояниями определяются текущим состоянием и характеристиками поступающего внешнего набора, но не зависят от полной траектории, по которой система достигла текущего состояния.

Третье предположение — однородность переходов — отражает идею, что правила обновления доверенности не должны зависеть от того, на каком шаге работы системы происходит очередная операция. Это допущение не является самоочевидным: можно представить себе сценарии, в которых правила обновления изменяются по мере увеличения числа объединённых наборов (например, ужесточение требований при росте объёма обучающей выборки). Однако в базовой постановке такие зависимости не вводятся, и переходы предполагаются неизменными вне зависимости от времени. Это позволяет использовать степени матрицы переходов и обеспечивает удобство теоретического анализа.

Четвёртое предположение — конечность пространства состояний. Чтобы получить вычислительно осуществимую модель, отрезок [0, 1], на котором определена доверенность, разбивается на конечное число подынтервалов, и каждому подынтервалу сопоставляется состояние марковской цепи. Такая дискретизация согласуется с практикой использования качественных шкал доверия в управлении данными (например, шкалы «низкое — среднее — высокое доверие») и одновременно делает доступными результаты теории конечных цепей. Шаг дискретизации в дальнейшем выбирается равным 0,1, что обеспечивает разумный баланс между точностью представления и удобством анализа.

При совокупном принятии этих допущений целесообразно описывать процесс комплексирования как марковскую цепь с конечным числом состояний, соответствующих интервалам доверенности, и матрицей переходов, выводимой из функции обновления доверенности и распределения характеристик добавляемых данных. Содержательные ограничения, накладываемые предположениями, изложенными ниже, при анализе модели, и в основном связаны с тем, что некоторые сценарии, в которых правила обновления зависят от истории, выходят за рамки однородной марковской цепи.

2.2.3. Переход между состояниями набора данных

Переход между состояниями марковской цепи в развитой модели описывает изменение уровня доверенности набора данных при его последовательном комплексировании с другими наборами. Формализация переходов опирается на введённые в первой главе понятия доверенности и схожести данных, а также на аксиоматику изменения доверенности при объединении наборов, предложенную автором ранее. Каждому набору данных D сопоставляется числовая характеристика доверенности t(D) ∈ [0, 1], определяющая состояние, в котором находится система. Дискретизация отрезка [0, 1] на интервалы S_i позволяет однозначно сопоставить значению доверенности соответствующее состояние.

$$S_i = [\tau_i, \tau_{i+1})$$

Пусть на шаге n текущий агрегированный набор данных D_n имеет доверенность t(D_n). К нему добавляется внешний набор данных B с доверенностью t(B) и схожестью sim(D_n, B). Результатом комплексирования является объединённый набор C = D_n ∪ B, для которого вычисляется доверенность t(C). В работе автора предложена функция изменения доверенности, удовлетворяющая аксиоматическим требованиям (см. главу 3) и обладающая свойством самоподтверждения данных. В обобщённом виде доверенность объединённого набора задаётся формулой

$$t(C) = \left(\frac{t(A) + t(B)}{2}\right)^{w}$$

где показатель степени w зависит от схожести объединяемых данных и определяется как

$$w = \frac{1}{sim(A,B) + c}$$

Здесь c > 0 — фиксированная константа, регулирующая интенсивность влияния схожести на изменение доверенности. Параметр w играет ключевую роль в модели. При высокой схожести данных (sim близко к единице) показатель w становится меньше единицы, и формула обеспечивает рост доверенности при объединении похожих наборов, включая случай равенства исходных доверенностей. Это соответствует свойству самоподтверждения:

$$t(A \cup A) \geq t(A)$$

Если параметры добавляемого набора рассматриваются как случайные величины, распределения которых отражают неопределённость информации о поступающих данных, то переходы между состояниями приобретают вероятностный характер. В этом случае вероятность перехода из состояния S_i в состояние S_j определяется как

$$p_{ij} = P\big(t(C) \in S_j | t(A) \in S_i\big)$$

и матрица переходов марковской цепи отражает как свойства функции изменения доверенности, так и статистические характеристики поступающих данных. В третьей главе подробно выводится явное выражение для p_{ij} в случае равномерного распределения доверенности добавляемых наборов.

Таким образом, в параграфе показано, что операция комплексирования наборов данных содержательным образом порождает марковский процесс: текущая доверенность является достаточной характеристикой состояния, переход к следующему состоянию определяется характеристиками добавляемого набора, а сами правила перехода могут считаться однородными по времени. Эти выводы создают основу для построения замкнутой математической модели, представленной в ниже.

2.3. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ИЗМЕНЕНИЯ ДОВЕРЕННОСТИ

2.3.1. Формализация состояния набора данных

Система, с которой проведена работа рассматривается в дискретные моменты времени, на каждом из которых происходит обогащение набора данных. На каждом таком шаге имеется текущий агрегированный набор данных, обозначаемый как D_n, являющийся результатом всех предыдущих операций объединения. Основным параметром, описывающим состояние системы, является доверенность текущего набора $t(D_n) \in [0, 1]$. В соответствии с допущениями, рассмотренными выше, значение t(D_n) уже учитывает в себе влияние всей предшествующей истории формирования набора, включая провенанс, качество источников и параметры предыдущих объединений.

Для последующего построения матрицы переходов значения доверенности разбиваются на конечное количество сегментов. Отрезок [0, 1] разбивается на N равных интервалов длиной 1/N с границами $\tau_0 = 0$, $\tau_1 = 1/N$, …, τ_N = 1, и каждому интервалу S_i сопоставляется состояние марковской цепи. В вычислительных экспериментах ниже принимается N = 10, что соответствует разбиению с шагом 0,1. Если $t(D_n) \in S_i$, считается, что система находится в состоянии S_i. Таким образом, состояние марковской цепи соответствует уровню доверенности текущего агрегированного набора данных.

2.3.2. Аксиоматика и формула обновления доверенности

Было предложено четыре аксиомы [31], которым должна удовлетворять формула обновления доверенности при объединении двух наборов.

Аксиома 1 (слияние данных): доверенность объединённого набора должна находиться в разумных границах относительно доверенностей исходных наборов, не превышая верхнего предела и не падая ниже определённого минимального уровня в случае согласованных данных.

Аксиома 2 (снижение доверенности при слиянии с недостоверными данными): если t(B) < t(A) и схожесть мала, то t(A ∪ B) ≤ t(A).

Аксиома 3 (слабое влияние малодостоверных данных): если два набора имеют высокую схожесть и одинаковую доверенность, то доверенность объединения не уменьшается, а слегка возрастает.

Аксиома 4 (объединение набора данных самого с собой): последовательное объединение набора с самим собой не приводит к уменьшению доверенности и при достаточном числе шагов стремится к насыщению.

На основе этой аксиоматики выводится формула обновления доверенности

$$t(C) = \left(\frac{t(A) + t(B)}{2}\right)^{w}$$

с показателем степени

$$w = \frac{1}{sim(A, B) + c}$$

Соответствие формулы аксиомам проверяется напрямую.

Для аксиомы 1: если t(A) и t(B) принадлежат отрезку [0, 1], то их среднее арифметическое также принадлежит этому отрезку, а возведение в положительную степень сохраняет это свойство.

Для аксиомы 2: при малом sim показатель w оказывается большим единицы, и возведение в такую степень числа из (0, 1) уменьшает его, что соответствует ожидаемому снижению.

Для аксиомы 3: при больших sim показатель w становится меньшим единицы, что приводит к росту значения по сравнению с исходным средним.

Для аксиомы 4: при объединении набора с собой среднее равно его текущей доверенности, и формула даёт постепенный рост, сходящийся к единице, но никогда её не достигающий, что обеспечивает разумную асимптотику самоподтверждения. Константа с регулирует значительность зависимости от схожести и обычно выбирается равной 0,8, что обеспечивает плавный переход между режимами роста и снижения доверенности.

2.3.3. Вычисление вероятностей переходов

После того как определена детерминированная функция обновления доверенности, ключевая задача состоит в выводе вероятностей переходов p_{ij} между интервалами доверенности. В рамках принятой модели предполагается, что в момент поступления нового набора данных **B** его доверенность неизвестна и моделируется как случайная величина, равномерно распределённая на отрезке [0, 1]. Это допущение соответствует ситуации максимальной неопределённости и используется как базовая гипотеза, на фоне которой могут быть рассмотрены более информативные распределения.

Пусть текущий набор **A** имеет доверенность a ∈ S_i (для дельты a представляется серединой соответствующего интервала). После объединения формируется доверенность t(C) = ((a + b) / 2)^w, где b = t(B) — случайная величина с равномерным распределением на [0, 1], а w фиксирована при заданном sim. Функция g(b) = ((a + b)/2)^w монотонно возрастает на отрезке [0, 1], что позволяет однозначно сопоставить каждому интервалу значений t(C) соответствующий диапазон значений b. Для вычисления вероятности перехода в состояние S_j используется обратное соотношение

$$b = 2\tau_j^{\frac{1}{w}} - a$$

С учётом ограничения b ∈ [0, 1] границы интервала по переменной b принимают вид

$$b_{lo} = max\left(0{,}2\tau_j^{\frac{1}{w}} - a\right)$$

и, аналогично, для верхней границы. Поскольку b распределена равномерно, вероятность попадания доверенности в интервал S_j равна длине соответствующего интервала по b:

$$p_{ij} = b_{hi} - b_{lo}$$

Полученная формула задаёт элементы матрицы переходов P в явном виде, причём как функция от характеристик добавляемых данных (через параметр w), так и от текущего состояния (через значение a). При фиксированном sim и одной и той же модели поступающих данных матрица переходов оказывается стационарной во времени, что и обеспечивает однородность марковской цепи. Если sim меняется во времени или зависит от случайного процесса

(например, в сценарии 4, описанном в параграфе 1.4), формально цепь становится неоднородной, но в каждый конкретный момент её динамика по-прежнему описывается матрицей P, вычисляемой по приведённому правилу.

2.3.4. Свойства матрицы переходов и интерпретация модели

Матрица переходов, получаемая по приведённой формуле, обладает рядом важных свойств, делающих её удобной для дальнейшего анализа. Во-первых, каждая её строка является корректным распределением вероятностей, что обеспечивается тем, что длины интервалов по b неотрицательны и пересекают только участки [0, 1]. Во-вторых, при высоких значениях sim матрица сосредотачивает массу около диагонали и в верхних интервалах, отражая эффект самоподтверждения данных. В-третьих, при низких значениях sim матрица распределяет массу преимущественно в средних и нижних интервалах, отражая эффект деградации доверенности. В-четвёртых, при умеренных значениях sim возникает многосторонний компромисс, при котором система стабилизируется в области средних значений доверенности.

Содержательная интерпретация модели заключается в следующем. На каждом шаге система находится в состоянии, соответствующем некоторому уровню доверенности её текущего набора данных. При получении внешнего набора с неизвестной доверенностью система переходит в новое состояние, причём вероятности переходов определяются известной функцией обновления доверенности и предположением о распределении доверенности добавляемых данных. Если поступающие данные систематически близки к текущему набору, доверенность стабилизируется на высоком уровне; если они систематически удалены, доверенность снижается; в смешанных режимах возникают промежуточные распределения.

Отметим важную особенность модели: вероятности переходов не задаются эмпирически или экспертно, а выводятся из формальной аксиоматики и предположения о распределении неизвестных характеристик. Это обеспечивает воспроизводимость модели и возможность её проверки в численных экспериментах.

Кроме того, при изменении базовых предположений (например, при переходе от равномерного распределения доверенности добавляемых данных к более информативному) формула вычисления p_{ij} может быть пересчитана, что делает модель гибкой по отношению к различным сценариям и областям применения.

Таким образом, построена замкнутая математическая модель изменения доверенности набора данных при его последовательном комплексировании. Сформулирована аксиоматика, на которой строится функция обновления доверенности, выведена явная формула вычисления вероятностей переходов между состояниями марковской цепи и обсуждены свойства полученной матрицы переходов. Содержательная интерпретация модели согласуется с практическими ожиданиями: при близких данных доверенность растёт, при удалённых — снижается, при смешанном потоке — стабилизируется в области средних значений. Полученная модель служит основой для вычислительных экспериментов, описываемых ниже.

2.4. МОДЕЛИРОВАНИЕ ТИПОВЫХ СЦЕНАРИЕВ КОМПЛЕКСИРОВАНИЯ

2.4.1. Постановка моделируемых сценариев

В предыдущем параграфе представлено развитие математической модели изменения доверенности при комплексировании, основанной на аппарате марковских цепей. Доверенность текущего агрегированного набора данных рассматривается как состояние системы, а добавление нового набора — как переход между этими состояниями. Существенной особенностью модели является явный учёт неопределённости доверенности добавляемых данных через предположение о её равномерном распределении и вероятностное описание результата объединения.

Цель моделирования в настоящем параграфе — изучить поведение доверенности в типовых сценариях комплексирования и охарактеризовать асимптотические распределения. Для этого выбраны четыре сценария, отражающие различные практические

ситуации. Первые три сценария являются достаточно интуитивно прозрачными и используются для проверки корректности модели — в них ожидаемое поведение известно заранее. Четвёртый сценарий представляет наиболее общую ситуацию, в которой характеристики поступающих данных варьируются во времени, и его исследование без построенной модели было бы затруднительно.

Во всех рассматриваемых сценариях начальное состояние системы соответствует полностью доверенному набору данных — доверенность принадлежит интервалу [0,9; 1,0]. Это соответствует ситуации, в которой начальный набор сформирован в условиях строгой валидации: например, прошёл ручную проверку, был получен из авторитетного источника или прошёл автоматизированную процедуру верификации. Такое начальное состояние позволяет в наиболее очевидной форме изучить процесс изменения доверенности по мере шагов комплексирования.

Сценарий 1 (самоподтверждение, sim = 0,95) описывает последовательное добавление к текущему набору почти идентичной по характеристикам информации. Параметр схожести зафиксирован равным 0,95, что отражает высокую, но не абсолютную степень совпадения. На практике такому сценарию соответствуют ситуации повторного включения данных, полученных из близких источников, или повторной выборки в условиях стабильного измерительного процесса.

Сценарий 2 (загрязнение шумом, sim = 0,05) описывает добавление слабосвязанных, нерелевантных или ошибочных данных. Параметр схожести зафиксирован равным 0,05. На практике подобному сценарию соответствуют ситуации поступления неконтролируемых данных, неправильной маркировки записей или артефактов сбора. Ожидаемое поведение — снижение доверенности и стабилизация её на сравнительно низком уровне.

Сценарий 3 (частично согласованные данные, sim = 0,5) описывает добавление данных со средней степенью схожести. Подобный режим характерен для интеграции данных из разных, но связанных источников: например, объединение данных нескольких региональных подразделений одной компании, или агрегация данных из нескольких лабораторий, изучающих один и тот же объект. Ожидается, что в этом сценарии доверенность

стабилизируется в области средних значений и не достигает крайних состояний.

Сценарий 4 (случайная схожесть) описывает наиболее общую ситуацию: степень схожести добавляемых данных заранее неизвестна и на каждом шаге выбирается случайно. Параметр sim моделируется как случайная величина, равномерно распределённая на отрезке [0, 1]. На практике этому соответствует случай, когда данные поступают из множества источников с непредсказуемой характеристикой совпадения. Этот сценарий служит общим тестом устойчивости модели в условиях неоднородного потока данных.

2.4.2. Методика моделирования

Для численного исследования поведения доверенности используется программная реализация построенной марковской модели. Поскольку аналитическое описание динамики возможно лишь для одного шага комплексирования, а поведение на длинных горизонтах требует последовательного применения матрицы переходов, исследование проводится с применением вычислительных средств. Реализация выполнена на языке Python без использования внешних библиотек для вероятностного моделирования, что обеспечивает прозрачность алгоритмической стороны и воспроизводимость результатов.

Состояние системы представлено вектором распределения вероятностей π, элементы которого π_i соответствуют интервалам доверенности S_i. Начальное распределение π_0 задано как вырожденное распределение, сосредоточенное в верхнем интервале [0,9; 1,0]. На каждом шаге распределение обновляется по правилу π_{n+1} = π_n P, где матрица P вычисляется по приведённой выше формуле. Для сценариев 1–3 матрица P фиксирована во времени и вычисляется однажды; для сценария 4 матрица P пересчитывается на каждом шаге в соответствии с выбранным значением sim.

Корректность аналитического вычисления вероятностей переходов проверена методом Монте-Карло. Для каждого исходного состояния многократно (с числом испытаний порядка 200 000) проводится имитация добавления нового набора данных,

фиксируется получившееся значение t(C), относится к соответствующему интервалу. Эмпирические частоты используются для оценки p_{ij} и сравниваются с аналитическими. Совпадение в пределах статистической погрешности подтверждает корректность как аналитической формулы, так и её программной реализации.

Моделирование проводится на горизонтах 5, 15 и 50 шагов. Эти значения выбраны так, чтобы охватить как переходный режим (5 шагов), так и режим выхода на квазистационарное распределение (50 шагов). Промежуточное значение (15 шагов) позволяет уловить характерные времена релаксации модели и сопоставить их с асимптотикой. Для каждого сценария фиксируется распределение доверенности на этих горизонтах, и средние значения сравниваются между собой.

2.4.3. Результаты моделирования

В первом сценарии быстро устанавливается стационарное распределение, сосредоточенное в верхних интервалах доверенности. Основная масса вероятности удерживается в области [0,8; 1,0], при этом верхний интервал [0,9; 1,0] не является доминирующим: формула обновления доверенности приводит к насыщению, и значения, близкие к единице, оказываются достижимыми лишь асимптотически. Среднее значение доверенности после 50 шагов составляет приблизительно 0,9, что согласуется с интуитивным ожиданием при повторном объединении почти идентичных данных. Различия между распределениями на горизонтах 5, 15 и 50 шагов оказываются минимальными, что подтверждает быстрое достижение квазистационарного режима.

Во втором сценарии устанавливается совершенно иное стационарное распределение. Вероятность нахождения доверенности в верхних интервалах быстро убывает, а масса концентрируется в области средних и нижних значений. Среднее значение доверенности после 50 шагов составляет приблизительно 0,40, что соответствует существенной деградации исходного полностью доверенного набора при систематическом загрязнении шумом. Распределение остаётся достаточно широким, что отражает

сохраняющуюся вариативность результата объединения при низкой схожести.

В третьем сценарии распределение стабилизируется в области средних значений с центром масс около 0,65. Полная деградация не происходит, однако крайние значения доверенности (как высокие, так и низкие) становятся маловероятными. Это соответствует компромиссу между качеством исходного набора и неопределённостью добавляемых данных, естественному для типовых задач интеграции данных из разнокачественных, но связанных источников.

В четвёртом сценарии возникает наиболее сложная картина. На малых горизонтах (5 шагов) распределение существенно различается между прогонами в силу случайного характера sim; на средних горизонтах (15 шагов) оно постепенно сглаживается; на больших горизонтах (50 шагов) распределение становится близким к распределению третьего сценария, со средним значением около 0,60. Это согласуется с математическим ожиданием: при равномерном распределении sim на [0, 1] среднее значение sim равно 0,5, что соответствует третьему сценарию, и асимптотика к нему сходится. Тем не менее на промежуточных горизонтах поведение четвёртого сценария отличается заметной вариативностью, что важно учитывать при практических оценках.

2.4.4. Анализ поведения доверенности

Совокупный анализ четырёх сценариев позволяет выявить ряд устойчивых закономерностей.

Во-первых, модель воспроизводит ожидаемое содержательное поведение в крайних режимах: при близких данных доверенность сохраняется, при удалённых — снижается. Это служит первичной проверкой корректности модели.

Во-вторых, асимптотические распределения существенно зависят от характеристик потока поступающих данных, и в стационарном режиме доверенность стабилизируется в области, определяемой средним уровнем схожести.

В-третьих, переходные режимы характеризуются конечными временами релаксации, причём при высокой схожести релаксация происходит существенно быстрее, чем при низкой.

В-четвёртых, при неоднородном потоке стационарное распределение определяется именно средним уровнем характеристик, а не отдельными выбросами, что согласуется с математическим ожиданием марковских цепей с переменной матрицей переходов.

Эти выводы имеют ряд практических следствий. В системах, проектируемых с учётом контроля доверенности обучающих данных, имеет смысл задавать целевой уровень среднего sim, обеспечивающий желаемое стационарное распределение доверенности. При проектировании политик допуска данных в обучающую выборку существенен не только текущий уровень доверенности данного источника, но и ожидаемый уровень схожести его данных с уже накопленными. В системах интеграции с неоднородным потоком данных полезно оценивать средние характеристики входящего потока, поскольку именно они определяют асимптотическое поведение.

Дополнительно, модель позволяет интерпретировать ряд эмпирически наблюдаемых явлений в системах машинного обучения. Эффект «деградации модели от обучения на собственных выходах» — известный феномен, при котором последовательное переобучение модели на её же собственных порождениях приводит к ухудшению качества — допускает интерпретацию в терминах настоящей модели: повторное объединение с почти идентичными, но не идентичными данными приводит к насыщению доверенности, не достигающему максимальной отметки, и при определённых условиях может приводить к нежелательному смещению. Эффект «выравнивания» качества крупных датасетов при их агрегации также находит интерпретацию через четвёртый сценарий: при объединении большого числа источников с разнообразными характеристиками доверенность стабилизируется в области средних значений, что наблюдается на практике.

Так, в результате вычислительного моделирования четырёх типовых сценариев комплексирования и анализа полученных распределений доверенности подтверждено качественное

соответствие модели интуитивно ожидаемому поведению в крайних режимах и охарактеризовано её поведение в смешанном режиме. Показано, что асимптотические распределения определяются средним уровнем схожести в потоке поступающих данных, что имеет важные практические следствия для проектирования систем интеграции и обработки данных. Полученные результаты создают основу для дальнейших исследований, в первую очередь — для применения подхода к оценке доверия к обученным моделям, представленного ниже.

2.5. ПЕРСПЕКТИВЫ: ОЦЕНКА ДОВЕРИЯ К ОТЧУЖДЁННЫМ ОБУЧЕННЫМ МОДЕЛЯМ

2.5.1. Постановка задачи

Развитая в предыдущих параграфах теория описывает изменение доверенности набора данных при операциях комплексирования. Логическим продолжением подхода является перенос аналогичной формализации на обученные модели. В современных условиях существенная часть систем искусственного интеллекта оказывается отчуждённой от исходных данных, на которых она обучалась: модель передаётся в виде набора весов, поставляется через API, реплицируется как библиотека или включается в качестве компонента в обрабатывающий конвейер, в котором исходные обучающие данные уже недоступны. Возникает задача оценки доверия к такой модели в условиях отсутствия прямого доступа к её обучающим данным.

Эта задача отличается от стандартной задачи оценки качества модели на тестовой выборке. Тестовое качество измеряется метриками точности, полноты, F-меры, AUC и тому подобными показателями, отражающими согласие модели с конкретной выборкой; оно зависит от свойств самой выборки и не отвечает на вопрос о пригодности модели к работе в условиях, отличающихся от тестовых. Доверенность модели, в отличие от тестового качества, должна отражать обоснованность её использования в классе задач, для которого она разрабатывалась, и устойчивость её поведения к

изменению контекста применения. Формальная постановка задачи такова: требуется построить функцию

$$T(M) \in [0,1]$$

значение которой характеризует степень доверия к модели M на отрезке [0, 1] и удовлетворяет ряду естественных требований: согласованность с метриками поведения модели на тестовых выборках; учёт информации о происхождении модели (если она доступна) и о её производных характеристиках; монотонное реагирование на типовые модификации модели — дообучение, тонкую настройку, дистилляцию; способность учитывать передачу модели через цепочки распространения.

2.5.2. Состояние области и существующие подходы

В современной литературе по доверенному искусственному интеллекту проблема оценки доверия к отчуждённым обученным моделям рассматривается в нескольких направлениях. Описательные подходы — модельные карточки (model cards) [32] и карточки наборов данных (datasheets for datasets) [33] — предполагают, что вместе с моделью поставщик публикует структурированные сведения о её источниках, условиях обучения, целевой области применения и измеренных метриках. Диагностические подходы — методы оценки калибровки (expected calibration error) [34], оценки робастности (по отношению смещениям распределений) [35], оценки неопределённости (Bayesian-методы, ensembles) [36] и детекции данных вне распределения [37] — дают числовые характеристики моделей, однако каждая отражает один аспект пригодности модели и не объединяется с другими в общий интегральный показатель.

Отдельную линию составляют работы, посвящённые количественной оценке доверия в форме интегральных метрик. В работе Wong и соавторов (2020) [38] предложены метрики Question-Answer Trust, Trust Density, Trust Spectrum и NetTrustScore, интегрирующие доверие модели по всем возможным сценариям ответа. В работе Dai, Lin, Bertino и Kantarcioglu (2008) [39] описана процедура оценки доверенности данных на основе провенанса, в которой доверие к выводу формируется как функция доверия к

источникам и характеристик передачи. В работах последних лет (DLProv, Model DNA) [40] развивается линия инфраструктурного описания провенанса моделей, в которой каждой модели сопоставляется криптографически защищённая цепочка преобразований обучающих данных и параметров обучения.

Несмотря на разнообразие предложенных решений, в существующих подходах отсутствует интегральная числовая характеристика доверия к отчуждённой обученной модели, которая: (1) выводилась бы из явной аксиоматики; (2) учитывала бы информацию о провенансе и о метриках поведения модели в единой формуле; (3) изменялась бы по предсказуемым правилам при типовых операциях с моделями. Разработка такой характеристики составляет одну из центральных задач намечаемого направления.

2.5.3. Эскиз модели доверия к отчуждённой модели

Естественное обобщение предложенной в разделах 2.1–2.5 модели на случай обученных моделей состоит в том, чтобы рассматривать доверенность модели T(M) как функцию двух составляющих: вклад данных, на которых модель была обучена, и вклад собственных свойств модели, проявляющихся в её поведении. Содержательно эти составляющие отвечают двум источникам информации: первый связан с процессом, второй — с результатом. Их объединение представимо как декомпозиция

$$T(M) = F\big(t(D_{train}), q(M), r(M)\big)$$

где t(D_train) — интегральная доверенность обучающих данных, q(M) — характеристика поведения модели на тестовых выборках (включая калибровку и стабильность ответов), r(M) — характеристика робастности модели (устойчивость к возмущениям и к смещениям распределений). Функция F агрегирует эти три величины в единую оценку. Калибровочная мера q(M) определяется через ожидаемую калибровочную ошибку (expected calibration error, ECE):

$$q(M) = 1 - ECE(M)$$

где ECE вычисляется стандартным образом — разбиением предсказаний модели на M бинов по предсказанной вероятности и

сравнением накопленной в каждом бине доли правильных предсказаний с предсказанной вероятностью:

$$ECE(M) = \sum_{m=1}^{M} \frac{|B_m|}{n} |acc(B_m) - conf(B_m)|$$

Робастность r(M) формализуется как вероятность сохранения предсказания при ограниченных по норме возмущениях входа из заданного класса, например ε-окрестности по l∞:

$$r(M) = P_{x \sim D}(M(x + \delta) = M(x), |\delta| \leq \varepsilon)$$

Конкретный вид функции F зависит от области применения. В качестве базовой формы предлагается взвешенное геометрическое среднее, обладающее свойствами монотонности по каждому аргументу, ограниченности отрезком [0, 1] и устойчивости к экстремальным значениям отдельных составляющих:

$$T(M) = t(D_{train})^{\alpha_1} \cdot q(M)^{\alpha_2} \cdot r(M)^{\alpha_3}$$

при условии

$$\alpha_1 + \alpha_2 + \alpha_3 = 1, \ \alpha_k \geq 0$$

Веса α_1, α_2, α_3 отражают относительную значимость источника, поведения и устойчивости для конкретного класса задач. Для критических приложений, в которых ключевым является поведение модели в редких или сложных случаях, целесообразно увеличивать α_2 и α_3; для приложений, в которых ключевым является происхождение обучающих данных (например, в исследованиях, опирающихся на открытые корпуса), — α_1.

2.5.4. Агрегация доверенности по графу провенанса данных

Первый сомножитель в декомпозиции, доверенность обучающих данных t(D_train), сам по себе является агрегированной характеристикой, в значительной степени определяемой структурой провенанса. В стандарте PROV-DM провенанс представляется в виде направленного ациклического графа

$$G = (V, E, \tau)$$

вершинами которого V являются сущности (datasets), активности (преобразования данных) и агенты (организации, конвейеры обработки), а рёбрами E — отношения wasDerivedFrom, wasGeneratedBy, used, wasAttributedTo и им подобные. Функция τ: V ∪ E → [0, 1] сопоставляет каждой вершине и каждому ребру значение

доверия: вершинам — априорное доверие к источнику или активности, рёбрам — коэффициент сохранности доверия при соответствующем преобразовании. Для оценки доверенности набора D, представленного вершиной v_D в графе провенанса, используется агрегация значений вдоль путей π = (v_0, e_1, v_1, ..., e_n, v_n), ведущих от первичных источников v_0 к рассматриваемой вершине v_n = v_D.

Доверенность вдоль одного пути в провенансе формализуется как произведение доверия к исходному источнику и коэффициентов передачи доверия вдоль рёбер пути:

$$t(\pi) = t(v_0) \cdot \prod_{k=1} \tau(e_k)$$

Такая мультипликативная форма согласуется с поведением, наблюдаемым в моделях передачи доверия в сетях источников: каждая дополнительная активность преобразования способна снижать доверенность пропорционально собственному коэффициенту, а независимость рёбер обеспечивает корректность произведения. При наличии нескольких путей Π(v) = {π_1, π_2, ..., π_m}, ведущих к одной вершине, доверенность вершины определяется агрегацией по этим путям. В качестве канонической формы агрегации применяется вероятностная дизъюнкция, согласующаяся с интерпретацией пути как независимого источника подтверждения:

$$t(v) = 1 - \prod_{\pi \in \Pi(v)} \big(1 - t(\pi)\big)$$

Указанная форма обладает свойствами, делающими её естественной для задач провенанса: добавление нового подтверждающего пути не уменьшает доверенности, а в пределе бесконечного числа независимых путей с положительной доверенностью значение t(v) стремится к единице. Альтернативной формой агрегации, применимой в случаях, когда независимость путей не обеспечивается, является взвешенное среднее доверенностей путей, в котором веса определяются длиной пути или априорной значимостью соответствующих источников.

В тех случаях, когда граф провенанса допускает циклы взаимных подтверждений или содержит большое число тесно связанных источников, целесообразно использовать итерационную

процедуру оценки, известную в литературе как процедура ДеГрута. Она задаётся в виде матричной итерации над вектором доверенностей вершин:

$$t^{(k+1)} = W \cdot t^{(k)}$$

где W — строчно-стохастическая матрица весов взаимных подтверждений между вершинами:

$$\sum_j w_{ij} = 1,\ w_{ij} \geq 0$$

При выполнении условий неприводимости и апериодичности графа взаимных подтверждений последовательность t^(k) сходится к единственному стационарному вектору t*, не зависящему от начального приближения. Этот стационарный вектор интерпретируется как согласованная оценка доверенности всех узлов графа с учётом их взаимных подтверждений. Полученное значение t(v_D) = t*_D в дальнейшем используется как первый сомножитель в декомпозиции T(M).

При обучении модели на нескольких подмножествах D_1, D_2, …, D_N, каждое из которых характеризуется собственной доверенностью t(D_i), интегральная доверенность обучающей выборки вычисляется как взвешенное среднее с весами, пропорциональными вкладу каждого подмножества в обучение (например, числу примеров или числу шагов градиентного спуска, выполненных на данном подмножестве):

$$t(D_{train}) = \sum_{i=1}^{N} \omega_i t(D_i)$$

Указанная аддитивная агрегация применима в случаях, когда подмножества D_i рассматриваются как независимые и сопоставимые по природе. В случаях, когда подмножества порождаются последовательным комплексированием, более содержательной является марковская модель, развитая в разделах 3–5 настоящей главы: значение t(D_train) определяется по матрице переходов марковской цепи комплексирования, и приведённая аддитивная форма соответствует одношаговой аппроксимации этой динамики.

2.5.5. Марковское описание динамики доверия к модели

Развитие модели на динамику повторяет схему предыдущих разделов с учётом особенностей моделей. В случае модели M, проходящей через последовательность модификаций, каждая модификация — дообучение, тонкая настройка, дистилляция — играет роль шага марковской цепи. Текущая модель M_n обладает доверенностью T(M_n), которая после очередного шага обновляется по правилу

T(M_{n+1}) = Φ(T(M_n), t(B_n), α(M_n, B_n)),

где Φ — функция обновления доверенности модели, аналогичная функции обновления доверенности данных, B_n — пакет данных, использованных на шаге дообучения, а α — параметр согласованности, отражающий, насколько новые данные согласуются с текущим поведением модели. Аксиомы, которым должна удовлетворять Φ, наследуют аксиомы доверенности данных: дообучение на качественных и согласованных с моделью данных не должно уменьшать T(M); дообучение на данных, противоречащих текущему поведению модели, ведёт к снижению T(M); многократное дообучение на одних и тех же данных приводит к насыщению, не достигающему максимальной отметки. Свойства Φ, аналогичные свойствам функции изменения доверенности данных, обеспечивают преемственность аппарата.

2.5.6. Типовые сценарии для моделей

По аналогии со сценариями предыдущего параграфа для моделей формулируется ряд типовых сценариев.

Сценарий стабильного дообучения: модель регулярно дообучается на новых данных, согласованных с её текущим поведением; T(M) сохраняется на высоком уровне или плавно растёт.

Сценарий последовательной дистилляции: исходная модель M_0 дистиллируется в более компактную M_1, та — в M_2, и так далее; ожидается монотонное уменьшение T(M) с каждым шагом, что соответствует известному в литературе эффекту накопления ошибок при многократной дистилляции [41].

Сценарий fine-tuning на смещённой выборке: модель, обученная на данных одной области, дообучается на данных другой

области; T(M) убывает до промежуточного значения, зависящего от степени смещения.

Сценарий объединения моделей (ensemble или federated averaging): несколько моделей объединяются в одну; T(M_ens) определяется не только средней доверенностью входящих моделей, но и согласованностью их предсказаний — аналог параметра sim для данных.

2.5.7. Численный пример: оценка доверия к классификатору

В качестве иллюстрации применения предложенного аппарата рассмотрим оценку доверия к классификатору рентгенограмм грудной клетки, обученному распознавать наличие пневмонии. Указанный сценарий характерен для медицинской практики, в которой модель проходит сертификацию, развёртывается в клинических учреждениях и периодически дообучается на новых данных. Пусть модель M обучена на объединённой обучающей выборке D_train, состоящей из трёх подмножеств: D_1 — публичный корпус ChestX-ray из 30 тыс. снимков с проверенной экспертом разметкой, D_2 — внутренний корпус крупной клиники из 12 тыс. снимков с автоматизированной разметкой по результатам обследования, D_3 — открытый корпус из 8 тыс. снимков с разметкой, полученной краудсорсингом. Веса вкладов в обучение, оцениваемые по числу шагов градиентного спуска на каждом подмножестве: $\omega_1 = 0{,}6$; $\omega_2 = 0{,}3$; $\omega_3 = 0{,}1$.

Оценка доверенности подмножеств производится по графу провенанса. Для D_1 путь от первичного источника содержит сертифицированную клиническую систему регистрации и экспертную проверку: $t(v_0) = 0{,}95$, коэффициенты передачи доверия по двум рёбрам — 0,95 и 0,95. Доверенность подмножества по единственному пути: $t(D_1) = 0{,}95 \cdot 0{,}95 \cdot 0{,}95 \approx 0{,}86$. Для D_2 путь содержит автоматизированную разметку: $t(v_0) = 0{,}90$, коэффициенты — 0,90 и 0,85, что даёт $t(D_2) = 0{,}90 \cdot 0{,}90 \cdot 0{,}85 \approx 0{,}69$. Для D_3 путь содержит краудсорсинговую разметку с дополнительной фильтрацией: $t(v_0) = 0{,}80$, коэффициенты — 0,85 и 0,80, что даёт $t(D_3) = 0{,}80 \cdot 0{,}85 \cdot 0{,}80 \approx 0{,}54$. Интегральная доверенность обучающего набора по формуле взвешенного среднего:

t(D_train) = 0,6 · 0,86 + 0,3 · 0,69 + 0,1 · 0,54 = 0,516 + 0,207 + 0,054 ≈ 0,77.

Калибровка модели измеряется на удерживаемой тестовой выборке из 5 тыс. снимков. Разбиение предсказаний на десять бинов по предсказанной вероятности и сравнение средней предсказанной вероятности с фактической долей положительных классов даёт ECE(M) = 0,08, откуда q(M) = 1 – 0,08 = 0,92. Робастность измеряется как доля сохранённых предсказаний при добавлении гауссовского шума с σ = 0,01 и стандартных аугментаций (поворот ±5°, сдвиг ±5%): r(M) = 0,87. Веса для агрегации, отражающие приоритеты медицинской сертификации: α_1 = 0,5 (доверие к обучающим данным является ключевым), α_2 = 0,3 (калибровка существенна для интерпретации вероятностей врачом), α_3 = 0,2 (робастность к шумам приборов — необходимый, но менее критический фактор). Подставляя значения в формулу взвешенного геометрического среднего, получаем

$$T(M) = 0{,}77^{0{,}5} \cdot 0{,}92^{0{,}3} \cdot 0{,}87^{0{,}2} \approx 0{,}84$$

Полученное значение T(M) ≈ 0,84 интерпретируется как количественная оценка доверия к модели в условиях, для которых она разрабатывалась. Это значение допускает прямое сопоставление с порогами, устанавливаемыми политикой сертификации: для допуска к клиническому использованию в качестве вспомогательного инструмента в литературе и регуляторных проектах обсуждается порог в диапазоне 0,80–0,85, для использования в качестве самостоятельного диагностического средства — порог 0,90 и выше. Согласно полученной оценке, модель M соответствует первому диапазону и не соответствует второму, что задаёт чёткую границу её допустимого применения.

Рассмотрим теперь динамику T(M) при типовых операциях. Пусть модель проходит дообучение на смещённой выборке: данные из новой клиники, в которой используется иной протокол съёмки, не вполне согласованные с исходным распределением (параметр согласованности α = 0,5). Аналог формулы изменения доверенности данных, применённый к доверенности модели, даёт после одного шага дообучения t(D_train) ≈ 0,71 (агрегированная доверенность пополненной обучающей выборки), а измерения на тестовой выборке после дообучения показывают q(M′) = 0,88 и r(M′) = 0,84. Итоговая доверенность модели после дообучения:

$$T'(M') = 0{,}71^{0{,}5} \cdot 0{,}88^{0{,}3} \cdot 0{,}84^{0{,}2} \approx 0{,}80$$

Снижение T(M) с 0,84 до 0,80 находится на границе допустимого диапазона и сигнализирует о необходимости повторной валидации модели. При дальнейшем дообучении на дополнительных смещённых данных значение T(M) может опуститься ниже допустимого порога, и модель должна быть выведена из эксплуатации до проведения дополнительной верификации. Аналогичные расчёты могут быть произведены для сценария дистилляции: дистилляция в более компактную модель M_student с тем же D_train, но с собственным q и r — обычно более низкими, — даёт T(M_student) ≈ 0,79, что согласуется с известным эффектом снижения доверия при сжатии моделей.

Сводная характеристика рассмотренных сценариев приведена в форме нижеследующего перечисления: исходная модель M обладает T(M) ≈ 0,84 при значениях t(D_train) ≈ 0,77, q(M) = 0,92, r(M) = 0,87; после дообучения на смещённой выборке T(M′) ≈ 0,80 при t(D_train) ≈ 0,71, q(M′) = 0,88, r(M′) = 0,84; после дистилляции T(M_student) ≈ 0,79 при t(D_train) ≈ 0,77, q(M_student) = 0,86, r(M_student) = 0,80. Сопоставление этих значений с порогами сертификации позволяет в количественной форме принимать решения о допуске, о необходимости повторной валидации и о выводе модели из эксплуатации.

Подчеркнём, что приведённые числовые значения не претендуют на роль эмпирически подтверждённых констант — их роль состоит в иллюстрации применения формального аппарата на содержательном примере. Калибровка конкретных коэффициентов τ, ω, α требует отдельных исследований на репрезентативных выборках моделей и сопоставления с экспертными оценками практиков. Тем не менее структура расчёта остаётся неизменной, что обеспечивает воспроизводимость подхода и его пригодность для встраивания в автоматизированные процедуры сертификации.

2.5.8. Связь с задачами безопасности и сертификации

Формализация доверия к отчуждённым обученным моделям непосредственно связана с задачами безопасности и сертификации систем искусственного интеллекта. В отраслях с высоким уровнем

регулирования — в медицине, в финансах, в государственном управлении, в авиации — модели не могут быть приняты в эксплуатацию без процедуры сертификации, в ходе которой оценивается их пригодность к работе в заявленной области применения. Существующие процедуры сертификации опираются преимущественно на тестовые метрики и на качественные оценки экспертов; они недостаточно формализованы для полной автоматизации. Введение количественной меры T(M), вычисляемой по приведённой формуле и проверяемой против установленного порога, создаёт основу для автоматизации части процедур и установления формальных порогов доверия, обязательных для допуска модели к эксплуатации.

В контексте информационной безопасности модель доверия к отчуждённым обученным моделям имеет ещё одно приложение — детектирование подменённых, изменённых или скомпрометированных моделей. Если порядок изменения T(M) при типовых модификациях известен, отклонение фактически наблюдаемой динамики от ожидаемой служит индикатором несанкционированного вмешательства. В предельном случае, при наличии цифровой подписи на модели и при сохранении истории её модификаций, можно установить факт подмены модели вредоносной копией. Эта область — защищённое распространение моделей, реализуемое в проектах OpenSSF Model Signing и аналогичных инициативах, — активно развивается, и количественный аппарат, развиваемый в настоящей работе, может послужить её математическим основанием.

2.5.9. Открытые вопросы и направления исследования

Развитие очерченной теории сталкивается с рядом открытых вопросов, каждый из которых может составить отдельный сюжет диссертационного исследования. Первый — выбор минимального представления состояния модели, при котором сохраняется марковское свойство. Числовое представление через T(M) удобно, но может оказаться недостаточным: для адекватного предсказания результата следующего шага может потребоваться дополнительная информация о модели — её внутренняя структура, распределение её ошибок, спектр Гессиана функции потерь. Поиск компромисса

между богатством представления и его компактностью представляет самостоятельную задачу.

Второй вопрос — выбор аксиоматики функции Ф для моделей. В случае данных аксиоматика опиралась на интуитивно ясные требования к операции объединения; в случае моделей перечень возможных операций существенно шире, и для каждой требуется собственный набор аксиом. Возможны как универсальные аксиомы, общие для всех типов операций, так и специализированные, привязанные к конкретным процедурам — дистилляции, federated averaging, knowledge transfer, prompt tuning, low-rank adaptation. Поиск минимального самостоятельного набора аксиом, охватывающего наиболее значимые операции, — открытая исследовательская задача.

Третий вопрос — связь T(M) с вероятностью корректного поведения модели в практической эксплуатации. В идеальном случае требуется установить функциональную связь между T(M) и вероятностью того, что модель не нарушит требования заказчика в условиях эксплуатации. Эта связь, по-видимому, не является функциональной в строгом смысле, но может быть выражена через стохастические ограничения, аналогичные PAC-байесовским границам. Установление таких связей сделает T(M) непосредственно интерпретируемым показателем риска и расширит его прикладную ценность.

Четвёртый вопрос — масштабируемость подхода на современные большие модели. Foundation models содержат миллиарды параметров и обучаются на терабайтах данных. Прямое применение развиваемой теории к ним требует решения вычислительных вопросов: агрегация доверенности обучающих корпусов, состоящих из тысяч источников; учёт многоэтапного цикла pretrain–finetune–RLHF; описание дообучения через подсказки и адаптеры. Каждый из этих вопросов является предметом активных исследований.

Пятый вопрос — перенос аппарата на уровень отдельных предсказаний модели. После того как модель развёрнута, она производит выводы по входящим запросам, и пользователю важно понимать, насколько каждому конкретному выводу можно доверять. Это перекликается с задачей оценки уверенности модели в её

собственных предсказаниях (Bayesian-методы, ensembles, conformal prediction), но допускает и более широкие формулировки, учитывающие соответствие конкретного запроса распределению обучающих данных. Перенос аппарата доверенности на уровень отдельных предсказаний открывает дополнительный аспект исследования, не сводимый к оценке модели в целом.

2.5.10. Заключительные замечания

Оценка доверия к отчуждённым обученным моделям существенно расширяет область применения представленной в этой главе теории: от анализа набора данных как объекта обработки к анализу обученной модели как самостоятельной сущности, способной к собственной эволюции и взаимодействию с другими моделями. Этот переход является принципиальным шагом на пути построения целостной теории доверия в системах искусственного интеллекта.

В рамках разработки этого направления предполагается: разработать аксиоматику функции обновления доверия к модели; выписать явные формулы для типовых операций; провести вычислительные эксперименты на представительных моделях; сопоставить полученные оценки с эмпирически наблюдаемыми показателями надёжности; разработать методологические рекомендации по применению меры T(M) в задачах сертификации и безопасности; реализовать прототип библиотеки, обеспечивающей вычисление T(M) на стандартных моделях машинного обучения.

Если доверенность набора данных есть состояние системы, эволюционирующей при операциях с данными, то доверенность модели — состояние более сложной системы, эволюционирующей при операциях с моделями. Перенос аппарата марковских процессов на случай моделей не является механическим, но направление этого переноса ясно, и формализация представляется выполнимой задачей. Реализация такой программы способна продвинуть область доверенного искусственного интеллекта от описательных и качественных оценок к строгому количественному анализу, открывая путь к новым стандартам сертификации, новым формам управления жизненным циклом моделей и новым средствам обеспечения безопасности систем.

2.6. ЗАКЛЮЧЕНИЕ

Полученные результаты позволяют сформулировать ряд практических рекомендаций.

Во-первых, в системах, в которых данные поступают из множества источников, целесообразно проектировать политики допуска данных с учётом не только текущей доверенности каждого источника, но и ожидаемого среднего уровня схожести его данных с уже накопленным набором: именно этот средний уровень определяет асимптотическое поведение системы.

Во-вторых, в системах, в которых модели проходят через цепочку модификаций, целесообразно вести количественный учёт ожидаемого изменения доверия и устанавливать формальные пороги, при достижении которых требуется повторная сертификация модели.

В-третьих, в задачах распределённого обучения, в которых данные не покидают своих локальных хранилищ, полезно вести агрегированный учёт доверенности на уровне распределённых вкладов в общий процесс обучения, что позволяет принимать обоснованные решения о допуске того или иного участника к взаимодействию.

Основной вывод главы заключается в том, что переход от формального изучения информационных процессов к формальному изучению самих данных и обученных моделей требует введения количественных характеристик, изменяющихся при операциях с этими объектами по содержательно обоснованным правилам. Доверенность – одна из таких характеристик, и аппарат марковских процессов оказывается адекватным языком описания для описания её динамики. Развитие этого языка применительно к более широкому кругу объектов представляется одним из перспективных направлений исследования в области доверенного искусственного интеллекта. В рамках этого направления материал главы может рассматриваться как первый шаг, демонстрирующий применимость аппарата марковских цепей к задаче количественной оценки доверия и открывающий путь к его обобщениям.

Глава 3. ОЦЕНКА ПАРАМЕТРОВ СИСТЕМ СЛЕПОЙ ОБРАБОТКИ ДАННЫХ, БЛОКИРУЮЩИХ КОСВЕННЫЕ УТЕЧКИ

П. А. Галманов

Системы «слепой» обработки данных строятся так, чтобы пользователь или внешний компонент не получали прямого доступа к защищённым записям. Взаимодействие с такими системами переносится на уровень заранее заданных ИТ-конвейеров: пользователь запускает разрешённую последовательность операций и наблюдает только её выходы — отчёты, агрегированные показатели, метрики качества, ответы модели или их постобработанные варианты [42].

Ограничение прямого доступа не устраняет возможность косвенного восстановления. Если выходы конвейера многократно наблюдаются, сравниваются между собой или дополняются открытыми признаками, они могут содержать достаточно информации для восстановления защищаемого скалярного атрибута $t = x_i$. Такая реконструкция не требует чтения исходной записи: она возникает как обратная задача, в которой по накопленным наблюдаемым выходам восстанавливается скрытая компонента входных данных.

Далее ИТ-конвейер обозначается через F, наблюдаемый ответ — через y, а полный наблюдаемый вектор — через $z = (y, x_{\text{obs}})$ или, при явном учёте шума ответа, через $z = (y + \varepsilon, x_{\text{obs}})$. Истинная функция утечки обозначается через $\varphi^{\dagger}$, а её регуляризованная оценка — через $\hat{\varphi}_{\alpha}$.

3.1. ПОСТАНОВКА ПРОБЛЕМЫ И ИНЖЕНЕРНАЯ МОТИВАЦИЯ

Прямая утечка возникает тогда, когда защищаемые данные становятся доступны в явном виде: пользователь читает исходные записи, получает выгрузку персональных атрибутов, извлекает идентификаторы из журналов или копирует фрагменты базы. Для такого класса угроз применяются классические меры технической защиты: разграничение доступа, криптографическая защита каналов и хранилищ, контроль целостности, аудит действий и изоляция вычислительного контура.

Косвенная утечка имеет другую природу. Пользователь может не видеть ни одной исходной записи, но наблюдать результаты обработки, из которых восстанавливается защищаемый атрибут или его информативная характеристика. В этом случае раскрывающим объектом становится совокупность выходов системы: агрегаты, отчёты, промежуточные и итоговые метрики, параметры модели, вероятностные оценки, скоринговые ответы или статистики по выбранным группам.

Агрегированный ответ не является нейтральным только потому, что он не содержит персонального идентификатора. Среднее значение, доля события, максимум, квантиль или значение метрики могут изменяться при добавлении или исключении одного субъекта. Если такие изменения доступны для наблюдения, то последовательность близких запросов способна раскрыть вклад отдельного атрибута; близкая по смыслу проблема восстановления информации из статистических ответов обсуждается в работах по атакующим наборам запросов и дифференциальной конфиденциальности [43, 44]. В задачах машинного обучения аналогичный эффект возникает, когда ответы модели или отчёты о её качестве сохраняют функциональную зависимость от защищаемых атрибутов обучающих или обслуживаемых объектов [45].

В системах слепой обработки данных пользователь обычно взаимодействует не с произвольной базой запросов, а с разрешёнными ИТ-конвейерами [42]. Такой конвейер фиксирует допустимую последовательность операций: подготовку данных, фильтрацию, агрегирование, обучение модели, расчёт метрик,

применение модели и постобработку результата. Поэтому анализ утечки должен относиться не к абстрактному одиночному запросу, а к конкретному конвейеру F и множеству его наблюдаемых выходов.

Фиксация конвейера уменьшает произвольность взаимодействия с системой, но не делает вопрос тривиальным. Даже разрешённая цепочка операций может порождать выходы, которые при накоплении наблюдений становятся информативными относительно защищаемого атрибута. Более того, несколько слабых выходных каналов могут совместно давать более сильную реконструкцию, чем каждый из них по отдельности, особенно если наблюдатель знает открытые или управляющие признаки, сопровождающие запрос.

Для администратора системы и владельца процесса недостаточно бинарного ответа «утечка есть» или «утечки нет». Практически значимы вопросы о масштабе и режиме проявления зависимости: при каком числе наблюдений она становится статистически обнаружимой, как на неё влияет уровень шума, какие компоненты ответа дают основной вклад и насколько меняется риск при удалении или модификации отдельных выходов.

Количественный анализ позволяет перейти от общего запрета к управлению параметрами эксплуатации. Если известно, что при заданном наборе выходов и уровне шума восстановление не достигает заданного уровня обнаруживаемости до некоторого числа наблюдений, можно выбирать лимиты запросов, интервалы обновления отчётов, степень агрегирования и величину добавляемого шума. Такие решения должны формулироваться относительно явно указанных предположений о данных, модели наблюдения и классе функций восстановления.

Порог обнаруживаемости не является абсолютной гарантией отсутствия восстановления. Он означает лишь, что при выбранных критериях качества и статистической значимости зависимость становится обнаружимой начиная с определённого объёма данных. При меньшем объёме риск может оставаться ненулевым: удачная реализация шума, дополнительная внешняя информация или коррелированные наблюдения способны улучшить восстановление раньше расчётного порога.

Поэтому инженерная интерпретация порога должна быть вероятностной. Число наблюдений, уровень шума и набор выходов задают область, в которой риск успешного восстановления удерживается ниже заданного уровня при принятых предположениях. Если предположения меняются, количественные оценки должны пересчитываться.

Естественный математический язык для такой постановки — регуляризованное восстановление функции утечки. Наблюдаемые выходы конвейера рассматриваются как данные обратной задачи, защищаемый атрибут — как скрытая компонента входа, а регуляризация используется для отделения устойчивой зависимости от случайной подгонки конечной и шумной выборки.

3.2. ФОРМАЛЬНАЯ МОДЕЛЬ ИТ-КОНВЕЙЕРА И НАБЛЮДЕНИЙ

Пусть

$$x = (t, x_{\text{rest}}) \in X_t \times X_{\text{rest}}$$

— вектор защищённых данных. Здесь

$$t = x_i$$

— защищаемый скалярный атрибут, относительно которого анализируется возможность восстановления, а x_{rest} обозначает остальные компоненты записи или набора записей. Отдельно выделим открытые или управляющие признаки

$$x_{\text{obs}} \in X_{\text{obs}},$$

которые не являются защищаемым атрибутом t, но могут быть известны наблюдателю. К ним относятся параметры сценария, публичные признаки, сведения о группе запроса, настройки отчёта, временные метки или другие контекстные величины. Наличие x_{obs} может усиливать восстановление, поскольку одна и та же величина y может быть по-разному информативна при разных открытых условиях.

ИТ-конвейер зададим отображением [42, 46]

$$F: X_t \times X_{\text{rest}} \times X_{\text{obs}} \to Y, \qquad y = F(t, x_{\text{rest}}, x_{\text{obs}}),$$

где Y — пространство наблюдаемых ответов. Отображение F может включать подготовку данных, фильтрацию записей, агрегирование, обучение модели, расчёт метрик, применение модели к новым объектам и постобработку результата.

Наблюдаемый ответ y может быть скаляром, например одним агрегированным показателем, или вектором отчётных величин, метрик и модельных выходов.

Если ответ публикуется с шумом или измеряется с погрешностью, будем писать

$$y_\varepsilon = y + \varepsilon,$$

где ε обозначает шумовую компоненту. Полный наблюдаемый вектор, используемый при восстановлении, имеет вид

$$z = (y, x_{\text{obs}}) \quad \text{или} \quad z = (y + \varepsilon, x_{\text{obs}}).$$

Тем самым пространство аргументов функции восстановления содержит не только выход конвейера, но и открытый контекст его получения.

Функция утечки описывает зависимость защищаемого атрибута от наблюдаемого вектора:

$$\varphi^\dagger : Y \times X_{\text{obs}} \to \mathbb{R}, \qquad t \approx \varphi^\dagger(z).$$

Функция $\varphi^\dagger$ не предполагается известной наблюдателю в аналитическом виде. Она выражает ту зависимость, которую можно пытаться восстановить по данным, если выходы конвейера содержат информацию о t.

Для оценки риска используется контрольная, аудиторская или модельная выборка размера N [46]:

$$\mathcal{D}_N = \{(z^{(k)}, t^{(k)})\}_{k=1}^N, \qquad z^{(k)} = \left(y^{(k)}, x_{\text{obs}}^{(k)}\right),$$

где

$$y^{(k)} = F\left(t^{(k)}, x_{\text{rest}}^{(k)}, x_{\text{obs}}^{(k)}\right)$$

или соответствующее зашумлённое значение. Такая выборка не является эксплуатационным доступом внешнего пользователя к персональным данным. Она используется в контролируемом режиме для оценки того, насколько наблюдаемые выходы конвейера позволяют восстанавливать защищаемый атрибут.

Косвенная утечка в этой модели означает, что по z существует восстановление защищаемого атрибута с заданной точностью. Формально это можно записать как существование алгоритма R или функции φ, для которых на существенной области входов выполняется оценка

$$|R(z) - t| \le \varepsilon_R \quad \text{или} \quad |\varphi(z) - t| \le \varepsilon_R,$$

где $\varepsilon_R > 0$ — допустимый уровень ошибки восстановления. Для практического вывода важна не только сама возможность

аппроксимации, но и её устойчивость при шуме, конечной выборке, ограниченном классе моделей и выбранной процедуре проверки значимости.

3.3. МЕСТО ПОДХОДА СРЕДИ МЕТОДОВ АНАЛИЗА КОСВЕННЫХ УТЕЧЕК

3.3.1. Качественная проверка возможности утечки

Первый уровень анализа косвенной утечки отвечает на вопрос структурной возможности: можно ли в принципе восстановить защищаемый атрибут t из тех величин, которые становятся наблюдаемыми после выполнения конвейера F. На этом уровне не требуется сразу оценивать вероятность атаки или строить статистический критерий. Достаточно понять, содержит ли набор выходов такую функциональную зависимость, которая делает скрытую компоненту входных данных локально определимой.

В простых гладких моделях этот вопрос естественно связывается с ранговой логикой. Если наблюдаемые ответы зависят от скрытых переменных дифференцируемым образом, то локальная возможность восстановления проверяется через ранг соответствующих матриц чувствительности. При полном ранге по интересующей компоненте малые изменения защищаемого атрибута оставляют различимый след в выходах; при недостаточном ранге разные значения скрытой компоненты могут давать одинаковые наблюдения, и локальное восстановление становится неопределённым.

Такая проверка важна как предварительный фильтр. Она позволяет увидеть, какие выходы вообще несут структурную информацию о t, какие признаки могут быть избыточными, а какие агрегаты почти не меняются при изменении защищаемого атрибута. На человеческом языке это соответствует вопросу о том, есть ли у разрешённого конвейера наблюдаемые степени свободы, через которые защищаемая величина может проявиться.

Однако ранговая или якобианная логика не даёт полного ответа о риске эксплуатации. Она обычно описывает локальную возможность восстановления в идеализированной модели и не

отвечает на вопросы о конечном числе наблюдений, уровне шума, устойчивости к изменению выборки, статистической значимости и вероятности раннего успеха. Иными словами, структурная возможность ещё не означает, что восстановление будет воспроизводимо обнаружено в контрольном эксперименте или достигнет заданной вероятности при ограниченном бюджете наблюдений.

Поэтому качественная проверка полезна как начало анализа. Она показывает, на какие компоненты конвейера следует обратить внимание, а последующая регуляризованная диагностика оценивает, насколько эта возможность проявляется на данных и при выбранном режиме наблюдения.

3.3.2. Дифференциальная конфиденциальность и её роль

Другой важный класс методов связан с дифференциальной конфиденциальностью. В этой постановке проектируется механизм выдачи ответов, для которого влияние одной записи на распределение публикуемого результата ограничено заранее заданным параметром. Такой подход даёт строгий механизмный язык контроля раскрытия и особенно полезен при проектировании статистических запросов, публикации агрегатов и обучении моделей с контролируемым вкладом отдельных объектов [43, 44].

В терминах настоящей главы дифференциальная конфиденциальность отвечает прежде всего на вопрос о построении или модификации механизма ответа. Она может подсказывать, какой шум добавить, как ограничить чувствительность агрегата и как вести бюджет приватности при повторных обращениях. Это мощный стандартный подход, и он не конкурирует с регуляризованной диагностикой как с заменой.

Акцент здесь иной. Рассматривается уже заданный ИТ-конвейер F, который может включать фильтрацию, агрегирование, обучение модели, вычисление метрик и постобработку. Для такого конвейера требуется оценить, позволяют ли его фактические наблюдаемые выходы восстанавливать защищаемый атрибут t при доступном числе наблюдений и принятом уровне шума. Поэтому задача формулируется как диагностика существующего канала

наблюдения, а не как построение нового механизма с заранее доказанной приватностью.

Эти подходы являются дополнительными. Дифференциальная конфиденциальность задаёт принципы проектирования ответов с контролируемым влиянием одной записи, а регуляризованное восстановление оценивает, что происходит в конкретной реализованной процедуре и насколько её выходы информативны относительно защищаемого атрибута. В практической системе первый подход может использоваться при проектировании конвейера, а второй — при аудите и настройке эксплуатационных ограничений.

3.3.3. Регуляризованное восстановление как диагностический инструмент

Регуляризованная постановка переносит вопрос о косвенной утечке из области качественного описания в область измеряемых характеристик. Вместо утверждения «зависимость возможна» рассматривается прикладной вопрос: при каком числе наблюдений, при каком уровне шума и при каком наборе наблюдаемых выходов восстановление становится статистически обнаружимым.

Для этого строится оценка функции утечки $\hat{\varphi}_\alpha$, связывающая полный наблюдаемый вектор $z = (y, x_{\text{obs}})$ с защищаемым атрибутом t. Регуляризация ограничивает сложность функции, кросс-валидация проверяет воспроизводимость качества на новых наблюдениях, а перестановочный тест отделяет зависимость от случайной подгонки конечной выборки. Именно эта связка — оценка шума, регуляризация, $R^2_{\text{CV}}(N)$, $p_{\text{perm}}(N)$ и порог N^*_{det} — используется как диагностический контур оценки риска [47, 48].

При этом статистическая обнаруживаемость не исчерпывает риск. До достижения порога N^*_{det} отдельное восстановление может оказаться успешным с ненулевой вероятностью, особенно если шум благоприятен, наблюдения повторяются или внешний пользователь располагает дополнительной открытой информацией. Поэтому вместе с порогом обнаруживаемости рассматривается функция $p_A(N)$ и вероятностный порог $N_A(p_{\text{crit}})$, описывающие риск раннего восстановления в локальной модели накопления информации [49].

Итоговый язык оказывается удобным для администратора системы. Он переводит вопрос о косвенной утечке в параметры управления: допустимый бюджет наблюдений, величину шума, набор публикуемых выходов, частоту обновления отчётов и запас относительно пороговых значений.

3.4. ФУНКЦИЯ УТЕЧКИ КАК РЕГУЛЯРИЗОВАННАЯ ОБРАТНАЯ ЗАДАЧА

Положим

$$S = Y \times X_{\text{obs}}.$$

Истинная функция утечки действует как

$$\varphi^{\dagger}: S \to \mathbb{R}, \qquad t \approx \varphi^{\dagger}(z).$$

Восстановление $\varphi^{\dagger}$ по конечной выборке $\mathcal{D}_N$ является обратной задачей: по наблюдаемым выходам конвейера и открытому контексту требуется восстановить скрытую компоненту входных данных.

Прямое направление задаётся конвейером F: из защищённых данных, фоновых компонент и открытого контекста строится наблюдаемый ответ y. Диагностическая процедура или процедура восстановления движется в противоположном направлении. Она получает y и x_{obs}, объединяет их в z и пытается восстановить t. Это не обращение F в строгом алгебраическом смысле, поскольку F может терять информацию, агрегировать множество записей, выполнять обучение модели или публиковать только метрики. Тем не менее по смыслу задача обратная: скрытая причина оценивается по наблюдаемому следствию.

Потеря информации делает задачу неустойчивой. Один и тот же ответ может соответствовать многим значениям защищаемого атрибута, а малое изменение шума в y способно заметно изменить восстановленное значение t. Если конвейер выдаёт агрегат, то вклад отдельной записи обычно сжат в малую разность между близкими ответами. Если конвейер выдаёт метрику модели, то зависимость может проходить через обучение, валидацию и постобработку. В обоих случаях прямой выход не содержит явного значения t, но

может сохранять достаточно информации для статистического восстановления.

В этой постановке присутствуют три уровня неизвестности. Во-первых, неизвестен сам защищаемый атрибут t, который требуется восстановить или риск восстановления которого оценивается. Во-вторых, неизвестна функция утечки $\varphi^{\dagger}$: даже если зависимость существует, её аналитическая форма обычно не задана. В-третьих, неизвестно, будет ли найденная зависимость устойчивой на новых наблюдениях, а не только на той контрольной выборке, по которой она была получена.

Именно третий уровень отличает практический аудит от простой аппроксимации. Можно построить модель, которая почти идеально согласуется с конечной выборкой, но окажется чувствительной к разбиению, шуму или удалению нескольких наблюдений. Для анализа утечки такой результат недостаточен: он может создавать ложное впечатление риска там, где модель воспроизвела случайную структуру данных. Обратная ситуация также возможна: слишком грубый класс функций или чрезмерно сильный штраф могут скрыть слабую, но устойчивую зависимость.

Эта задача некорректна в смысле Адамара в нескольких отношениях. Наблюдаемые ответы могут содержать шум; выборка конечна и не покрывает всё пространство сценариев; часть переменных, влияющих на ответ, остаётся ненаблюдаемой; компоненты выходного вектора могут быть сильно коррелированы; размерность z может быть высокой; сами наблюдения могут собираться адаптивно, когда последующие запросы зависят от предыдущих ответов. В таких условиях простая минимизация ошибки на обучающей выборке может приводить к неустойчивой функции, хорошо подгоняющей конечные данные, но плохо отражающей воспроизводимую зависимость [51–52].

Регуляризация вводит явный контроль сложности функции восстановления. Пусть $\mathcal{H}$ — выбранное функциональное пространство, ℓ — функция потерь, а $\mathcal{R}$ — функционал штрафа. Тогда регуляризованный функционал имеет вид

$$J_{\alpha}(\varphi) = \frac{1}{N}\sum_{k=1}^{N} \ell\left(\varphi(z^{(k)}), t^{(k)}\right) + \alpha\,\mathcal{R}(\varphi), \qquad \varphi \in \mathcal{H}.$$

Параметр $\alpha > 0$ задаёт баланс между согласованием с контрольной выборкой и ограничением сложности функции. Регуляризованной оценкой функции утечки называется

$$\hat{\varphi}_\alpha \in \arg\min_{\varphi \in \mathcal{H}} \mathcal{J}_\alpha(\varphi).$$

Смысл оценки $\hat{\varphi}_\alpha$ состоит в том, что обнаруживаемая зависимость должна быть не только эмпирически точной, но и устойчивой относительно шума, разбиения выборки и малых изменений данных.

Функция утечки похожа на регрессионную модель только формально. В обычной регрессии хорошее предсказание является целевым полезным свойством: модель строится, чтобы предсказывать отклик. В анализе косвенной утечки ситуация обратная. Чем лучше по наблюдаемым выходам восстанавливается защищаемый атрибут, тем сильнее диагностический сигнал риска. Поэтому качество модели интерпретируется не как успех сервиса, а как свидетельство того, что выходы конвейера несут информацию, которую желательно ограничить или учесть в регламенте.

По этой причине нельзя просто взять максимально гибкую модель и объявить любое высокое качество утечкой. Такая модель может использовать случайные совпадения в конечной выборке и давать завышенный риск. Но нельзя и ограничиться слишком простым линейным описанием, если фактическая зависимость проходит через нелинейные операции конвейера. Регуляризация нужна как управляемый компромисс: она допускает достаточно богатый класс функций, но требует, чтобы найденная зависимость была простой или гладкой в выбранной геометрии и воспроизводилась на контрольных разбиениях.

Замечание. Низкое качество восстановления в выбранном классе функций не доказывает отсутствия всех возможных утечек; оно означает, что в данном испытательном режиме и выбранном классе восстановление не достигло заданного уровня обнаруживаемости.

В квадратичном случае эта постановка переходит в регуляризацию Тихонова [53]. Если

$$\ell(u, v) = \frac{1}{2}(u - v)^2$$

и в параметрическом классе

$$\varphi(z;w)=\sum_{j=1}^{p} w_j\,\psi_j(z),\qquad w\in\mathbb{R}^p,$$

используется штраф $\|\,w\,\|_2^2$, то задача принимает вид

$$Q_\alpha(w)=\frac{1}{2N}\,\|\,Xw-t_N\,\|_2^2+\frac{\alpha}{2}\,\|\,w\,\|_2^2,$$

где $X_{k,j}=\psi_j(z^{(k)})$, а $t_N=(t^{(1)},\dots,t^{(N)})^\top$. Минимизатор удовлетворяет нормальным уравнениям

$$\left(\frac{1}{N}X^\top X+\alpha I\right)\widehat{w}_\alpha=\frac{1}{N}X^\top t_N.$$

Выбор α может выполняться по принципу невязки Морозова: ошибка согласования с данными сопоставляется с оценённым уровнем шума [54]. Для итерационных методов регуляризующим параметром служит момент остановки. Если оптимизацию продолжать слишком долго, модель подгоняет случайные особенности выборки. Остановка при стабилизации валидационной ошибки или при достижении допустимой невязки ограничивает сложность полученной оценки.

Поэтому восстановление функции утечки не сводится к неограниченной регрессии. Вывод о наличии косвенной утечки должен опираться на устойчивую оценку $\widehat{\varphi}_\alpha$, воспроизводимое качество на контрольных разбиениях и статистическую проверку того, что связь между z и t не объясняется случайной перестановкой ответов.

3.5. ВОССТАНОВЛЕНИЕ ФУНКЦИИ УТЕЧКИ В RKHS

3.5.1. Почему RKHS подходит для функции утечки

Зависимость защищаемого атрибута t от наблюдаемого вектора z редко обязана быть линейной. Даже если отдельные операции конвейера F имеют простую форму, их последовательное соединение может включать фильтрацию, пороговые правила, агрегирование, обучение модели и постобработку, поэтому итоговая функция восстановления может быть существенно нелинейной. При этом заранее задать конечный набор признаков, достаточный для описания всех возможных зависимостей, обычно невозможно.

Гильбертовы пространства с воспроизводящим ядром (RKHS) дают удобный компромисс между гибкостью и регуляризуемостью [55, 56]. Пусть

$$S = Y \times X_{\text{obs}}$$

— пространство наблюдаемых векторов z, $K: S \times S \to \mathbb{R}$ — положительно определённое ядро, а $\mathcal{H}_K$ — соответствующее RKHS. Ядро K задаёт допустимый класс гладкости и сложности функции восстановления: разные ядра по-разному штрафуют быстрые изменения, локальные особенности и взаимодействия компонент z.

Интуитивно $K(z, z')$ можно рассматривать как меру близости двух сценариев наблюдения. Если два наблюдаемых вектора близки в геометрии ядра, то функция из $\mathcal{H}_K$ с малой нормой склонна принимать на них близкие значения. Поэтому выбор ядра является частью предположений аудита: он фиксирует, какие сценарии считаются похожими и какие изменения функции восстановления считаются сложными.

Линейное ядро соответствует предположению, что зависимость защищаемого атрибута от наблюдаемых выходов приблизительно линейна. Гауссово, или RBF-ядро, удобно как модель гладких локальных зависимостей: близкие наблюдения сильно влияют друг на друга, а далёкие сценарии почти не связываются. Полиномиальное ядро позволяет учитывать взаимодействия признаков фиксированной степени. Эти примеры не задают новых экспериментов; они лишь показывают, как выбор K переводит содержательное предположение о форме утечки в математический класс функций.

Норма

$$\| \varphi \|_{\mathcal{H}_K}$$

служит мерой сложности функции. Поэтому в задаче восстановления функции утечки естественно выбирать регуляризатор

$$\mathcal{R}(\varphi) = \| \varphi \|^2_{\mathcal{H}_K}.$$

Такой выбор позволяет искать нелинейную зависимость $t \approx \varphi^{\dagger}(z)$ без ручного перечисления всех признаков, но одновременно подавлять функции, которые достигают малого эмпирического остатка только за счёт подгонки шума. Именно эта логика

используется в регуляризованной постановке анализа косвенной утечки [47, 48].

Малая RKHS-норма означает не просто малую амплитуду функции, а меньшую сложность в геометрии выбранного ядра. Для одного ядра это может соответствовать гладкости, для другого — ограничению высоких степеней взаимодействия, для третьего — слабой локальной изменчивости. Поэтому штраф по норме является способом формализовать требование: функция утечки должна объяснять данные устойчиво, а не за счёт резких колебаний между отдельными контрольными наблюдениями.

3.5.2. Оператор выборки и регуляризация

При квадратичной потере получаем функционал

$$\mathcal{J}_\alpha(\varphi) = \frac{1}{2N}\sum_{k=1}^{N}(\varphi\left(z^{(k)}\right) - t^{(k)})^2 + \frac{\alpha}{2}\parallel \varphi \parallel^2_{\mathcal{H}_K}, \qquad \varphi \in \mathcal{H}_K. \quad (3.5.1)$$

Обозначим оператор выборки значений через

$$\mathcal{A}_N: \mathcal{H}_K \to \mathbb{R}^N, \qquad (\mathcal{A}_N\varphi)_k = \varphi(z^{(k)}).$$

Тогда (3.5.1) переписывается как

$$\mathcal{J}_\alpha(\varphi) = \frac{1}{2N}\parallel \mathcal{A}_N\varphi - t_N \parallel_2^2 + \frac{\alpha}{2}\parallel \varphi \parallel^2_{\mathcal{H}_K}.$$

Оператор $\mathcal{A}_N$ переводит функцию из бесконечномерного пространства в конечный вектор её значений на контрольных наблюдениях. Его сопряжённый оператор имеет вид

$$\mathcal{A}_N^* v = \sum_{k=1}^{N} v_k\, K(\cdot, z^{(k)}), \qquad v = (v_1, \dots, v_N)^\top.$$

Условие минимума имеет операторную форму

$$\left(\frac{1}{N}\mathcal{A}_N^*\mathcal{A}_N + \alpha I\right)\hat{\varphi}_\alpha = \frac{1}{N}\mathcal{A}_N^* t_N.$$

При $\alpha > 0$ оператор в левой части строго положителен, поэтому регуляризованная задача устойчива относительно малых изменений вектора ответов t_N.

Без регуляризации решение задачи на конечной выборке может быть неустойчивым: малые собственные направления эмпирического оператора $\mathcal{A}_N^*\mathcal{A}_N/N$ усиливают шум в ответах $t^{(k)}$. Добавка αI действует как спектральный фильтр: направления с собственными значениями, малыми по сравнению с α, подавляются

сильнее, чем направления с выраженным сигналом. Поэтому параметр α задаёт не только гладкость решения, но и степень доверия к слабо наблюдаемым компонентам зависимости [52, 53].

В прикладной интерпретации α связан с тем, насколько осторожно аудитор относится к слабым и шумным зависимостям. Чем выше шум ответов, тем сильнее должна быть регуляризация, иначе модель начнёт воспроизводить случайные флуктуации. Чем меньше контрольная выборка, тем меньше оснований доверять сложной функции. Если требования к устойчивости вывода высоки, отчёт должен показывать не одно оптимальное значение α, а диапазон, в котором вывод о наличии или отсутствии обнаруживаемой зависимости сохраняется.

3.5.3. Теорема представления

По теореме представления минимизатор (3.5.1) лежит в конечномерной линейной оболочке ядровых сдвигов в точках наблюдений:

$$\hat{\varphi}_\alpha(\cdot) = \sum_{k=1}^{N} a_k \, K(\cdot, z^{(k)}).$$

Это означает, что бесконечномерная вариационная задача фактически сводится к системе линейных уравнений размера N, где N — число контрольных наблюдений. Если G_N — матрица Грама,

$$[G_N]_{jk} = K(z^{(j)}, z^{(k)}),$$

то коэффициенты $a = (a_1, \dots, a_N)^\top$ находятся из системы

$$(G_N + N\alpha I)a = t_N. \qquad (3.5.2)$$

Эта система согласована с нормировкой функционала (3.5.1): множитель $N\alpha$ возникает из сочетания среднего квадратичного остатка и штрафа $\alpha \parallel \varphi \parallel^2_{\mathcal{H}_K}/2$.

Матрица G_N описывает попарную близость наблюдений в геометрии, заданной ядром. Если наблюдения близки или выходные каналы сильно коррелированы, G_N может быть плохо обусловлена. Добавка $N\alpha I$ сдвигает спектр матрицы Грама и улучшает численную обусловленность системы. Слишком малое α делает решение чувствительным к шуму и почти линейно зависимым наблюдениям;

слишком большое α сглаживает функцию настолько, что слабая, но воспроизводимая утечка может перестать обнаруживаться.

Численная стабилизация в (3.5.2) не является ухудшением модели в содержательном смысле. Она означает, что плохо наблюдаемые направления, по которым данные не позволяют надёжно отличить сигнал от шума, получают меньший вес. Если G_N имеет малые собственные значения, то нерегуляризованное решение может требовать больших коэффициентов a_k с взаимной компенсацией вкладов ядровых функций. Такая компенсация часто даёт высокую точность на обучающей выборке и слабую переносимость на новые наблюдения. Слагаемое $N\alpha I$ ограничивает этот механизм.

3.5.4. Выбор параметра регуляризации

Выбор α является частью процедуры диагностики, а не технической деталью. Если имеется оценка уровня шума, можно использовать принцип невязки Морозова: параметр выбирается так, чтобы средняя ошибка согласования с данными соответствовала ожидаемой шумовой невязке [54]. В эмпирической постановке часто применяется кросс-валидационный подбор: α выбирается по качеству на валидационных разбиениях, а затем проверяется устойчивость результата при близких значениях параметра.

Интерпретация α связана с уровнем шума. При более шумных ответах или при малом числе наблюдений требуется более сильная регуляризация, иначе оценка $\hat{\varphi}_\alpha$ начинает воспроизводить случайные флуктуации. При слишком большом α модель становится чрезмерно гладкой и может потерять слабую зависимость между z и t. Поэтому для анализа косвенной утечки важно не только найти одно значение α, дающее высокий R^2_{CV}, но и проверить, сохраняется ли вывод о зависимости при разумном диапазоне регуляризации [47, 48].

Замечание. В отчёте по риску нельзя указывать только найденное значение R^2_{CV}. Необходимо фиксировать ядро K, диапазон значений α, способ выбора α, число фолдов кросс-валидации, число перестановок в тесте и оценку уровня шума. Без этих сведений одинаковое значение качества может иметь разный смысл: оно может быть устойчивым эффектом или результатом удачного подбора модели к конкретному разбиению.

3.5.5. Ранняя остановка Ландвебера

Явный штраф в функционале (3.5.1) можно дополнять или заменять итерационной регуляризацией. Для квадратичной невязки без явного штрафа базовая итерация Ландвебера в операторной форме записывается как

$$\varphi_{m+1} = \varphi_m - \eta\left(\frac{1}{N}\mathcal{A}_N^*(\mathcal{A}_N\varphi_m - t_N)\right),$$

где шаг η выбирается из стандартного условия устойчивости градиентной схемы [57]. При наличии явного тихоновского штрафа к градиенту добавляется слагаемое $\alpha\varphi_m$.

Ранняя остановка является формой регуляризации: малое число итераций не успевает восстановить высокочастотные и плохо обусловленные компоненты, которые чаще всего связаны с шумом и случайной структурой конечной выборки. Поздняя остановка, напротив, приближает решение к нерегуляризованной подгонке и может ухудшать обобщение. Поэтому момент остановки можно рассматривать как дополнительный параметр контроля сложности функции утечки [58].

3.6. КРИТЕРИЙ ОБНАРУЖИВАЕМОСТИ И ПОРОГ N^*_{DET}

После построения оценки $\hat{\varphi}_\alpha$ необходимо отделить статистически значимую зависимость от случайной подгонки конечной выборки. Для этого используется кросс-валидированная предсказуемость [47, 48].

3.6.1. Почему нужны два критерия: качество и значимость

Один показатель качества не даёт полной картины. Высокое значение $R^2_{\mathrm{CV}}(N)$ показывает, что модель воспроизводимо предсказывает t лучше константного прогноза на выбранных разбиениях. Но при малой выборке, сложном подборе параметров или случайной структуре данных часть такого качества может возникнуть без реальной связи между z и t. Поэтому качество должно сопровождаться проверкой статистической значимости.

С другой стороны, малое значение $p_{\mathrm{perm}}(N)$ само по себе не всегда означает практический риск. На больших выборках можно обнаружить очень слабый эффект, который статистически отличим от нуля, но объясняет ничтожную долю вариации защищаемого атрибута.

Поэтому критерий обнаруживаемости использует два условия одновременно:

$$R^2_{\mathrm{CV}}(N) \geq \tau_R, \qquad p_{\mathrm{perm}}(N) \leq \alpha_{\mathrm{sig}}.$$

Первое условие отвечает за практическую величину эффекта, второе — за статистическую убедительность вывода. Только их совместное выполнение позволяет говорить, что в выбранном испытательном режиме зависимость стала не только заметной по качеству, но и маловероятной как случайная подгонка.

3.6.2. Практический протокол вычисления $R^2_{\mathrm{CV}}(N)$

Для фиксированного числа наблюдений N контрольная выборка $\mathcal{D}_N$ разбивается на обучающие и валидационные части. На обучающей части строится регуляризованная оценка $\hat{\varphi}_\alpha$, а на валидационной части вычисляется среднеквадратичная ошибка. В L-кратной кросс-валидации эта процедура повторяется для L разбиений, после чего ошибки усредняются [59]:

$$\mathrm{MSE}_{\mathrm{CV}}(N) = \frac{1}{L}\sum_{\ell=1}^{L}\frac{1}{|I_\ell|}\sum_{k\in I_\ell}(\hat{\varphi}_{\alpha,\ell}(z^{(k)}) - t^{(k)})^2,$$

где I_ℓ — валидационный фолд, а $\hat{\varphi}_{\alpha,\ell}$ — оценка, построенная без наблюдений из I_ℓ.

Кросс-валидированная предсказуемость задаётся формулой

$$R^2_{\mathrm{CV}}(N) = 1 - \frac{\mathrm{MSE}_{\mathrm{CV}}(N)}{\mathrm{Var}_{\mathrm{val}}(t)},$$

где $\mathrm{Var}_{\mathrm{val}}(t)$ — дисперсия защищаемого атрибута на валидационных наблюдениях. Значение $R^2_{\mathrm{CV}}(N) > 0$ означает, что восстановление по z лучше константного прогноза по среднему значению t. Отрицательные значения возможны и важны: они показывают, что построенная модель на валидации хуже тривиального прогноза, то есть при данном N, классе функций и регуляризации воспроизводимая зависимость не выявлена.

3.6.3. Перестановочный тест

Кросс-валидация оценивает воспроизводимость предсказания, но сама по себе не задаёт уровень статистической значимости. Для этого применяется перестановочный тест [60]. Наблюдаемые векторы $z^{(k)}$ фиксируются, а значения $t^{(k)}$ случайно переставляются между наблюдениями. На каждой переставленной выборке модель заново обучается с той же процедурой выбора или фиксации α, после чего вычисляется значение $R^2_{\mathrm{CV},b}(N)$ для перестановки $b = 1, \dots, B_{\mathrm{perm}}$.

Полученное распределение соответствует нулевой гипотезе отсутствия связи между z и t при сохранении маргинального распределения защищаемого атрибута. Если $R^2_{\mathrm{CV},obs}(N)$ — наблюдаемое качество на непереставленных данных, то p-value перестановочного теста определяется как

$$p_{\mathrm{perm}}(N) = \frac{1 + \#\{b: R^2_{\mathrm{CV},b}(N) \geq R^2_{\mathrm{CV},obs}(N)\}}{1 + B_{\mathrm{perm}}}.$$

Добавление единицы в числитель и знаменатель предотвращает нулевое p-value при конечном числе перестановок. Малое значение $p_{\mathrm{perm}}(N)$ означает, что сравнимое качество редко возникает после разрушения связи между z и t.

3.6.4. Определение порога

Порог обнаруживаемости задаётся как минимальный объём наблюдений, при котором одновременно достигнуты требуемое качество и статистическая значимость:

$$N^*_{\text{det}} = \min\{N \in \mathbb{N}: R^2_{\text{CV}}(N) \geq \tau_R, \quad p_{\text{perm}}(N) \leq \alpha_{\text{sig}}\},$$

Здесь τ_R отвечает за практическую значимость: зависимость считается существенной только если она объясняет заданную долю вариации защищаемого атрибута. Параметр α_{sig} отвечает за статистическую значимость: наблюдаемое качество должно быть маловероятным при случайной перестановке ответов.

3.6.5. Выбор порогов τ_R и α_{sig}

Уровень α_{sig} обычно выбирается как стандартный уровень статистической значимости, например в соответствии с внутренней политикой аудита или требованиями к числу ложных срабатываний. Его роль состоит в том, чтобы ограничить вероятность принять случайную зависимость за содержательную при нулевой гипотезе перестановочного теста.

Порог τ_R имеет прикладную природу. Он должен отвечать на вопрос, какая доля объяснённой вариации защищаемого атрибута уже делает восстановление значимым для процесса. Для разных атрибутов t этот порог может отличаться. Если атрибут связан с высоким правовым, финансовым или репутационным риском, то даже небольшая воспроизводимая предсказуемость может быть нежелательной, и τ_R выбирается ниже. Для менее чувствительных атрибутов или для грубых агрегированных оценок допустимый уровень практической значимости может быть выше.

Важно, что τ_R не следует выбирать после просмотра результата так, чтобы подтвердить желаемый вывод. Он должен задаваться до основной проверки или явно фиксироваться в отчёте как часть сценария анализа. Тогда порог N^*_{det} становится управленческой характеристикой выбранного режима, а не подгонкой под уже полученную кривую качества.

3.6.6. Немонотонность эмпирических кривых

На конечных выборках кривые $R^2_{\text{CV}}(N)$ и $p_{\text{perm}}(N)$ могут быть немонотонными: добавление наблюдений меняет разбиения, уровень шума, оценку α и локальные условия выборки. Поэтому

нельзя механически брать первое случайное пересечение порога, если соседние значения N не подтверждают тот же вывод.

Практически полезны повторные разбиения, сглаживание кривой, bootstrap- или повторные подвыборки, а также требование устойчивого выполнения обоих критериев на нескольких соседних значениях N. Эти приёмы помогают отделить устойчивый переход в область обнаруживаемости от случайного колебания конечной выборки. Если отдельно выделяются пороги N_{sig} и N_{pow}, то запись

$$N_{\mathrm{det}}^{*} = \max\{N_{\mathrm{sig}}, N_{\mathrm{pow}}\}.$$

корректна только для монотонной или явно монотонизированной оценки критериев.

Когда ниже для краткости используется обозначение N^{*}, оно относится к уже определённому порогу обнаруживаемости N_{det}^{*}; вероятностный порог атаки обозначается отдельно как $N_{A}(p_{\mathrm{crit}})$.

3.6.7. Протокол оценки порога обнаруживаемости

Практическая процедура оценки N_{det}^{*} может быть записана в следующем виде.

1. Зафиксировать ИТ-конвейер F и защищаемый атрибут t.
2. Сформировать контрольные выборки $\mathcal{D}_{N}$ для сетки значений N.
3. Для каждого N подобрать или зафиксировать параметр регуляризации α.
4. Обучить регуляризованную оценку $\hat{\varphi}_{\alpha}$.
5. Вычислить $R_{\mathrm{CV}}^{2}(N)$ по кросс-валидационной процедуре.
6. Провести перестановочный тест и получить $p_{\mathrm{perm}}(N)$.
7. Найти первое N, при котором выполнены оба критерия обнаруживаемости.
8. Проверить устойчивость результата к разбиениям, шуму и набору наблюдаемых каналов.

3.6.8. Что должно попасть в технический отчёт

Чтобы значение N_{det}^{*} было воспроизводимым, технический отчёт должен содержать не только итоговое число, но и описание

процедуры его получения. Минимальный перечень элементов отчёта включает:

- описание ИТ-конвейера F и его наблюдаемых выходов;
- описание защищаемого атрибута t;
- описание полного наблюдаемого вектора z и открытых признаков x_{obs};
- объём и способ формирования контрольной выборки;
- уровень шума или способ его оценки;
- выбранный класс функций, ядро K и диапазон параметра α;
- значения $R^2_{\mathrm{CV}}(N)$ на рассматриваемой сетке N;
- значения $p_{\mathrm{perm}}(N)$ и число перестановок;
- итоговый порог N^*_{det} и правило обработки немонотонности;
- чувствительность результата к шуму, набору каналов и разбиениям;
- список предположений, при которых сделан вывод.

Если позже меняется набор выходов, уровень шума или доступный наблюдателю контекст, прежнее значение N^*_{det} не следует переносить без повторной проверки.

N^*_{det} не является границей безопасности. Это минимальный бюджет контрольных наблюдений, при котором зависимость между z и t стала воспроизводимо обнаружимой в выбранном испытательном режиме. Регламентный лимит обращений должен задаваться меньше этого значения с запасом. Если у наблюдателя есть внешняя информация, возможность адаптивного выбора запросов или доступ к коррелированным источникам, запас следует увеличивать, поскольку фактическая информативность наблюдений может быть выше, чем в контрольной процедуре.

3.7. ВЕРОЯТНОСТЬ РАННЕЙ УТЕЧКИ

В предыдущих разделах N обозначал размер контрольной выборки, используемой для обучения и диагностики функции утечки. В настоящем разделе рассматривается локальная модель накопления информации об одном фиксированном защищаемом атрибуте t при повторных наблюдениях одного типа. Формально это другой уровень описания: он нужен для оценки вероятности

успешного восстановления до достижения статистического порога обнаруживаемости [8].

Рассмотрим линеаризованную модель очищенного наблюдения

$$\tilde{z} = s\,t + u, \qquad \mathbb{E}u = 0, \qquad \mathrm{Cov}(u) = \Sigma_u,$$

где $\tilde{z}$ — центрированная или очищенная версия ранее введённого наблюдаемого вектора z после вычитания фонового вклада, s — вектор чувствительности наблюдаемого канала к защищаемому атрибуту t, а u — шум. Независимость повторных наблюдений ниже используется как модельное упрощение. Если наблюдения коррелированы, эффективное число независимых наблюдений меньше фактического, и формулы с N следует интерпретировать через эффективный объём данных.

Для одного наблюдения внешняя процедура восстановления может использовать линейную оценку

$$q^{(k)} = w^{\top}\tilde{z}^{(k)}.$$

Условие несмещённости имеет вид $w^{\top}s = 1$, тогда $\mathbb{E}q^{(k)} = t$. Оптимальный по дисперсии выбор весов пропорционален $\Sigma_u^{-1}s$, а эффективная дисперсия одной оценки равна

$$\sigma_{\mathrm{eff}}^2 = (s^{\top}\Sigma_u^{-1}s)^{-1} = \frac{1}{\kappa}, \qquad \kappa = s^{\top}\Sigma_u^{-1}s.$$

Параметр κ характеризует отношение сигнала к шуму в канале: чем он больше, тем меньше эффективная дисперсия и тем быстрее накапливается информация о t.

Величина σ_{eff}^2 не является шумом одного наблюдаемого ответа. Она описывает остаточную неопределённость после оптимального линейного объединения всех компонент очищенного вектора $\tilde{z}$. Если несколько каналов несут согласованную информацию о t, эффективная дисперсия может быть меньше дисперсии каждого отдельного грубого канала. Если же каналы слабы или сильно зашумлены, σ_{eff}^2 остаётся большой, и вероятность успешного восстановления растёт медленно.

По N независимым наблюдениям строится средняя оценка

$$\hat{t}^{(N)} = \frac{1}{N}\sum_{k=1}^{N} q^{(k)}.$$

Она несмещённа, а её дисперсия убывает обратно пропорционально числу наблюдений:

$$\mathrm{Var}\left(\hat{t}^{(N)}\right) = \frac{\sigma_{\mathrm{eff}}^2}{N}.$$

При нормальном шуме

$$\hat{t}^{(N)} - t \sim \mathcal{N}\left(0, \frac{\sigma_{\mathrm{eff}}^2}{N}\right). \qquad (3.7.1)$$

Вероятность успешной атаки при фиксированном N обозначим через

$$p_A(N) = \Pr\left\{|\hat{t}^{(N)} - t| < \varepsilon_R\right\}.$$

Из (3.7.1) следует

$$p_A(N) = \Phi\left(\frac{\varepsilon_R\sqrt{N}}{\sigma_{\mathrm{eff}}}\right) - \Phi\left(-\frac{\varepsilon_R\sqrt{N}}{\sigma_{\mathrm{eff}}}\right) = 2\Phi\left(\frac{\varepsilon_R\sqrt{N}}{\sigma_{\mathrm{eff}}}\right) - 1, \qquad (3.7.2)$$

где Φ — функция распределения стандартного нормального закона.

Для заданной критической вероятности p_{crit} введём вероятностный порог атаки

$$N_A(p_{\mathrm{crit}}) = \min\{N \in \mathbb{N}: p_A(N) \geq p_{\mathrm{crit}}\}.$$

Порог $N_A(p_{\mathrm{crit}})$ не совпадает по смыслу с N_{det}^* : первый относится к вероятности успешного восстановления, второй — к статистической обнаруживаемости зависимости.

Таким образом, в анализе присутствуют три разные величины. N_{det}^* относится к статистической обнаруживаемости зависимости на контрольной выборке; $N_A(p_{\mathrm{crit}})$ относится к достижению заданной вероятности успешной атаки в локальной модели накопления информации; N_{limit} задаёт эксплуатационный лимит в регламенте доступа. Эти величины не обязаны совпадать и не должны смешиваться в интерпретации.

В зависимости от модели, шума, точности ε_R и выбранных критериев одно из этих значений может оказаться меньше другого. Если слабая зависимость воспроизводимо проявляется только на большой контрольной выборке, но отдельная удачная реализация шума уже даёт достаточную для восстановления точность, то $N_A(p_{\mathrm{crit}})$ может быть критичным раньше, чем возникает устойчивое выполнение критерия обнаруживаемости. В другой ситуации статистическая связь может быть обнаружена относительно рано, но требуемая точность восстановления достигается только при большем числе повторов.

Рациональная политика доступа должна учитывать минимум из этих ограничений не буквально, а с запасом: N_{limit} выбирается так, чтобы одновременно не достигать области обнаруживаемости и удерживать $p_A(N_{\text{limit}})$ ниже допустимого уровня.

Формула (3.7.2) описывает раннюю утечку: при $N < N^*_{\text{det}}$ вероятность успешного восстановления не равна нулю, хотя может быть малой. По мере приближения к пороговым значениям $p_A(N)$ растёт, поэтому инженерное ограничение числа запросов должно задаваться с запасом относительно как N^*_{det}, так и $N_A(p_{\text{crit}})$.

Ранняя утечка не противоречит порогу обнаруживаемости. Порог N^*_{det} описывает момент, когда зависимость становится воспроизводимым статистическим фактом в контрольной процедуре. Функция $p_A(N)$ описывает вероятность успеха отдельной процедуры восстановления при фиксированном бюджете наблюдений. Поэтому утверждение "до порога" не означает "нулевой риск": оно означает, что в выбранном испытательном режиме зависимость ещё не достигла заданного уровня обнаруживаемости.

Более детальную характеристику даёт случайная величина времени до восстановления

$$\tau = \inf\{N: |\hat{t}^{(N)} - t| < \varepsilon_R\}.$$

Распределение τ описывает не только вероятность успеха к фиксированному числу наблюдений, но и разброс момента, когда требуемая точность впервые достигается. Для инженерной оценки часто достаточно функции $p_A(N)$ и порога $N_A(p_{\text{crit}})$, однако величина τ подчёркивает, что раннее восстановление является вероятностным событием, а не резким переходом в одной точке.

Распределение τ полезно в задачах, где важен не только риск к заранее заданному числу обращений, но и вероятность слишком раннего достижения точности. Например, два режима могут иметь одинаковое значение $p_A(N_{\text{limit}})$, но различаться вероятностью того, что точность будет достигнута уже на первой половине разрешённого бюджета.

Следующая характеристика относится не к повторному усреднению одного фиксированного t, а к популяционному пределу качества модели восстановления в линейно-гауссовой схеме, когда размер контрольной выборки достаточен для оценки зависимости. Она показывает, какая доля дисперсии случайного защищаемого

атрибута t объяснима наблюдаемым каналом при фиксированной шумовой модели. Если

$$\sigma_t^2 = \mathrm{Var}(t),$$

то общий вид этой величины записывается как

$$R_\infty^2 = \frac{\sigma_t^2 \kappa}{1 + \sigma_t^2 \kappa}, \qquad \sigma_t^2 = \mathrm{Var}(t).$$

При нормировке $\mathrm{Var}(t) = 1$ эта формула принимает вид $R_\infty^2 = \kappa/(1+\kappa)$. Без такой нормировки величина κ сама по себе не задаёт безразмерную долю объяснённой дисперсии: масштаб t входит в отношение сигнал–шум через множитель σ_t^2. Вместе κ, R_∞^2, $R_{\mathrm{CV}}^2(N)$, $p_{\mathrm{perm}}(N)$, $p_A(N)$ и N_{det}^* дают согласованное описание силы утечки, её статистической обнаруживаемости и риска раннего успеха атаки.

При сравнении сценариев необходимо фиксировать масштаб защищаемого атрибута. Если в одном расчёте t измеряется в исходных единицах, а в другом стандартизован, то одинаковая величина κ имеет разную интерпретацию для доли объяснённой дисперсии. Поэтому отчёт должен указывать, проводилась ли стандартизация t, и в каких единицах задаётся точность ε_R.

3.8. МОДЕЛЬНЫЕ ПРИМЕРЫ

3.8.1. Простая утечка через среднее значение

Пусть в школьном классе обучается m учеников, а внешний пользователь может запрашивать только средний балл по заданной группе. Баллы исходных учеников обозначим через $x_1, \ldots, x_m$, а балл нового ученика, который является защищаемым атрибутом, через

$$t = x_i.$$

До прихода нового ученика пользователь получает

$$y^{(1)} = \frac{1}{m}\sum_{j=1}^{m} x_j.$$

После добавления нового ученика повторный запрос даёт

$$y^{(2)} = \frac{1}{m+1}\left(\sum_{j=1}^{m} x_j + t\right).$$

Если обозначить $S_0 = \sum_{j=1}^{m} x_j$, то

$$y^{(1)} = \frac{1}{m} S_0, \qquad y^{(2)} = \frac{1}{m+1}(S_0 + t).$$

Отсюда немедленно следует

$$t = (m+1)y^{(2)} - my^{(1)}. \tag{3.8.1}$$

Эта формула показывает алгебраическую определённость: при двух точных ответах неизвестные S_0 и t определяются однозначно.

Поэтому величину $N_{\mathrm{rank}} = 2$ следует понимать как число запросов, достаточное для восстановления в детерминированной модели. Порог N^*_{det} относится к другой процедуре: он определяется по кросс-валидированному качеству восстановления и перестановочному тесту на контрольной выборке.

Формально это можно записать как линейную систему

$$\begin{pmatrix} y^{(1)} \\ y^{(2)} \end{pmatrix} = \begin{pmatrix} \frac{1}{m} & 0 \\ \frac{1}{m+1} & \frac{1}{m+1} \end{pmatrix} \begin{pmatrix} S_0 \\ t \end{pmatrix}.$$

При двух запросах матрица имеет полный ранг, поэтому регуляризованная оценка в пределе $\alpha \to 0^+$ переходит в точное решение системы. При одном запросе ранг равен единице: либо раскрывается только S_0, либо только сумма $S_0 + t$, но не сам защищаемый атрибут.

Следовательно, в детерминированной модели без шума минимальное число наблюдений для алгебраического восстановления равно

$$N_{\mathrm{rank}} = 2.$$

Этот пример показывает, что интуитивный момент утечки совпадает с формальным условием определённости линейной обратной задачи, но не заменяет статистическую диагностику для шумных и конечных данных.

Если опубликованные средние содержат аддитивные погрешности, то вместо $y^{(1)}$ и $y^{(2)}$ наблюдаются

$$y_\varepsilon^{(1)} = y^{(1)} + \varepsilon_1, \qquad y_\varepsilon^{(2)} = y^{(2)} + \varepsilon_2.$$

Прямое применение той же алгебраической формулы даёт оценку

$$\hat{t} = (m+1)y_\varepsilon^{(2)} - my_\varepsilon^{(1)}.$$

Подстановка определений шумных ответов и формулы (3.8.1) немедленно приводит к ошибке

$$\hat{t} - t = (m+1)\varepsilon_2 - m\varepsilon_1.$$

Если ε_1 и ε_2 независимы, имеют нулевые средние и дисперсии σ_1^2 и σ_2^2, то

$$\mathrm{Var}(\hat{t} - t) = (m+1)^2\sigma_2^2 + m^2\sigma_1^2 \qquad (3.8.2)$$

Тем самым шум снижает точность восстановления, но его влияние усиливается коэффициентами, с которыми сравниваются два близких агрегата. При большом размере исходной группы даже малые ошибки средних могут давать заметную неопределённость оценки t.

Содержательный вывод состоит в том, что простейший агрегат становится опасным не сам по себе, а в паре с близким агрегатом, отличающимся включением или исключением одного объекта. Если такие пары ответов недоступны, алгебраическое восстановление может быть не определено; если они доступны многократно или публикуются с малым шумом, риск восстановления сохраняется.

Минимальный размер группы сам по себе не устраняет проблему, если разрешены близкие пересекающиеся группы. Пусть каждый отдельный отчёт строится по достаточно большой группе и выглядит агрегированным. Если пользователь может получить два отчёта, различающиеся одним объектом, то разность масштабированных средних выделяет вклад этого объекта. Поэтому требование к минимальному размеру группы должно рассматриваться вместе с ограничениями на близость групп и частоту повторов.

В шумовой версии пример показывает ещё один важный эффект. Добавление шума в каждый ответ увеличивает неопределённость восстановления, но коэффициенты $(m+1)$ и m усиливают шумы при обратном пересчёте к индивидуальному атрибуту. Если шумы независимы и достаточно велики, дисперсия (3.8.2) может сделать восстановление неточным. Если же шум мал, повторные запросы усредняются или наблюдатель получает несколько близких пар агрегатов, вероятность восстановления может оставаться заметной.

Важно контролировать не только форму одного отчёта, но и набор разрешённых отчётов в целом. Для агрегатов по группам существенны ограничения на пересечение групп, запрет публикации точных изменений при добавлении или исключении малых подмножеств, задержка обновления отчётов и единый учёт

повторных обращений. Иначе формально агрегированные ответы могут образовывать систему уравнений, из которой индивидуальный вклад выделяется алгебраически.

Вывод из примера. Проверка одного агрегата недостаточна. Нужно анализировать семейство разрешённых выходов конвейера и те линейные комбинации, которые становятся доступны наблюдателю через сравнение отчётов. Даже если каждый ответ по отдельности не достигает заданного уровня обнаруживаемости, их совместное использование может существенно изменить риск восстановления.

3.8.2. Многоканальная утечка в страховом сценарии

Рассмотрим агрегированные показатели медицинского страхования по малым сегментам застрахованных. Для каждого сегмента доступны четыре наблюдаемых ответа:

$$y_1, \quad y_2, \quad y_3, \quad y_4.$$

Они могут соответствовать средним годовым выплатам, числу койко-дней, доле дорогостоящей терапии и доле тяжёлой коморбидности. Защищаемый атрибут $t = x_i$ — индивидуальный риск конкретного клиента.

После вычитания оценённого фонового вклада зависимость наблюдений от t задаётся линейной шумовой моделью

$$\tilde{y}_j = b_j t + \xi_j, \qquad j = 1, \dots, 4,$$

или в векторной форме

$$\tilde{y} = Bt + \xi, \qquad B = (b_1, \dots, b_4)^\top, \qquad \xi \sim \mathcal{N}(0, \Sigma).$$

Здесь

$$\Sigma = \mathrm{diag}(\sigma_1^2, \dots, \sigma_4^2),$$

то есть σ_j обозначает стандартное отклонение шума j -го наблюдаемого канала, а не его дисперсию.

Линейная функция утечки имеет вид

$$\varphi(\tilde{y}; w) = w^\top \tilde{y}.$$

По выборке из N наблюдений параметры восстанавливаются через функционал Тихонова

$$\mathcal{L}_\alpha(w) = \frac{1}{2N} \sum_{k=1}^{N} (w^\top \tilde{y}^{(k)} - t^{(k)})^2 + \frac{\alpha}{2} \| w \|_2^2.$$

При больших N и малом α оптимальная линейная оценка использует веса

$$w^\star = \frac{\Sigma^{-1}B}{B^\top\Sigma^{-1}B}, \qquad B^\top\Sigma^{-1}B > 0.$$

Для среднего по N наблюдениям ошибка восстановления имеет дисперсию

$$\frac{\sigma_{\text{eff}}^2}{N}, \qquad \sigma_{\text{eff}}^2 = (B^\top\Sigma^{-1}B)^{-1}.$$

В базовом численном сценарии используются параметры

$$B = \begin{pmatrix} 0{,}020 \\ 0{,}004 \\ 0{,}003 \\ 0{,}002 \end{pmatrix}, \qquad \sigma = \begin{pmatrix} 0{,}082 \\ 0{,}089 \\ 0{,}067 \\ 0{,}045 \end{pmatrix}.$$

Вектор $\sigma = (0{,}082, 0{,}089, 0{,}067, 0{,}045)^\top$ содержит стандартные отклонения каналов. Дисперсии входят в матрицу Σ как квадраты этих величин.

Для величин

$$c_j = \frac{b_j^2}{\sigma_j^2}, \qquad C = \sum_{j=1}^{4} c_j = B^\top\Sigma^{-1}B$$

получается $C \approx 6{,}55 \cdot 10^{-2}$ и

$$\sigma_{\text{eff}}^2 \approx 15{,}3.$$

Численные вклады каналов приведены в таблице 3.1.

Таблица 3.1. Вклады наблюдаемых каналов в информационную сумму

j	b_j	σ_j	$c_j = b_j^2/\sigma_j^2$	c_j/C
1	0,020	0,082	$5{,}95 \cdot 10^{-2}$	90,8%
2	0,004	0,089	$2{,}02 \cdot 10^{-3}$	3,08%
3	0,003	0,067	$2{,}00 \cdot 10^{-3}$	3,06%
4	0,002	0,045	$1{,}98 \cdot 10^{-3}$	3,02%

Доли в последнем столбце суммируются приблизительно до 100% с учётом округления. Величина c_j не является универсальной “важностью признака”; она показывает вклад канала в информационную сумму именно в данной линейно-гауссовой параметризации. Большой вклад первого канала означает, что в

этом сценарии ответ y_1 наиболее эффективно уменьшает неопределённость восстановления t.

Доминирование первого канала объясняется соотношением чувствительности и шума. Коэффициент $b_1 = 0{,}020$ существенно больше остальных b_j, а стандартное отклонение $\sigma_1 = 0{,}082$ не настолько велико, чтобы подавить этот вклад. Поэтому отношение b_1^2/σ_1^2 оказывается главным слагаемым суммы C. Если бы первый канал имел гораздо больший шум или меньшую чувствительность к t, распределение вкладов изменилось бы.

Эта таблица полезна именно как диагностический инструмент. Она показывает не абстрактную полезность медицинского показателя, а его вклад в конкретной модели восстановления. Поэтому управленческий вывод не должен звучать как универсальное правило о первом канале. Корректная формулировка: в рассматриваемой параметризации первый наблюдаемый ответ создаёт основную часть информационной суммы, и его изменение сильнее всего влияет на σ_{eff}^2.

Если задана точность восстановления $\varepsilon_R = 0{,}7$ и критическая вероятность $p_{\text{crit}} = 0{,}95$, то из формулы

$$p_A(N) = 2\Phi\left(\frac{\varepsilon_R\sqrt{N}}{\sigma_{\text{eff}}}\right) - 1$$

следует явное выражение для вероятностного порога атаки

$$N_A(p_{\text{crit}}) = \left\lceil \frac{\sigma_{\text{eff}}^2}{\varepsilon_R^2}\left[\Phi^{-1}\left(\frac{1+p_{\text{crit}}}{2}\right)\right]^2 \right\rceil.$$

Это вероятностный порог атаки, он отвечает на вопрос о достижении заданной вероятности успешного восстановления в принятой шумовой модели.

В базовой параметризации

$$N_{A,\text{base}}(p_{\text{crit}}) = N_A(0{,}95) \approx 120.$$

С учётом погрешности оценки шумов расчёт в данном модельном сценарии даёт ориентировочно 120 ± 11 наблюдений.

Два защитных изменения иллюстрируют инженерную интерпретацию порога. Если усилить шум в наиболее информативном ответе y_1, например увеличить стандартное отклонение с $0{,}082$ до $0{,}113$, то эффективная дисперсия возрастает до

$$\sigma_{\text{eff,noise}}^2 \approx 26{,}79,$$

а порог сдвигается к

$$N_{A,\mathrm{noise}}(p_{\mathrm{crit}}) \approx 211.$$

Если же исключить наиболее информативный ответ из публикуемого набора, оставив только y_2, y_3, y_4, то эффективная дисперсия возрастает до

$$\sigma^2_{\mathrm{eff,drop}} \approx 166{,}66,$$

а порог становится порядка

$$N_{A,\mathrm{drop}}(p_{\mathrm{crit}}) \approx 1307.$$

Увеличение шума в первом канале или исключение первого канала приводит к росту σ^2_{eff} и, следовательно, к росту $N_A(p_{\mathrm{crit}})$. Тем самым пример показывает, как изменение набора наблюдаемых ответов и уровня шума меняет практическую достижимость восстановления без изменения математической схемы анализа.

Эти числа относятся к вероятностному порогу атаки $N_A(p_{\mathrm{crit}})$. Они показывают, как быстро в принятой линейно-гауссовой модели растёт вероятность восстановления с заданной точностью. Для вывода о статистической обнаруживаемости зависимости по контрольной выборке требуется отдельная процедура с $R^2_{\mathrm{CV}}(N)$ и $p_{\mathrm{perm}}(N)$.

Как это понимать в реальной жизни. Если один выход даёт основной вклад, его можно рассматривать как первый кандидат на изменение: укрупнение, дополнительное зашумление, более редкое обновление или исключение из публикуемого набора. Если вклады распределены равномерно, управление одним каналом даст ограниченный эффект, и требуется менять всю группу наблюдаемых ответов или общий бюджет обращений. Абляция каналов помогает объяснить владельцу процесса, какие именно выходы создают риск и почему изменение одного отчёта может сдвигать вероятностный порог сильнее, чем изменение другого.

3.9. ОГРАНИЧЕНИЯ ПРИМЕНИМОСТИ И ИНТЕРПРЕТАЦИЯ РЕЗУЛЬТАТОВ

Количественные показатели, введённые выше, имеют смысл только относительно выбранного испытательного режима.

Контрольная выборка $\mathcal{D}_N$ должна отражать реальные сценарии эксплуатации конвейера F: диапазоны входных данных, типичные фильтры, набор наблюдаемых ответов, шумы и доступные открытые признаки. Если контрольная выборка систематически отличается от промышленного режима, оценка риска может быть смещена как в сторону завышения, так и в сторону занижения.

Представительность особенно важна для редких сценариев. Если контрольная выборка построена в основном на типичных случаях, но реальные обращения часто попадают в малые группы, крайние значения признаков или периоды нестабильной работы системы, то оценка $R^2_{\mathrm{CV}}(N)$ может не отражать наиболее информативные режимы. Поэтому при подготовке аудита необходимо фиксировать, какие сценарии покрыты выборкой, а какие остаются вне неё.

Существенным элементом процедуры является выбор класса функций восстановления. RKHS с ядром K задаёт конкретное семейство допустимых зависимостей и конкретную меру сложности $\| \varphi \|_{\mathcal{H}_K}$. Отсутствие обнаружения в этом классе означает, что в выбранном испытательном режиме и при выбранном ядре зависимость не достигла заданного уровня обнаруживаемости. Это не доказывает отсутствия утечки для всех возможных внешних процедур, особенно если используется другой класс моделей или дополнительная внешняя структура данных.

Калибровка шума также является модельным предположением. Величины σ_{eff}, δ и связанные с ними оценки регуляризации зависят от того, как именно собираются повторные наблюдения, как обновляются отчёты, какие источники случайности используются и насколько стабилен режим работы системы. При дрейфе данных или изменении механизма постобработки ранее оценённые шумовые параметры могут перестать соответствовать фактической неопределённости восстановления.

Формулы для $p_A(N)$ предполагают независимость наблюдений или, по крайней мере, корректную замену фактического числа наблюдений эффективным. В промышленных системах ответы часто коррелированы: соседние отчётные периоды пересекаются по объектам, метрики строятся на близких сегментах, а повторные запросы используют одну и ту же базовую выборку. В таких условиях

фактический прирост информации за одно обращение может отличаться от независимой модели, поэтому требуется оценивать эффективное число наблюдений или проводить отдельную проверку чувствительности к корреляции.

Неадаптивная контрольная процедура не всегда описывает поведение активного наблюдателя. Если следующие запросы выбираются с учётом предыдущих ответов, то последовательность наблюдений может концентрироваться в наиболее информативных областях пространства $S = Y \times X_{\mathrm{obs}}$. При такой адаптивности фактический риск может быть выше, чем риск, оценённый по заранее зафиксированной сетке сценариев. Поэтому результаты диагностики следует интерпретировать как характеристику указанного режима доступа, а не произвольной интерактивной стратегии.

Открытые признаки x_{obs} и сторонняя информация могут существенно усилить восстановление. Признак, не являющийся защищаемым атрибутом сам по себе, способен уменьшить неопределённость относительно t при совместном использовании с ответами конвейера. По этой причине оценка должна фиксировать не только выход y, но и полный наблюдаемый вектор $z = (y, x_{\mathrm{obs}})$, а также явно перечислять внешние признаки, которые считались доступными или недоступными наблюдателю.

Наконец, пороги N^{*}_{det} и $N_A(p_{\mathrm{crit}})$ не являются абсолютными границами безопасности. Первый порог характеризует статистическую обнаруживаемость зависимости в выбранном протоколе кросс-валидации и перестановочного теста, второй — достижение заданной вероятности восстановления в принятой вероятностной модели. Оба показателя являются инженерными характеристиками модели, данных и режима испытаний; регламентный лимит N_{limit} должен задаваться с запасом и пересматриваться при изменении этих предпосылок.

Отдельно следует учитывать дрейф данных и моделей. Изменение популяции пользователей, структуры отчётности, версии модели, правил фильтрации или политики зашумления может изменить как функцию утечки $\varphi^{\dagger}$, так и оценку $\hat{\varphi}_{\alpha}$. Поэтому результаты аудита имеют срок применимости, связанный с устойчивостью эксплуатации. Корректное применение метода

требует не только вычисления порога, но и фиксации предположений, при которых этот порог был получен.

3.10. ПРАКТИЧЕСКАЯ ИНТЕРПРЕТАЦИЯ И МЕРЫ УПРАВЛЕНИЯ

Полученные показатели позволяют перейти от бинарного вопроса «есть ли утечка» к управлению риском при заданном числе обращений [47, 48, 49]. Практическая процедура может включать следующие шаги.

1. Описать ИТ-конвейер F и перечень наблюдаемых ответов y, доступных внешним пользователям или смежным компонентам системы.

2. Выделить защищаемые скалярные атрибуты $t = x_i$ и наблюдаемые признаки x_{obs}, которые могут усиливать восстановление.

3. Сформировать контрольную выборку $\mathcal{D}_N = \{(z^{(k)}, t^{(k)})\}_{k=1}^{N}$ в моделировании, аудите или контролируемом журналировании.

4. Построить регуляризованную оценку $\hat{\varphi}_\alpha$ и проверить устойчивость результата при изменении α, разбиений кросс-валидации и набора наблюдаемых признаков.

5. Рассчитать $R^2_{\mathrm{CV}}(N)$, $p_{\mathrm{perm}}(N)$, оценку $p_A(N)$ и порог N^*_{det}.

6. Выбрать эксплуатационный лимит N_{limit} с запасом относительно N^*_{det} и $N_A(p_{\mathrm{crit}})$, учитывая вероятность ранней утечки и неопределённость оценки параметров.

7. При необходимости изменить конвейер: добавить шум, снизить детализацию публикации, укрупнить сегменты, убрать наиболее информативные компоненты ответа или ограничить частоту обновления.

Таблица 2. Интерпретация основных показателей анализа косвенной утечки

Показатель	Что означает	Какое управленческое решение поддерживает
$R^2_{\mathrm{CV}}(N)$	Воспроизводимая предсказуемость t по наблюдаемым выходам	Оценка практической значимости зависимости при заданном числе наблюдений

Показатель	Что означает	Какое управленческое решение поддерживает
$p_{\mathrm{perm}}(N)$	Вероятность получить не худшее качество при разрушенной связи между z и t	Проверка статистической значимости обнаруженной зависимости
N^*_{det}	Минимальный объём наблюдений, где зависимость стала статистически обнаружимой	Выбор лимита обращений с запасом относительно области обнаруживаемости
$p_A(N)$	Вероятность успешного восстановления при заданном бюджете наблюдений	Оценка риска раннего восстановления до или около пороговых значений
$N_A(p_{\mathrm{crit}})$	Бюджет, при котором вероятность атаки достигает критического уровня	Задание эксплуатационного ограничения по допустимой вероятности восстановления
σ^2_{eff}	Эффективная неопределённость восстановления в шумовой модели	Выбор уровня шума, степени агрегирования и частоты обновления
c_j или c_j/C	Вклад канала в информационную сумму	Решение об удалении, укрупнении или дополнительном зашумлении отдельных выходов
N_{limit}	Регламентный лимит обращений, задаваемый с запасом	Настройка квот, счётчиков и процедур реагирования
α	Сила регуляризации при восстановлении функции утечки	Контроль устойчивости вывода и требований к качеству контрольной выборки

Показатель	Что означает	Какое управленческое решение поддерживает
B_{perm}	Число перестановок в permutation test	Точность оценки $p_{\mathrm{perm}}(N)$ и доверие к статистической проверке

Такая схема не должна формулироваться как обещание полного исключения утечек. Корректнее говорить, что она снижает вероятность успешного восстановления до заданного уровня риска при явно описанных предположениях о данных, шуме, классе функций и числе обращений. Если предположения меняются, порог N^*_{det}, вероятностный порог $N_A(p_{\mathrm{crit}})$ и функция $p_A(N)$ должны пересчитываться.

Также важен мониторинг накопления информации. Даже если каждое отдельное обращение само по себе не достигает заданного уровня обнаруживаемости, последовательность обращений может постепенно уменьшать шумовую неопределённость. Поэтому в промышленной системе полезны счётчики запросов, раздельные бюджеты по пользователям и объектам, сигналы приближения к доле от N^*_{det} и $N_A(p_{\mathrm{crit}})$, а также автоматическое усиление защитных мер при подозрительном режиме доступа.

Для непреднамеренных утечек эта логика не менее существенна, чем для злонамеренных атак. ИИ-система может многократно вызывать один и тот же отчёт или модель в ходе нормальной работы: при периодическом обновлении, мониторинге качества, подборе параметров или обслуживании большого числа похожих запросов. Количественный учёт N позволяет увидеть риск накопления информации до того, как он проявится в явном инциденте.

Администратор получает не одно изолированное число, а связанный набор характеристик: качество восстановления, статистическую значимость, вероятность успеха при заданном бюджете, чувствительность к шуму и вклад отдельных каналов. Такой набор позволяет различать ситуации, в которых зависимость уже воспроизводимо обнаружима, и ситуации, где статистический критерий ещё не выполнен, но вероятность раннего восстановления становится неприемлемой.

На уровне регламента эти характеристики переводятся в конкретные настройки конвейера. Если основной вклад даёт отдельный выход, его можно исключить из публикации, укрупнить по группам или дополнительно зашумить. Если риск связан с накоплением наблюдений, применяются лимиты обращений, интервалы обновления, раздельные квоты для пользователей и контроль повторяющихся сценариев доступа.

Оценки должны пересчитываться при изменении данных, модели, перечня отчётов или режима эксплуатации. Дрейф распределений, новая версия модели, изменение фильтров или появление дополнительного открытого признака x_{obs} могут изменить как $R^2_{\mathrm{CV}}(N)$, так и $p_A(N)$. Поэтому отчёт по косвенной утечке должен содержать как значения порогов, так и перечень предположений, при которых они получены.

Для администратора результат анализа удобно читать как карту управления риском. N^*_{det} показывает, при каком объёме контрольных наблюдений зависимость становится статистически обнаружимой; $p_A(N)$ показывает, как растёт вероятность восстановления при накоплении обращений; c_j/C показывает, какие выходы дают основной вклад; чувствительность к шуму показывает, насколько изменение механизма публикации способно сдвинуть вероятностный порог. В совокупности это не одно "магическое число", а набор параметров, позволяющих выбирать режим эксплуатации.

Перевод в решения начинается с лимита обращений. Если расчётный порог обнаруживаемости и вероятностный порог атаки лежат в диапазоне, достижимом за обычный рабочий период, то N_{limit} должен задаваться существенно ниже этих значений. Если рабочий процесс требует большого числа обращений, то следует менять не только лимит, но и саму информативность выходов: добавлять шум, укрупнять когорты, уменьшать частоту обновления или исключать наиболее информативные компоненты ответа.

Особое внимание требуется к близким сравнимым агрегатам. Даже если каждый отчёт строится по достаточно большой группе, последовательность похожих отчётов может выделять вклад малого числа объектов. Поэтому регламент должен задавать не только максимальное число обращений, но и правила публикации

изменений: как часто обновляются отчёты, какие пересечения групп допустимы, можно ли сравнивать соседние периоды и как учитываются повторные обращения с почти одинаковыми параметрами.

Для реальной жизни результаты лучше формулировать через сценарии. Например: при текущей конфигурации зависимость становится статистически обнаружимой после указанного объёма контрольных наблюдений; если увеличить шум в конкретном выходе, вероятностный порог атаки сдвигается; если пользователь располагает дополнительными признаками x_{obs}, запас по лимиту должен быть больше.

Регламент должен фиксировать, кто имеет право запускать конвейер F, как считается бюджет наблюдений, переносится ли неиспользованный бюджет между периодами, кто утверждает изменение набора выходов и при каких событиях проводится повторный аудит. К таким событиям относятся изменение модели, появление нового отчёта, изменение политики зашумления, расширение круга пользователей, обнаружение коррелированных внешних данных и существенный дрейф распределений.

Наконец, отчёт должен явно разделять результаты диагностики и управленческие допущения. Диагностика сообщает значения $R^2_{\mathrm{CV}}(N)$, $p_{\mathrm{perm}}(N)$, $p_A(N)$ и пороги. Управленческая часть описывает, какой запас выбран относительно этих порогов, почему он принят достаточным для данного процесса и при каких изменениях вывод подлежит пересмотру. Такая структура помогает избежать неверной интерпретации порога как абсолютной границы и поддерживает повторяемость аудита.

3.11 ЗАКЛЮЧЕНИЕ

Регуляризованное восстановление функции утечки задаёт единую постановку анализа косвенных утечек в ИТ-конвейерах. Наблюдаемый ИТ-конвейер описывается отображением F, защищаемый скалярный атрибут обозначается через $t = x_i$, а полный наблюдаемый вектор через $z = (y, x_{\mathrm{obs}})$ или $z = (y +$

$\varepsilon, x_{\mathrm{obs}}$). Истинная функция утечки $\varphi^{\dagger}$ восстанавливается по конечной выборке в виде регуляризованной оценки $\hat{\varphi}_{\alpha}$.

Математическое ядро состоит из трёх связанных частей. Первая часть — регуляризация Тихонова и её итерационные аналоги, включая раннюю остановку и принцип невязки Морозова. Вторая часть — RKHS-постановка, где теорема представления сводит восстановление к конечномерной задаче. Третья часть — статистическая диагностика утечки через $R^2_{\mathrm{CV}}(N)$, $p_{\mathrm{perm}}(N)$, вероятность успешной атаки $p_A(N)$ и порог обнаруживаемости N^*_{det}.

Эта связка важна именно потому, что косвенная утечка не сводится к факту прямого доступа к записи. Пользователь может видеть только отчёты, метрики, агрегаты или выходы модели, но при накоплении наблюдений эти величины становятся данными обратной задачи. Регуляризация позволяет отделить устойчивую зависимость от случайной подгонки, а статистическая проверка показывает, становится ли восстановление воспроизводимым в выбранном испытательном режиме.

Существенно, что обнаруживаемость зависимости и вероятность раннего восстановления описывают разные аспекты риска. Порог N^*_{det} относится к статистически воспроизводимой обнаруживаемости связи между z и t, а $N_A(p_{\mathrm{crit}})$ — к достижению заданной вероятности успешной атаки в локальной модели накопления информации. Эти величины имеют разный смысл и должны использоваться совместно, но не взаимозаменяемо.

Модельные примеры показывают две крайние ситуации: детерминированную утечку через среднее значение, где два близких агрегата дают алгебраическую определённость, и многоканальную шумовую утечку, где вероятность восстановления зависит от эффективной дисперсии, набора наблюдаемых ответов и выбранной точности. При этом N_{rank}, N^*_{det} и $N_A(p_{\mathrm{crit}})$ относятся к разным уровням описания и не должны подменять друг друга.

Подход является диагностическим и регламентным, а не абсолютной гарантией. Он позволяет в выбранном испытательном режиме установить, достигает ли восстановление заданного уровня обнаруживаемости, какова вероятность раннего успеха атаки и какие компоненты наблюдаемого выхода дают основной вклад. Если

испытательный режим меняется, меняется и область применимости вывода.

Практический отчёт должен включать N^*_{det}, $N_A(p_{\mathrm{crit}})$, функцию $p_A(N)$, оценку σ^2_{eff}, вклады каналов c_j или c_j/C, выбранный N_{limit}, параметры регуляризации, ядро K, число фолдов кросс-валидации, число перестановок и список предположений о данных, шуме, классе функций и доступной внешней информации. Только такой отчёт позволяет воспроизвести вывод и понять, какие изменения конвейера требуют повторной оценки.

Предложенный подход задаёт последовательную схему анализа косвенной утечки:

ИТ-конвейер → обратная задача → регуляризация
→ статистическая проверка → инженерные параметры управления.

Каждый переход сохраняет связь между математической моделью и эксплуатационным решением. Конвейер задаёт наблюдаемые выходы, обратная задача описывает восстановление скрытого атрибута, регуляризация повышает устойчивость восстановления, статистическая проверка отделяет зависимость от случайности, а инженерная интерпретация переводит результат в лимиты, шум, набор выходов и частоту обновления.

Метод особенно полезен там, где риск нужно объяснить не только специалисту по математическим моделям, но и владельцу процесса. Администратор получает язык для ответа на прикладные вопросы: сколько наблюдений допустимо, какой выход даёт основной вклад, насколько помогает зашумление, когда нужен повторный аудит и какой запас следует оставить относительно порогов. При этом корректная формулировка остаётся осторожной: риск удерживается ниже заданного уровня при принятых предположениях, а не исчезает безусловно.

Поэтому итоговый управленческий вывод должен строиться не вокруг одного числа, а вокруг набора согласованных характеристик. $R^2_{\mathrm{CV}}(N)$ показывает качество восстановления, $p_{\mathrm{perm}}(N)$ — статистическую значимость, $p_A(N)$ — вероятность раннего успеха, N^*_{det} и $N_A(p_{\mathrm{crit}})$ — разные пороговые режимы, а N_{limit} — выбранное регламентное ограничение с запасом. Совместное использование этих величин позволяет принимать решения о настройке конвейера и контроле доступа в терминах измеримого риска.

Глава 4. О ДОВЕРЕННОСТИ В ГИБРИДНЫХ СИСТЕМАХ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

С. С. Буянов

Под гибридными системами искусственного интеллекта здесь понимаются вычислительные системы, в которых выполнение технологических процессов обеспечивается совместным участием автономных агентов, программных компонентов, моделей, внешних инструментов, вычислительных узлов и человека. Взаимодействие в таких системах выходит за пределы заранее установленной доверенной части: отдельные компоненты могут принадлежать разным владельцам, функционировать на различных платформах, иметь неодинаковый уровень автономности и обмениваться данными через неоднородные каналы связи.

До сих пор в подходах к построению доверенных вычислительных систем основное внимание уделялось целостности программного обеспечения, аутентификации и контролируемой среде функционирования [61]. Такой подход остаётся принципиально верным и важным, однако он недостаточен для гибридных систем ИИ, поскольку в них невозможно заранее установить неизменный состав участников, полностью контролировать каналы связи и гарантировать доверенность каждого компонента.

Особенность гибридных систем ИИ состоит в том, что недоверенность может проявляться не только на уровне аппаратуры, программного обеспечения или канала, но и на уровне агентного действия: агент может ошибиться, делегировать задачу неподходящему участнику, использовать скомпрометированный

инструмент, нарушить политику контекста, раскрыть приватные данные или стать частью скрытой координации. Поэтому доверенность такой системы не может быть сведена к сертификату узла, статическому уровню доступа или защищённому каналу. Современные исследования по безопасности мультиагентных систем показывают, что при взаимодействии автономных агентов возникают угрозы, которые не сводятся к безопасности одного агента: скрытые коллюзии, роевые атаки, каскадное распространение ошибок, обход ограничений распространения доверия, утечки приватных данных и подрыв механизмов надзора [62].

Научная проблема, решение которой предлагается в этой главе, состоит в отсутствии формальных моделей и инженерных подходов, позволяющих формировать и поддерживать доверенные сегменты в открытых гибридных системах ИИ при наличии недоверенных узлов, автономных агентов, внешних инструментов, изменяющихся контекстов взаимодействия и угроз коллективной деградации.

В главе рассмотрены модели, методы и механизмы динамического формирования, поддержания, проверки и восстановления доверенности в гибридных системах ИИ, основанные на наблюдаемом поведении агентов, верифицируемых технологических цепочках, трейлерах безопасности, цифровых феромонах, распределённой памяти доверия и контекстно-зависимых политиках принятия решений, и представлена формальная модель и архитектурные механизмы динамической доверенности, обеспечивающие формирование и устойчивое функционирование доверенных сегментов в гибридных системах ИИ без единого доверенного центра. Эта формальная модель может применяться при проектировании систем агентов на основе больших языковых моделей, периферийного ИИ, федеративного обучения, автономных систем киберзащиты, интеллектуальных городских инфраструктур, промышленных систем управления и платформ агентного взаимодействия.

Особенность предлагаемого подхода в том, что доверенность гибридной системы ИИ определяется не статической доверенностью

отдельного узла, а проверяемостью цепочки действий автономных агентов в заданном контексте.

4.1. ТЕОРЕТИЧЕСКИЕ ОСНОВАНИЯ ДОВЕРЕННОСТИ ОТКРЫТЫХ СИСТЕМ

Теоретические основания доверенности открытых систем формируются на пересечении трёх направлений: доверенных вычислительных систем, защищённых информационных технологий и моделей доверия в распределённых средах. Принципиально, что доверенность не отождествляется ни с защищённостью, ни с надёжностью, ни с проверенностью отдельного компонента. Она рассматривается как возможность обоснованно полагаться на корректность поведения системы в заданных условиях и при заданной технологии выполнения операций.

4.1.1. Ретроспективный анализ развития доверенных систем

Доверенные системы проектировались и создавались в условиях, когда системы представляли собой локальные вычислительные комплексы с полностью контролируемым периметром [61]. Узлы находились в единой контролируемой зоне, администратор мог контролировать каналы связи, аппаратные компоненты и программное обеспечение.

Концепция доверенной вычислительной среды рассматривалась как логическое развитие идеи изолированной программной среды (далее — ИПС) [63]. Термин доверенной вычислительной среды (далее — ДВС) был введён как расширение модели ИПС и подразумевал, что такая система включает только те узлы, подлинность и целостность которых подтверждены [61]. Реализация данных требований предполагает использование аппаратных средств защиты, выполняющих функции резидентного компонента безопасности [7].

На начальном этапе подобная модель оказывалась релевантной для локальных СВТ и впоследствии стала основой механизмов формирования доверия в корпоративных системах, где

обеспечивался полный контроль над каналами связи, программным обеспечением и аппаратными средствами.

Впоследствии с развитием облачных технологий, в том числе онлайн-банкинга, возникла необходимость переосмысления модели доверенной системы. Это способствовало признанию, что доверенность — не всегда постоянное свойство. В некоторых сценариях она должна задаваться на время сессии или канала связи, особенно в мобильных или облачных средах [14].

В этих условиях формируется концепция доверенного сеанса связи [14] (далее — ДСС), при котором:

1. узлы аутентифицируются при создании ДСС;
2. каналы защищаются на время соединения;
3. среда функционирования считается доверенной лишь на период сеанса.

Однако эти представления предполагают возможность централизованной подготовки и унификации среды, то есть ее ограничения. В открытых системах подобный подход принципиально нереализуем в силу их гетерогенности. Обеспечение всех участников системы аппаратными средствами контроля целостности экономически и организационно невозможно, так как противоречит самой природе открытой среды.

Рассмотрим пример формирования доверенного сегмента в открытой городской инфраструктуре. В городе произошло ДТП на перекрёстке, что привело к блокировке движения. Необходимо в течение 3–5 минут перенаправить транспортные потоки, чтобы предотвратить образование масштабной пробки. Для решения такой задачи камеры видеонаблюдения с системой компьютерного зрения должны обнаружить аварию и оценить плотность потока подъезжающих автомобилей. Городские серверы обработки данных должны провести симуляцию потоков по полученным данным от камер и выдать план по распределению потоков движения. Управляемые светофоры на смежных перекрёстках должны получить план и перенаправить автомобильный поток.

Применение существующих подходов в сценариях, подобных описанному, оказывается принципиально невозможным в силу ограничений. Модель ДВС, предполагающая единую контролируемую среду с гарантированной целостностью всех узлов,

не может быть реализована из-за технологической и административной гетерогенности участников. Разнородность аппаратных платформ, операционных систем и политик безопасности делает создание такой среды экономически и технически нецелесообразным. Даже в случае частичной реализации ДВС на отдельных узлах система остаётся уязвимой, например, к компрометации неконтролируемых каналов связи, что нарушает принцип доверенности всех компонентов.

Организация ДСС, основанная на предварительной аутентификации и установке защищённых каналов между известными участниками, также не применима в условиях динамически меняющегося состава узлов и неконтролируемых каналов связи. Непредсказуемость набора устройств, вовлекаемых в решение задачи, квадратичная сложность масштабирования защищённых соединений приводят к тому, что попытка развёртывания ДСС может парализовать систему из-за временных затрат.

Таким образом, существующие подходы демонстрируют неадаптивность к условиям открытых систем, где доверие не может быть предустановленным. Это обусловливает необходимость разработки нового подхода, который должен обеспечивать децентрализованную самоорганизацию узлов во временные логические группы.

Из вышесказанного следует, что в данный момент возникает потребность в описании поведения систем, состоящих из множества элементов, действующих в случайной среде и не имеющих центрального управляющего, но следующих общей цели.

В этом направлении развивается теория MAC. Пересечение концепций ДВС и MAC может лечь в основу создания доверенных сегментов открытых систем.

4.1.2. Доверенная среда функционирования в открытой системе

Для открытых систем становится актуальной задача формирования «областей доверенности» внутри открытой среды. В таких системах не существует единого контролируемого периметра, но существует необходимость координированной работы

нескольких узлов, чей совместный результат должен быть столь же надёжен, как и в локальных ДВС. Именно здесь возникает потребность в определении доверенной среды функционирования (далее — ДСФ), так как создание ДСФ вне контролируемой зоны сопряжено с рядом существенных проблем, требующих комплексного решения. Приведём определения из [61]:

• Доверенная среда функционирования — взаимодействующая совокупность доверенных узлов обработки данных;

• Доверенный узел — выделенная совокупность аутентифицированных и целостных технических средств с проверенным ПО.

Такие определения позволяют чётко различать два уровня доверенности: доверенность локального узла и доверенность распределённой группы узлов. В открытой системе невозможно организовать доверенность глобально, однако её можно обеспечить локально — на уровне каждого узла и на уровне каждого конкретного канала взаимодействия.

Когда узел функционирует в открытой среде и должен взаимодействовать с другими такими же узлами, предъявлять требования нужно не только к самим узлам, но и к каналам связи. Влиять на транспортную инфраструктуру часто невозможно, поэтому необходимо быть уверенными, что данные в процессе передачи не были скомпрометированы.

Из этого следуют два обязательных условия формирования доверенной среды функционирования в открытой системе:

1. наличие локальной доверенной вычислительной среды на каждом узле;

2. обеспечение целостности данных, передаваемых между доверенными узлами.

4.1.3. Доверенный узел и его свойства

Внутренним элементом доверенной среды является доверенный узел — выделенная совокупность аутентифицированных и целостных технических средств с проверенным программным обеспечением [61]. Доверенный узел

характеризуется подтверждённым состоянием своего программного обеспечения и наличием механизмов предотвращения и детектирования несанкционированных изменений [7]. Это создаёт основу для выполнения операций в соответствии с заданной технологией и исключает возможность произвольного отклонения поведения узла от предписанных правил.

Доверенность узла является не просто характеристикой состояния, но и условием включения узла в доверенный сегмент.

4.1.4. Доверенный сегмент открытой системы

Перенос устоявшихся представлений о доверенной среде на открытую систему сталкивается с ограничениями. Как следует из вышеизложенного, в подобных системах:

- невозможно обеспечить гарантированную идентификацию всех участников;
- невозможно контролировать каналы связи;
- состав участников не фиксирован и может меняться быстро;
- отсутствует единый администратор, который мог бы обеспечить мониторинг всей среды.

В результате доверенность в открытой системе невозможно обеспечивать централизованно и статически. Вместо этого она должна формироваться динамически — в реальном времени, по мере возникновения инцидента; распределённо — без обращения к единому центру, на основе прямых оценок узлов друг другом; и контекстно — в зависимости от конкретной задачи, например, оценка затора или управление светофором. Такой подход опирается на наблюдаемое поведение узлов, историю их взаимодействий и локальные политики принятия решений.

Доверенный сегмент — это группа доверенных узлов в открытой системе, для которых можно обеспечить контролируемый и проверяемый обмен информацией с подтверждением источника и целостности данных, а также корректность выполнения информационной технологии. В отличие от локальной доверенной среды, доверенный сегмент не совпадает с физическим контуром сети; он формируется логически на основе свойств узлов и соблюдаемой технологии.

Для доверенного сегмента характерны следующие признаки:

- каждый узел обладает подтверждённой доверенностью;
- каждый узел имеет знания о состоянии своих доверенных соседей;
- между узлами осуществляется контролируемый и проверяемый обмен информацией с подтверждением источника и целостности данных;
- отсутствует единый управляющий центр;
- все участники следуют единому набору технологических правил.

Эти признаки определяют распределённый характер сегмента и его способность функционировать в условиях отсутствия глобального управления.

4.1.5. Принципы формирования доверенного логического сегмента

В открытых системах невозможно обеспечить физическую изоляцию и контроль всех каналов связи. Поэтому доверенный сегмент должен иметь логическую природу: его границы определяются не аппаратными средствами, а выполнением определённых требований к доверенности, взаимодействию и применению информационной технологии.

Физический сегмент определяется топологией сети и административными ограничениями, а логический сегмент определяется соблюдением технологической цепочки и взаимным подтверждением доверенности узлов. Это позволяет объединить в один сегмент узлы, географически распределённые и принадлежащие различным административным доменам, при условии, что они поддерживают согласованные механизмы доверия.

Формальное определение доверенного сегмента

Для последующего анализа необходимо формально определить доверенный сегмент.

Пусть открытая система включает множество узлов $U = \{U_1, U_2, \dots, U_n\}$. Доверенный сегмент — это подмножество $TS \subseteq U$, удовлетворяющее следующим условиям:

- каждый узел обладает свойством доверенности и подтверждённой целостностью;
- обмен данными можно проконтролировать и проверить целостность этих данных с подтверждением источника;
- узлы поддерживают выполнение единой информационной технологии;
- взаимодействие организовано децентрализованно;
- каждый узел располагает сведениями о состоянии своих доверенных соседей.

Такое определение позволяет рассматривать доверенный сегмент как абстрактную, формально определённую структуру, на основе которой можно строить модель доверия.

Таким образом, доверенность открытой системы не может быть задана как глобальное состояние всей инфраструктуры. Она должна формироваться локально, динамически и технологически: локально — потому что каждый участник располагает лишь частичной информацией; динамически — потому что состав и поведение участников меняются; технологически — потому что корректность результата определяется не только состоянием узлов, но и порядком операций.

4.2. ДОВЕРЕННЫЙ ЛОГИЧЕСКИЙ СЕГМЕНТ ОТКРЫТОЙ СИСТЕМЫ

Понятие доверенного сегмента является центральным связующим элементом между доверенными вычислительными средами и гибридными системами ИИ. В отличие от физически изолированной подсети, такой сегмент определяется не сетевыми границами, а условиями технологической проверяемости, подтверждаемого происхождения данных и согласованности действий участников.

4.2.1. Защищённая информационная технология как основа доверенного взаимодействия

В любой системе ключевую роль играет порядок выполнения операций, поскольку именно он определяет применяемую технологию [11]. Даже корректные действия, выполненные вне

установленной последовательности, приводят к искажению результата: как в бытовом примере — при жарке картофеля масло добавляют на сковороду до начала жарки, иначе продукт пригорает. Аналогично и в вычислительных системах: если технологическая цепочка нарушена, итоговые данные теряют смысл.

Технологическая цепочка в информационной технологии должна включать всё, что циркулирует в системе, — будь то электронные документы, несущие сведения, или управляющие сигналы, определяющие ход процессов. Если порядок и состав операций не фиксируются и не контролируются, невозможно гарантировать корректность результата, доказуемость его происхождения и устойчивость системы к подменам и несанкционированным изменениям.

Ситуацию можно сравнить с игрой в «испорченный телефон»: когда отсутствуют средства проверки корректности передачи, последовательные искажения приводят к тому, что итоговое сообщение не имеет ничего общего с исходным.

В открытой вычислительной среде происходит то же самое: каждый узел может трактовать операцию по-своему, а другие участники не располагают механизмами проверки. Как следствие, результат может быть как корректным, так и испорченным, либо намеренно подмененным.

В рамках данной работы взаимодействие узлов рассматривается как технологический процесс, результат которого считается корректным при выполнении следующих условий: операции выполняются в фиксированной последовательности, технология их выполнения однозначно определена, результаты воспроизводимы на различных узлах, соответствие фактического процесса эталонной технологии поддаётся проверке, а история преобразований может быть восстановлена и подтверждена.

Нарушение любого из указанных условий приводит либо к искажению результата, либо к утрате возможности доказать его корректность и происхождение. Следовательно, все проблемы взаимодействия узлов в открытой системе сводятся к нарушениям перечисленных условий, что позволяет рассматривать их как необходимый минимум. Перечень проблем, возникающих при взаимодействии узлов в открытой системе, приведён в таблице 4.1.

Таблица 4.1. Проблемы взаимодействия узлов в открытой системе

Проблема	Описание
Невозможность гарантировать порядок выполнения операций	Порядок действий не фиксируется, поэтому корректные шаги могут давать искажённый результат.
Отсутствие проверки эквивалентности технологий	Нет формализованного способа сравнить фактическую цепочку операций с эталонной.
Невоспроизводимость результатов	Идентичные данные на разных узлах дают разные результаты из-за скрытых изменений в процессе обработки.
Риск фальсификации процесса формирования документа	Возможна подмена операций, внедрение лишних операций или пропуск обязательных, что остаётся незамеченным.
Недоказуемость корректности действий агентов	Узлы не могут подтвердить, что каждый участник исполнил предписанные операции.
Отсутствие трассируемости	История преобразований не фиксируется, невозможно восстановить, кто и какие действия выполнял.
Отсутствие механизма обнаружения несанкционированных воздействий	Отсутствие механизмов контроля позволяет незаметно подменять или дублировать операции.
Низкая устойчивость к ошибкам и атакам	Без встроенной верификации система уязвима как к случайным сбоям, так и к преднамеренным воздействиям.

Решение перечисленных проблем является условием построения доверенного сегмента. Если удаётся обеспечить фиксацию и контроль технологической цепочки, доверие переносится с отдельных аппаратных или программных компонентов на сам процесс взаимодействия. Таким образом, доверенный сегмент перестаёт зависеть от доверенной среды на

каждом узле: даже если часть оборудования не имеет верифицированной аппаратной базы, формализованная технология взаимодействия обеспечивает уверенность в корректности, целостности и воспроизводимости результата.

Ключевое отличие доверенного сегмента от обычной распределённой системы состоит в том, что акцент смещается с доверия к аппаратуре и каналам связи на доверие к формализованной технологической схеме, которая контролируется и верифицируется на каждом этапе.

Одним из механизмов, обеспечивающих такое свойство, выступает механизм трейлеров безопасности (ТБ) [11]. Их назначение — фиксировать порядок действий, обеспечивать верификацию технологической цепочки и контролировать целостность операций.

Каждый трейлер безопасности определяется как:

$$T_i = \{\text{заголовок,значение}\}. \quad (1)$$

где:

• заголовок содержит описание операции и алгоритм формирования ТБ;

• значение вычисляется на основе исходных данных и параметров операции.

Электронный документ в этом случае представляется в виде:

$$\text{ЭлД} = \{\text{сведения,последовательность ТБ}\}. \quad (2)$$

Подлинность документа подтверждается при выполнении следующих условий:

1. эквивалентность технологии (последовательности ТБ) базовому эталону;

2. целостность данных, то есть корректность значений ТБ.

Эквивалентность определяется допустимыми подстановками. Для каждого класса документов задаётся базовая технология, и любая применяемая информационная технология проверяется на эквивалентность этой базовой.

Таким образом, механизм трейлеров безопасности обеспечивает:

- фиксацию порядка действий;
- верификацию технологических цепочек;
- полную трассируемость операций.

И как следствие, в таком случае защита информационной технологии сводится к тому, чтобы можно было обосновать подлинность и целостность результата за счёт контроля над всей последовательностью преобразований. Благодаря этому защищенная информационная технология (ЗИТ) становится основой для оценки доверия к применяемой технологии и средством поддержания предсказуемости поведения узлов в сегменте.

Вместе с тем применение механизма ТБ предполагает выполнение ряда условий, которые не связаны с конкретной реализацией трейлеров, но определяют среду, в которой такой механизм может функционировать.

К числу этих условий относятся:

1. Наличие механизмов надёжного связывания трейлеров безопасности с результатами операций и средствами их хранения и передачи таким образом, чтобы последовательность трейлеров не могла быть изменена или частично утрачена без обнаружения.

2. Наличие формально заданной эталонной технологии, допускающей проверку эквивалентности фактической цепочки операций.

3. Подготовка системы к функционированию в условиях недоверенной среды: допускать наличие узлов с различной степенью доверенности, обеспечивать локальную верификацию технологической цепочки и не полагаться на единую доверенную точку.

Выполнение этих условий создаёт необходимые предпосылки для применения трейлеров безопасности в составе защищённой информационной технологии и позволяет рассматривать их как практический инструмент построения доверенного взаимодействия.

Для гибридных ИИ-систем описанная логика сохраняется, но состав операций расширяется. Технологическая цепочка может включать формирование запроса к модели, обработку контекста, выбор инструмента, обращение к внешнему источнику, делегирование другому агенту, проверку результата, фиксацию ответа и передачу управления человеку. Поэтому доверенный логический сегмент должен учитывать не только вычислительные операции, но и агентные действия.

4.3. МУЛЬТИАГЕНТНАЯ ПРИРОДА ДОВЕРЕННОГО СЕГМЕНТА

Доверенный сегмент в открытой среде имеет мультиагентную природу. Он формируется как множество автономных участников, каждый из которых действует на основании локальной информации, но при этом должен поддерживать общую технологическую цель и не разрушать проверяемость результата. В этом смысле теория мультиагентных систем задаёт поведенческий язык, а теория защищённых информационных технологий — механизм верификации.

4.3.1. Краткий обзор мультиагентных систем

Мультиагентные системы (МАС) представляют собой особый класс сложных систем, состоящих из множества взаимодействующих агентов, действующих в случайной среде. Агентом называют автономную сущность — реальную или виртуальную, способную воспринимать внешнюю среду через сенсоры, воздействовать на неё через эффекторные механизмы, а также обмениваться информацией и координировать действия с другими агентами [64]. Такой подход позволяет описывать поведение системы на микроуровне, фиксируя особенности отдельных компонентов, а затем выявлять эмерджентные эффекты на макроуровне.

Важнейшим свойством МАС является самоорганизация. По мнению исследователей, именно способность агентов согласовывать свои действия без централизованного управления обеспечивает системное качество, отличающее МАС от простого набора элементов. В [12] подчёркивается: если нет самоорганизации, то нет и МАС. Это свойство проявляется в способности агентов формировать устойчивые структуры, поддерживать согласованность операций и адаптироваться к изменениям среды.

Приведём полное определение МАС из источника [12]:

МАС — это совокупность агентов, каждый из которых наделён полным набором следующих взаимно дополняющих и взаимно обусловленных свойств: быть организацией; уметь формулировать собственные цели; быть способным планировать своё поведение; иметь средства для выполнения планов; быть погружённым в

случайную среду; обладать способностью вступать в обмен ресурсами, энергией или информацией с другими агентами.

Существует несколько классификаций агентов в зависимости от уровня их «интеллектуальности» [64]:

• простые рефлекторные агенты — реагируют только на текущее восприятие;

• агенты с внутренней моделью — учитывают историю состояний и действий;

• агенты, основанные на целях — ориентируются на достижение желаемых состояний;

• агенты, основанные на полезности — выбирают действия, максимизирующие функцию полезности;

• обучающиеся агенты — способны адаптироваться в неизвестной среде, накапливая знания о стратегиях поведения.

Широко применяется и модель убеждений, желаний и намерений (BDI), которая формализует когнитивные характеристики агента. В ней «убеждения» отражают знания о среде, «желания» задают цели, а «намерения» определяют действия для их достижения.

Примеры МАС встречаются в самых разных областях:

• в биологии — колонии муравьёв и пчёл, где самоорганизация обеспечивает выживание сообщества;

• в робототехнике — рой роботов-доставщиков, выполняющих доставку посылок.

4.3.2. Предпосылки использования теории МАС как основы доверенного сегмента

Использование теории мультиагентных систем в качестве теоретической базы для построения доверенных сегментов объясняется тем, что она предоставляет формальный аппарат для описания и анализа поведения автономных участников, взаимодействующих в неопределённой и динамической среде. Заложенные в данной теории предпосылки и допущения соответствуют реальным условиям функционирования открытых систем, где отсутствует единый управляющий центр, а узлы могут

свободно присоединяться, уходить, взаимодействовать и при этом изменять стратегии и характеристики.

Во-первых, автономность агента позволяет рассматривать каждый узел доверенного сегмента как независимого субъекта, принимающего решения на основе локально доступной информации. Это соответствует принципу распределённой доверенности, где отсутствие центра компенсируется локальными механизмами самоорганизации.

Во-вторых, взаимодействие агентов описывается в терминах обмена информацией, согласования действий и коллективного решения задач. В доверенном сегменте эти процессы прямо связаны с обеспечением целостности данных, выполнением технологической цепочки и достижением согласованного результата при отсутствии полного контроля над средой.

Наконец, для МАС характерны механизмы формирования устойчивых групп, коалиций, подсетей. В доверенном сегменте такие структуры необходимы: узлы вынуждены перераспределять доверие, учитывать историю взаимодействия и корректировать решения при изменении состава участников. Таким образом, существование доверенного сегмента — это пример самоорганизующегося множества агентов, объединённых едиными правилами и целью.

Из вышесказанного можно сделать вывод, что мультиагентный подход позволяет:

- интерпретировать доверенный сегмент как множество согласованно действующих агентов;
- описать динамику доверия как реакцию на наблюдаемое поведение;
- моделировать распространение информации между узлами;
- формализовать процессы самоорганизации и восстановления структуры после нарушений;
- исследовать устойчивость сегмента к ошибочным, случайным и злонамеренным воздействиям.

4.3.3. Основы формализации доверенного сегмента в контексте МАС

Формализовать переход от мультиагентной системы к доверенному сегменту можно, рассматривая доверенный сегмент как подмножество агентов случайной среды, удовлетворяющих определённым требованиям доверенности и соблюдающих общую защищённую информационную технологию.

Пусть имеется мультиагентная система:

$$\mathrm{MAS} = \langle A, E_s, V \rangle. \quad (3)$$

где

$A = \{a_1, \dots, a_n\}$ — множество агентов;

$E_s \subseteq A \times A$ — отношения между агентами;

V — среда, в которой они действуют.

Каждый агент обладает набором свойств, перечисленных ранее: автономность, способность к целеполаганию, планированию, обмену информацией и адаптации.

Доверенный сегмент определяется как выделенная подсистема в составе МАС:

$TS \subseteq A,$ (4)

удовлетворяющая следующим формальным критериям.

1. Взаимная доверенность агентов.

Каждая пара агентов должна иметь между собой подтверждённое состояние доверия

$$\forall a_i, a_j \in TS: \quad Trust(a_i, a_j) \geq T_{\min} \ \wedge \ Trust(a_j, a_i) \geq T_{\min}. \quad (5)$$

где $T_{\min}$ — минимальный порог доверия.

2. Согласованность технологической цепочки.

Агенты обязаны выполнять операции в соответствии с общей защищённой информационной технологией.

3. Проверяемость взаимодействий.

Каждое взаимодействие между агентами сегмента должно быть проверяемым.

4. Децентрализованность.

Внутри доверенного сегмента не существует агента с абсолютной властью.

5. Локальная наблюдаемость.

Каждый агент имеет доступ к информации о состоянии своих доверенных соседей.

Доверенный сегмент можно рассматривать как подмножество открытой системы, формируемое за счёт динамического доверия между узлами, согласованной технологии взаимодействия, проверяемости взаимодействия и отсутствия централизованного контроля. В этом контексте теория MAC задаёт общую поведенческую рамку открытой системы, а доверенный сегмент представляет собой её часть, в которой агенты обеспечивают как технологическую, так и поведенческую корректность.

4.3.4. Анализ жизненного цикла доверенного сегмента в контексте MAC

Жизненный цикл доверенного сегмента представлен на рисунке 4.1. Его условно можно разделить на четыре этапа: инициализация, самоорганизация, развитие и деградация.

Рис.4.1. Жизненный цикл мультиагентной системы

На старте или на этапе инициализации доверенный сегмент формируется: узлы проверяют свою готовность и проводят процедуру идентификации. Этот этап закладывает основу последующих процессов и обеспечивает готовность системы к взаимодействию.

После инициализации сегмент переходит к этапу самоорганизации. На данном этапе узлы устанавливают взаимные связи, определяют соседей, согласовывают правила обмена информацией и формируют первичную структуру взаимодействия.

На этапе развития доверенный сегмент функционирует как целостная система. Узлы обмениваются данными, координируют действия и совместно выполняют задачи. Накапливается опыт взаимодействия, происходит распределение ресурсов, формируются механизмы планирования и поддерживается баланс между узлами.

Со временем сегмент может переходить к этапу деградации. Характерными признаками являются снижение согласованности действий и уменьшение активности отдельных узлов, нарушение

распределения ресурсов. В результате снижается устойчивость сегмента и возрастает потребность во внешнем вмешательстве или внутренних механизмах стабилизации для сохранения его работоспособности.

Таким образом, жизненный цикл доверенного сегмента включает в себя как процессы его создания (инициализация и самоорганизация), так и процессы функционирования (развитие и деградация). Соответственно, возникающие трудности можно разделить на две группы: проблемы, связанные с построением сегмента, и проблемы, проявляющиеся в ходе его дальнейшей работы.

4.3.5. Проблемы создания доверенного сегмента

На этапе построения доверенного сегмента — то есть инициализации и самоорганизации взаимодействующей совокупности доверенных узлов обработки данных — возникает несколько ключевых проблем.

Во-первых, для признания узла доверенным необходимо обеспечить, что его состав и структура не подверглись изменениям. Это требовалось бы при использовании текущих подходов к моделям доверия, основанных на внедрении аппаратных механизмов контроля целостности, например, резидентного компонента безопасности [7, 14].

Во-вторых, в открытой системе требуется реализация механизма обнаружения соседних доверенных узлов для того, чтобы сегмент смог сформироваться. Это предполагает решение задачи идентификации агентов, способных к установлению доверенных связей.

На следующем этапе необходимо обеспечить установление доверительных отношений между доверенными узлами, что предполагает реализацию механизма взаимной аутентификации. Данная задача становится нетривиальной в открытой системе из-за отсутствия централизованного удостоверяющего центра.

Из всего вышесказанного можно выделить перечень потенциальных проблем, возникающих при создании доверенного сегмента:

- Обеспечение целостности используемого ПО;
- Отсутствие механизма идентификации и взаимной аутентификации доверенных узлов.

Без решения указанных проблем система оказывается неспособной к формированию доверенной среды функционирования в рамках открытой среды.

4.3.6. Проблемы функционирования доверенного сегмента

После завершения этапа создания и установления доверенных связей между узлами, то есть этапов инициализации и самоорганизации, система переходит к процессам планирования и коммуникации. На этом этапе проявляются внутренние проблемы, которые способны привести к утрате согласованности действий агентов.

Одной из таких проблем является коллапс первого рода — ситуация, при которой узлы утрачивают возможность формировать совместные планы или выполнять координированную деятельность из-за переполнения каналов связи. В контексте доверенного сегмента под планом будем понимать прогнозируемый объём вычислительных или сетевых ресурсов, который узел готов затратить на выполнение задач. Обозначим данный показатель как PL — запланированная нагрузка узла. Если на старте PL принимает завышенные значения, это может привести к разрушению общей стратегии вследствие дезинтеграции индивидуальных планов узлов, что, в свою очередь, вызывает коллапс первого рода [12].

Даже в случае избегания указанного коллапса при одинаковых начальных условиях в процессе функционирования возможны асимметрии: один или несколько узлов получают доминирующее положение, концентрируя информацию, ресурсы и принятие решений. В результате остальные узлы становятся зависимыми или пассивными, что ведёт к разбалансировке всей архитектуры доверенного сегмента и снижению его устойчивости.

При обмене данными между узлами критически важно контролировать целостность содержимого и его подлинность, особенно при использовании транспортных каналов, которые в условиях открытой системы априори не могут считаться

доверенными. Отсутствие такого контроля делает систему уязвимой для перехвата сообщений, их модификации или атак повторного воспроизведения.

При длительном функционировании без коррекции показатель достоверности информации между узлами, обозначаемый P^0 , снижается. Если $P^0 = 1$, данные полностью достоверны; если $P^0 = 0$, каждая операция завершается неудачей, и узел не владеет никакой полезной информацией. При $P^0 \to 0$ обмен теряет смысл, поскольку узлы перестают различать достоверные и ложные сообщения. Самоорганизация становится невозможной, и система переходит в состояние хаоса — информационный коллапс второго рода [12].

Коллапсом второго рода не считается ситуация, когда узел перестал функционировать по техническим причинам и в результате этого информация искажается, либо когда данные намеренно подделываются в ходе атаки.

В [12] авторы приводят закономерности функционирования мультиагентных систем. Спроецируем эти рассуждения на доверенный сегмент открытой системы, введя параметры эволюционно-симуляционной модели:

• D — коэффициент полезного действия узла, отражающий долю успешно выполненных задач или операций от общего числа попыток;

• K — доля владения ценными ресурсами сегмента, обеспечивающими доверие (например, ключи, сертификаты, пропускная способность доверенных каналов);

• Z — отношение риска избыточной активности к риску недостаточной активности (риск завышения связан с издержками при чрезмерной загрузке, когда PL выше оптимума; риск занижения — с упущенными возможностями при недостаточной активности, когда PL ниже потенциала, заданного K).

Предположим, что в начальный момент все N агентов доверенного сегмента абсолютно одинаковы и обладают одинаковыми значениями параметров, то есть

$$P_i^0 = P_j^0, \quad PL_i = PL_j, \quad D_i = D_j, \quad K_i = K_j, \quad Z_i = Z_j, \quad \forall i, j \in \{1, \dots, N\},\ i \neq j. \quad (6)$$

Согласно [9], в системе начнёт происходить следующее: возникнет разделение агентов по величинам D и K. В результате

ценными ресурсами будут обладать те агенты, чьё значение D выше, за счёт перераспределения ресурсов от агентов с меньшим коэффициентом полезного действия. Темпы этих изменений будут пропорциональны разности D между агентами. При этом значения D и K у одних агентов будут монотонно возрастать, тогда как у других — снижаться. Аналогичные тенденции будут наблюдаться и для PL и P^0, причём изменения продолжатся неограниченно.

В предельном случае у нескольких агентов D достигает максимального значения, в то время как остальные деградируют. Система ориентируется только на «лидеров», а отсутствие механизмов стабилизации делает этот процесс необратимым. В конечном состоянии сегмент можно охарактеризовать следующим образом:

• у одного агента PL , K и D принимают максимальные значения, а $P^0 \to 1$;

• у остальных агентов PL, K и D стремятся к минимуму, а $P^0 \to 0$;

• изменения прекращаются, система достигает стационарного состояния — информационного коллапса второго рода.

Таким образом, доверенный сегмент теряет свойства взаимодействующей системы: большинство агентов становится пассивными, они не участвуют в формировании доверия и не влияют на стратегию сегмента. Доминирующий агент оказывается перегруженным, что делает дальнейшее развитие невозможным без внешнего вмешательства. Деградация становится необратимой, и для сохранения устойчивости необходимы внутренние механизмы регуляции, позволяющие системе восстанавливаться или возвращаться к работоспособному состоянию при нарушениях взаимодействия [12].

Кроме того, возможна подмена или деградация отдельных узлов. Для обеспечения устойчивости доверенного сегмента требуется внедрение механизмов изоляции или исключения агентов, поведение которых выходит за пределы допустимой модели. При отсутствии таких механизмов скомпрометированный узел продолжает участвовать в функционировании системы, нарушая её безопасность и устойчивость [12].

В итоге доверенный сегмент как форма реализации МАС в открытых системах сталкивается с множеством проблем. Из проведённого анализа жизненного цикла можно выделить следующие ключевые трудности:

- доминирование отдельных узлов;
- коллапс первого рода;
- коллапс второго рода;
- отсутствие механизмов стабилизации;
- отсутствие механизмов, подтверждающих достоверность данных из открытых каналов на основе накопленной информации;
- отсутствие механизмов исключения агентов.

4.3.7. Выводы из анализа проблем построения доверенного сегмента

Проведённые наблюдения свидетельствуют о том, что доверенный сегмент в открытой системе сталкивается с двумя группами проблем в течение своего жизненного цикла — проблемами создания и проблемами функционирования. Эти проблемы связаны не только с поведением отдельных узлов, но и с тем, как они координируются, распределяют ресурсы и влияют друг на друга во времени. Нарушения на любом этапе — от инициализации до фазы стабильного развития — могут привести либо к разрушению структуры взаимодействия, либо к потере достоверности данных, либо к доминированию отдельных узлов и утрате согласованности.

Эти наблюдения позволяют перейти к формулировке основной научной задачи, без решения которой доверенный сегмент не может существовать устойчиво. Если рассматривать МАС как множество агентов A, связанных отношениями $E_s \subseteq A \times A$ и действующих в недоверенной среде V, то требуется выделить такое подмножество $TS \subseteq A$, которое сохраняет свои ключевые свойства на протяжении всего жизненного цикла. К таким свойствам относятся:

- соответствие всех взаимодействий согласованной ЗИТ;
- проверяемость и контролируемая целостность передаваемой информации;

• устойчивость сегмента при отсутствии централизованного управления.

Однако выявленные деструктивные эффекты демонстрируют, что без особых подходов к организации подобных систем такие свойства в открытой среде не обеспечиваются. Более того, непрерывное взаимодействие отсутствие предсказуемости и эмерджентные эффекты развития системы приводят к тому, что сегмент может разрушиться даже без внешнего воздействия.

Исходя из этого, формальная задача, которую необходимо решить, заключается в определении таких архитектурных, технологических и поведенческих условий, при которых подмножество агентов TS способно сохранять доверенность, согласованность и устойчивость на всех этапах жизненного цикла несмотря на открытую среду, отсутствие централизованного контроля и эмерджентный характер коллективного поведения.

Для решения этой задачи недостаточно анализировать доверенный сегмент изолированно. Необходимо рассмотреть, как распределённые системы развивались, какие архитектурные принципы и модели взаимодействия оказались жизнеспособными, какие механизмы согласованности применялись, и какие ограничения проявлялись при масштабировании и усложнении вычислений.

Только с учётом эволюции распределённых вычислений можно корректно определить, какие принципы должны лечь в основу архитектуры доверенного сегмента и какие свойства распределённых систем необходимо использовать или усиливать для достижения устойчивости.

4.4. ГИБРИДНЫЕ СИСТЕМЫ ИИ КАК РАЗВИТИЕ ОТКРЫТЫХ СИСТЕМ

Гибридная ИИ-система является дальнейшим развитием открытой мультиагентной системы. В ней агентность связана не только с автономностью программного узла, но и с использованием моделей, внешних источников данных, протоколов межагентного взаимодействия и человеческого участия. Поэтому доверенный

сегмент должен быть перенесён с уровня распределённых узлов на уровень проверяемых агентных действий.

4.4.1. Понятие гибридной системы ИИ

Под гибридной системой ИИ в настоящей работе понимается открытая система, в которой совместно функционируют автономные агенты, модели машинного обучения, агенты на основе больших языковых моделей, программные сервисы, источники данных, вычислительная инфраструктура и человек. Гибридность проявляется одновременно на уровне компонентов, целей, владельцев, протоколов и способов принятия решений.

В отличие от распределённых систем, гибридные системы искусственного интеллекта содержат компоненты, поведение которых не является полностью детерминированным и может зависеть от контекста обработки данных. Агент на основе большой языковой модели может строить план, взаимодействовать с другими агентами и корректировать поведение в зависимости от результатов наблюдения [65, 66].

4.4.2. Агенты на основе больших языковых моделей и агентные рабочие процессы

Агент на основе большой языковой модели представляет собой не просто языковую модель, а композицию из модели, системных инструкций, политик, механизма планирования, среды выполнения и журнала действий. В современных агентных фреймворках особое значение имеют делегирование, передачу управления, защитные ограничения, наблюдаемость и сохранение состояния выполнения[1].

Для вопроса доверенности это принципиально: доверять нужно не только ответу модели, но и всей траектории агентного выполнения.

[1] OpenAI. OpenAI Agents SDK Documentation. — URL: https://openai.github.io/openai-agents-python/ (дата обращения: 14.05.2026).

4.4.3. Периферийный ИИ и федеративное обучение

Периферийный ИИ создаёт естественную среду для динамической доверенности: вычисления выполняются близко к источнику данных, состав устройств меняется, ресурсы ограничены, а центральная проверка каждого действия невозможна. В таких условиях доверие должно учитывать локальные наблюдения, задержки, качество выполнения операций, физический контекст и риск компрометации узлов [67].

Федеративное обучение также требует переноса доверия с самого участника на результаты его работы. Хотя участник не раскрывает исходные данные, он передаёт изменения параметров модели. Такие изменения могут быть ошибочными, намеренно искажёнными, сформированными на недостоверных данных либо полученными без реального участия узла в обучении [68–70].

4.4.4. Агентные протоколы и открытая среда

Появление протоколов MCP и A2A делает задачу доверенности особенно актуальной. MCP предназначен для подключения ИИ-приложений к системам, где находятся данные и инструменты[2]. A2A ориентирован на безопасный обмен информацией и координацию действий между агентами[3]. Эти протоколы расширяют возможности агентных систем, но одновременно увеличивают поверхность атаки: агент получает доступ к новым инструментам, данным и другим агентам.

Следовательно, доверенный сегмент гибридной ИИ-системы должен поддерживать только идентификацию агента, проверяемость вызовов инструментов, контроль делегирования, журналирование межагентных сообщений.

---

[2] Model Context Protocol. Specification. Revision 2025-11-25. — URL: https://modelcontextprotocol.io/specification/2025-11-25 (дата обращения: 10.05.2026).

[3] Model Context Protocol. Specification. Revision 2025-11-25. — URL: https://modelcontextprotocol.io/specification/2025-11-25 (дата обращения: 10.05.2026).

4.4.5. Уровни автономности

Таблица 4.2. Уровни автономности агента и требования доверенного сегмента

Уровень	Описание	Решение доверенного сегмента
A0	Только чтение и анализ	Разрешён доступ к данным без действия во внешней среде
A1	Предложение решения	Агент формирует рекомендацию, действие выполняет человек
A2	Ограниченное использование инструментов	Разрешены заранее заданные инструменты и проверяемые операции
A3	Автономное выполнение	Агент выполняет действия при достижении порога доверия
A4	Делегирование другим агентам	Разрешено только при наличии трейлера делегирования и допустимого риска
A5	Внешнее воздействие	Требуется усиленный контроль, участие человека в контуре управления (human-in-the-loop) или независимая верификация

Существенное отличие таких систем состоит в том, что ошибка или нарушение может возникать не только в канале связи и не только в программном обеспечении узла, но и в контекстном поведении агента. Агент может корректно пройти идентификацию и при этом выполнить недопустимое действие: неверно интерпретировать контекст, обратиться к неподходящему инструменту, передать задачу другому агенту без достаточных оснований или включить в технологическую цепочку недостоверный источник.

4.5. МОДЕЛЬ УГРОЗ ДОВЕРЕННОСТИ

Модель угроз доверенности должна охватывать как исходные угрозы открытых распределённых систем, так и новые угрозы, возникающие из-за автономности ИИ-агентов. Поэтому в настоящей работе угрозы рассматриваются не только как компрометация узла или канала, но и как нарушение проверяемой технологической цепочки, искажение контекста, отравление памяти, ложное делегирование, скрытая координация и каскадная деградация сегмента.

4.5.1. Исходные допущения

Модель угроз строится для гибридной системы ИИ, в которой отсутствует единый доверенный центр, а вклад участников может быть ошибочным, недоверенным или злонамеренным. Предполагается, что нарушитель может контролировать отдельные узлы, внедрять недостоверные данные, воздействовать на контекст агента на основе большой языковой модели, инициировать ложное делегирование, подделывать сообщения при отсутствии криптографической защиты и координировать несколько агентов.

При этом модель не предполагает всемогущего нарушителя: криптографические примитивы считаются стойкими, подписи и хэш-связи не подделываются без компрометации ключей, а трейлеры безопасности позволяют обнаруживать нарушение последовательности операций при корректном хранении и проверке.

4.5.2. Классы угроз

Таблица 4.3. Классы угроз доверенности гибридной ИИ-системы

Класс угроз	Содержание угрозы	Возможные меры противодействия
Компрометация агента	Агент отклоняется от заданной политики функционирования либо начинает действовать в	Снижение уровня доверенности, изоляция агента, проверка трейлеров действий.

	интересах нарушителя.	
Внедрение управляющего воздействия через контекст	Внешнее сообщение изменяет цель, ограничения или порядок действий агента, использующего большую языковую модель.	Контроль входного контекста, трейлер запроса, проверка соответствия политике.
Некорректное использование инструментов	Агент обращается к инструменту вне допустимого сценария или с превышением предоставленных полномочий.	Контроль вызовов инструментов, разграничение уровней автономности, проверка допустимости действия.
Искажение памяти агента	В память агента вносятся ложные или неподтверждённые сведения, влияющие на последующие решения.	Версионирование памяти, механизм цифрового феромона, проверка источников сведений.
Искажение данных или модели	Данные обучения либо обновления параметров модели формируются ошибочно или намеренно искажаются.	Трейлер обновления, оценка доверенности вклада, аудит федеративного обучения.
Атака Сивиллы	В системе создаётся множество ложных агентов, предназначенных для искажения согласования или	Идентификация агентов, ограничение влияния одного источника, анализ структуры связей.

	усиления влияния нарушителя.	
Сговор агентов	Группа агентов согласованно действует против целей доверенного сегмента.	Анализ связей и согласованности поведения, ограничение доминирования, независимое наблюдение.
Роевая атака	Большое число агентов координированно создаёт перегрузку, имитирует легитимную активность или обходит механизмы контроля.	Ограничение интенсивности запросов, пороги доверенности, выявление аномальной плотности событий.
Каскадная деградация	Ошибка, ложное сообщение или вредоносное воздействие распространяется по сети агентов и нарушает согласованность сегмента.	Локализация распространения, цифровые феромоны, контроль маршрутов передачи воздействия.
Подделка истории выполнения	Нарушитель пытается изменить, скрыть или подменить след выполнения действий агента.	Хэш-связанные трейлеры, электронные подписи, распределённый журнал событий.

4.5.3. Угрозы, специфичные для агентов на основе ИИ

В системах ИИ-агентов доверенность нарушается не только в результате атак на инфраструктуру. Существенными становятся

угрозы, связанные с недетерминированностью модели, галлюцинациями, неверным планированием, ошибочным выбором инструмента и скрытым влиянием контекста. Поэтому в модель доверия должны входить не только бинарные события успеха или отказа, но и признаки качества результата, допустимости действия, устойчивости к внедрению управляющего воздействия через контекст и соответствия политике контекста.

Современные рекомендации OWASP [4] для приложений на основе больших языковых моделей выделяют внедрение управляющего воздействия через контекст, нарушение достоверности обучающих данных, отказ в обслуживании модели, уязвимости цепочки поставки и чрезмерная автономность как значимые классы рисков. Эти угрозы особенно важны для гибридных систем, поскольку агент не только генерирует текст, но и способен выполнять действия во внешней среде.

4.5.4. Коллективные угрозы

Безопасность мультиагентных систем рассматривает угрозы, возникающие или усиливающиеся именно через взаимодействие агентов: скрытая коллюзия, стеганографическая коммуникация, роевые атаки, разнородные атаки, скрытное состязательное поведение, атаки на надзор и каскадное распространение ошибок [62]. Следовательно, доверенный сегмент должен анализировать не только локальную историю каждого агента, но и структуру взаимодействий между агентами.

В данной работе коллективные угрозы описываются через граф доверия и показатели структурной устойчивости. Если небольшая группа агентов начинает концентрировать ресурсы,

[4] OWASP Foundation. OWASP Top 10 for Large Language Model Applications. — Version 2025. — URL: https://owasp.org/www-project-top-10-for-large-language-model-applications/ (дата обращения: 10.05.2026); OWASP GenAI Security Project. Agentic AI — Threats and Mitigations. — 2025. — URL: https://genai.owasp.org/resource/agentic-ai-threats-and-mitigations/ (дата обращения: 11.05.2026); OWASP GenAI Security Project. Multi-Agentic System Threat Modeling Guide v1.0. — 2025. — URL: https://genai.owasp.org/resource/multi-agentic-system-threat-modeling-guide-v1-0/ (дата обращения: 13.05.2026).

делегирование и информационные потоки, возникает риск доминирования. Если ложные данные быстро распространяются по сегменту, возникает риск каскадной деградации. Если агенты с независимыми внешними идентификаторами демонстрируют согласованные нарушения, возникает риск коллюзии.

4.5.5. Требования к модели защиты

1. контекстная оценка доверия должна зависеть от конкретной задачи, инструментов и уровня автономности;
2. каждое значимое действие агента должно сопровождаться трейлером безопасности;
3. история поведения должна агрегироваться в цифровой феромон, пригодный для анализа;
4. решение о делегировании должно учитывать не только доверие, но и риск действия;
5. доверенный сегмент должен поддерживать исключение, ограничение и восстановление агентов;
6. модель должна быть устойчива к шуму, единичным ошибкам и попыткам накопления ложной репутации.

В контексте доверенного сегмента каждая из перечисленных угроз имеет технологическое выражение. Внедрение управляющего воздействия через контекст означает, что внешний фрагмент данных начинает выполнять роль скрытой операции, меняющей цель агента. Некорректное использование инструментов означает включение в цепочку действия, не предусмотренного политикой. Преднамеренное искажение памяти означает подмену основания последующих решений. Коллюзия и роевые атаки означают, что нарушение доверенности становится не индивидуальным, а структурным свойством взаимодействия группы агентов.

Следовательно, модель защиты не может ограничиваться фильтрацией входов или статической авторизацией. Она должна фиксировать цепочку действий, проверять происхождение данных, агрегировать поведенческие следы, ограничивать автономность при снижении доверия и поддерживать процедуры восстановления сегмента после локальной деградации.

4.6. АНАЛИЗ СУЩЕСТВУЮЩИХ ПОДХОДОВ И МОДЕЛЕЙ ДОВЕРИЯ

Анализ существующих подходов необходим для определения границ применимости уже известных архитектур и моделей доверия. Рассматриваемые ниже распределённые архитектуры полезны как инженерные прототипы открытых взаимодействий, однако сами по себе они не образуют доверенный сегмент, поскольку не обеспечивают полноту проверки технологической цепочки и динамическую оценку поведения участников.

4.6.1. Ретроспективный обзор архитектур распределённых систем

Развитие открытых вычислительных систем отражает переход от корпоративных архитектур к гибридным децентрализованным моделям и далее — к распределённому интеллекту, функционирующему на периферии сети в промышленных платформах (Siemens MindSphere, Bosch IoT Suite, NVIDIA Isaac, ROS 2, MEC — Multi-access Edge Computing). Эта трансформация стала возможной благодаря одновременному прогрессу в области сетевых протоколов, парадигм искусственного интеллекта и вычислительных инфраструктур.

Ранние распределённые системы и становление P2P

Первые распределённые системы 1970–1990-х годов были ориентированы на прозрачный обмен ресурсами и абстракцию распределения, но опирались на централизованные или частично централизованные сервисы [71]. Рост требований к масштабируемости в конце 1990-х привёл к появлению сетей peer-to-peer (P2P), в которых узлы действовали как равноправные участники без единой точки отказа.

Одним из пионеров применения такого подхода стал Napster – сервис пиратского распространения музыки, использовавший централизованный каталог, но децентрализованное хранение [72]. Его развитие продолжили полностью децентрализованные сети Gnutella и Freenet [72], продемонстрировавшие устойчивость к отказам и динамическому составу узлов.

Следующим этапом стало формирование распределённых хеш-таблиц (DHT), обеспечивающих логарифмический поиск и

устойчивость к динамике состава узлов. Протоколы Chord [73] и Kademlia [74] задали стандарты адресации и маршрутизации в больших децентрализованных сетях. Например, Kademlia-подобные реализации используются в пиринговых сетях [74].

Web3 и блокчейн

Развитие P2P-технологий привело к появлению блокчейна, который можно рассматривать как устойчивое хранилище глобального состояния, основанное на принципах открытой и децентрализованной координации [75]. Блокчейн развил идеи DHT, добавив механизмы консенсуса и экономические стимулы, и тем самым стал фундаментом экосистемы Web3.

Переход к мультиагентным системам: от Distributed AI к BDI-агентам

Рост децентрализации совпал с развитием Distributed Artificial Intelligence (DAI) — направления, анализирующего коллективное поведение автономных вычислительных сущностей. Ещё в конце 1980-х годов учёные заложили основы исследований координации, распределённого поиска и коллективного решения задач [76].

С развитием когнитивных архитектур заметное значение приобрела модель убеждений, желаний и намерений (Belief–Desire–Intention, BDI) [77]. BDI-агенты умеют принимать решения в условиях неопределённости и неполной информации, что делает их естественными участниками децентрализованных систем. Ключевыми задачами в современных МАС являются задачи обеспечения консенсуса и координации между агентами. В [78] предложена математическая основа распределённого согласования и коллективного управления, применимую как к робототехнике, так и в сетевых вычислительных системах.

Fog/Edge computing

Одновременно с развитием мультиагентных систем сформировались два новых архитектурных подхода к организации вычислений:

1. edge computing — перенос вычислений ближе к источнику данных;

2. fog computing — многоуровневая распределённая архитектура, включающая промежуточные узлы между облаком и периферией.

Фундаментальной работой первого направления считается статья [79], определившая edge как способ снижения задержек и нагрузки на облако. Fog computing был представлен в [80] как архитектура для IoT, обеспечивающая локальную обработку, отказоустойчивость и быстрый отклик.

Появление fog/edge-инфраструктур создало естественную среду для мультиагентных подходов: автономных устройств, взаимодействующих в реальном времени, адаптивного управления ресурсами, локального консенсуса и самоорганизации.

4.6.2. Требования к распределённой системе для построения доверенного сегмента

Чтобы понять, подходят ли существующие архитектуры распределённых систем для построения доверенного сегмента, необходимо проанализировать их с точки зрения решаемых ими задач и заложенных архитектурных допущений. Поскольку в настоящее время отсутствует подход к организации доверия в открытых системах, рассмотрение одной отдельной архитектуры не позволяет получить обобщённые выводы. В этой связи целесообразно провести сравнительный анализ различных архитектурных решений, ориентированных на работу в распределённых и децентрализованных системах. Это позволит оценить, какие архитектурные решения действительно способны помочь в решении проблем транспортного уровня и жизненного цикла доверенного сегмента, а также выявить их ограничения.

Здесь формирование критериев оценки архитектур распределённых систем на предмет их применимости в проектировании доверенных сегментов осуществляется на основе положений, определяющих условия построения доверенного сегмента:

- особенности жизненного цикла доверенного сегмента;
- особенности мультиагентного взаимодействия;
- условия функционирования защищённой информационной технологии при обработке и передаче данных в открытой системе.

В совокупности эти положения позволяют выделить семь критериев, по которым возможно сопоставлять архитектуры распределённых систем с целью определения их пригодности для построения доверенного сегмента.

Перечень критериев пригодности архитектур распределённых систем для построения доверенного сегмента:

1. Критерий доверенности узлов и механизмов установления доверия

При проектировании архитектуры доверенного сегмента предполагается наличие формальных или поведенческих механизмов подтверждения корректности узла. Архитектура должна обеспечивать как минимум:

- проверяемость происхождения узла (идентичность),
- оценимость его поведения в прошлом (историчность),
- предсказуемость действий в рамках заданной технологии.

Требования данного критерия вытекают из задач инициализации доверенного сегмента и невозможности полагаться на централизованного администратора в открытых мультиагентных системах.

2. Критерий отсутствия центра и независимости от централизованной точки доверия

Доверенный сегмент функционирует в условиях отсутствия глобального удостоверяющего центра. Следовательно, пригодная архитектура должна поддерживать:

- децентрализованную идентификацию и аутентификацию,
- устойчивость к компрометации отдельных узлов,

Этот критерий отражает принцип автономности агентов в MAC и исключает архитектуры, полагающиеся на глобальные реестры.

3. Критерий проверяемости и контролируемости выполнения операций

Так как доверенный сегмент обеспечивает выполнение защищённой информационной технологии, архитектура должна позволять:

- фиксировать факт выполнения каждой операции,
- контролировать её корректность,
- обеспечивать трассируемость результатов,
- формировать доказуемые следы выполнения (например, через механизм трейлеров безопасности).

Наличие подтверждаемости необходимо для предотвращения накопления ошибок и информационного коллапса.

4. *Критерий устойчивости структуры взаимодействия*

Для доверенного сегмента критично, чтобы структура взаимодействия сохраняла согласованность и работоспособность при:

- динамическом изменении числа узлов (churn),
- неравномерной нагрузке,
- появлении асимметрии ресурсов,
- стремлении отдельных узлов к доминированию.

Этот критерий связан с известными явлениями в MAC — коллапсом первого рода, коллапсом второго рода и иными видами деградации — и задаёт требования к механизмам самоорганизации.

5. *Критерий исключения и изоляции нарушителей*

Открытая среда предполагает возможность появления скомпрометированных узлов. В этом случае архитектура должна поддерживать:

- изоляцию агентов, нарушающих предписанную технологию,
- исключение узлов с недопустимым поведением,
- восстановление согласованности после локальных нарушений,
- предотвращение деградации доверия.

Отсутствие данного механизма приводит к необратимому разрушению сегмента.

6. *Критерий стабилизации и самовосстановления*

Даже корректная система может столкнуться с перегрузками, временной дезорганизацией или асимметриями. Поэтому архитектура должна обеспечивать:

- перераспределение нагрузки,
- корректировку планов и активности агентов,

• стабилизацию доверия при отклонениях,

• возможность возврата к работоспособному состоянию без внешнего вмешательства.

Этот критерий непосредственно связан с предотвращением коллапса второго рода.

7. *Критерий соответствия защищённой технологии (ЗИТ)*

Доверенный сегмент строится вокруг защищённой информационной технологии, что предполагает:

- неизменность технологической цепочки обработки данных,
- унифицированность реализации операций,
- возможность верификации результатов между узлами,
- согласованность политик и контекстных правил.

Таким образом, архитектура должна быть способна встраивать в себя ЗИТ, а не просто обеспечивать обмен данными. Сведем полученный перечень в сводную таблицу и зададим условные обозначения критериям.

Таблица 4.4. Критерии применимости распределённых технологий для построения доверенного сегмента

Обозначение критерия	Название критерия	Смысл
K1	Доверенность узлов	Наличие механизмов идентификации и накопления оценок поведения узлов.
K2	Отсутствие центра	Децентрализованная модель доверия.
K3	Проверяемость операций	Наличие подтверждаемой технологической цепочки.
K4	Устойчивость структуры	Способность избегать коллапсов и асимметрий.
K5	Исключение нарушителей	Механизм исключения или блокировки узлов.

K6	Стабилизация	Способность восстанавливать согласованность.
K7	Соответствие ЗИТ (защищённой информационной технологии)	Возможность реализации ЗИТ.

Указанные критерии образуют методологическую основу дальнейшего анализа существующих распределённых архитектур и позволяют оценить их применимость для построения доверенного сегмента в открытых системах.

4.6.3. Результаты сравнения архитектур

В таблице представлены результаты сравнительного анализа архитектур. Ниже приведён анализ полученных результатов.

Таблица 4.5. Результаты сравнительного анализа распределённых архитектур

Технология / Критерий	K1	K2	K3	K4	K5	K6	K7
P2P-сети	–	+	–	±	–	–	–
Распределённые хеш-таблицы (DHT)	–	+	–	+	–	–	–
Блокчейн	±	+	+	+	±	–	±
BDI	±	+	–	–	–	–	–
Fog / Edge	–	±	–	±	–	–	–

Обозначения:
+ — соответствует критерию
± — частично соответствует
– — не соответствует

P2P-архитектуры демонстрируют высокую степень децентрализации, поскольку каждый узел выступает равноправным участником. Однако они не обеспечивают ни доверенности узлов,

ни подтверждаемости операций, ни механизмов исключения нарушителей. Устойчивость к динамическому составу участников (churn) достигается за счёт топологии, но структура не контролирует поведение агентов. Таким образом, P2P-сети непригодны для формирования доверенного сегмента.

Распределённые хеш-таблицы обеспечивают устойчивость сети к отказам отдельных узлов и динамическому изменению состава участников. Частичное соответствие критерию доверенности узлов обусловлено наличием механизмов идентификации и экономической ответственности валидаторов. Однако они не содержат механизмов доверия к узлам, подтверждаемости и стабилизации. Эти архитектуры эффективны для поиска и маршрутизации, но не способны реализовать защищённую технологию обработки данных.

Блокчейн обеспечивает подтверждаемость транзакций, устойчивость глобального состояния и децентрализованный механизм консенсуса. Однако доверие формируется только к данным, но не к поведению узлов, а исключение возможно только для валидаторов, но исключения не связаны с контролем выполнения технологических операций. Поведение агентов вне цепочки транзакций не стабилизируется, и встроить комплексную защищённую технологию возможно лишь частично.

BDI обеспечивают отсутствие центра и обладают моделью поведения агентов, но не имеют встроенных механизмов подтверждаемости операций, устойчивости структуры и стабилизации. Они подвержены коллапсам первого и второго рода.

Fog/Edge computing обеспечивают только частичную децентрализацию и устойчивость инфраструктуры, но не включают механизмов доверенности, подтверждаемости или исключения нарушителей. Эти архитектуры ориентированы на распределение вычислительных ресурсов и снижение задержек.

Несмотря на разнообразие современных децентрализованных систем, ни одна из них не содержит встроенной модели доверия между автономными участниками, удовлетворяющей совокупности критериев K1–K7. Блокчейн обеспечивает доверие только к глобальному состоянию данных, но не к действиям узлов, не контролирует выполнение защищённой технологии и не

стабилизирует поведение агентов. DHT обеспечивает лишь транспорт и адресацию, не решая задач верификации, доверенности или согласованности поведения. Мультиагентные системы описывают взаимодействие, но не задают механизмов подтверждаемости операций и устойчивости структуры.

Проведённый сравнительный анализ показывает, что существующие архитектуры могут служить компонентами инфраструктуры, но не формируют того уровня доверенности, который требуется для функционирования доверенного сегмента. Это подтверждает необходимость разработки специализированной модели доверия, интегрирующей поведенческие, технологические и структурные механизмы обеспечения доверенности в открытых мультиагентных системах.

В этой главе рассматриваются модели доверия, пригодные для использования в открытых мультиагентных системах.

4.6.4. Понятие модели доверия

Рассмотренные ранее подходы к архитектурам распределённых систем ориентированы преимущественно на решения транспортных, координационных или консенсусных задач, однако не решают задачу формирования доверия к действиям узлов в условиях открытой среды.

Как было сказано выше, доверие к открытым системам невозможно обеспечить существующими методами. В этой связи предлагается альтернативный подход, при котором объектом доверия становится не узел как технический или программный компонент, а реализуемая им информационная технология. Это приводит к необходимости трактовать доверие к действиям узлов как динамическую и контекстно-зависимую величину, формируемую в процессе функционирования системы. Оценка доверия должна формироваться на наблюдаемом поведении субъекта, истории взаимодействия с ним, локальных политик принимающего решения узла и данных независимых наблюдателей.

Для того чтобы определить, насколько существующие подходы к организации доверия в открытых системах решают вышеизложенные проблемы, необходимо сформировать набор

критериев, отражающих ограничения и особенности функционирования доверенного сегмента. Сформулированные критерии далее используются в качестве основания для обзора и сопоставления моделей вычисления доверия в распределённых системах.

4.6.4.1. Требования к моделям доверия в открытых распределённых средах

Проведённый выше анализ позволяет выделить три фундаментальные группы проблем, с которыми неизбежно сталкивается любая мультиагентная система, претендующая на формирование доверенного сегмента.

Первая группа связана с созданием доверенной среды в неконтролируемом окружении, где отдельные узлы могут функционировать на недоверенной инфраструктуре, взаимодействовать через потенциально небезопасные каналы и не иметь центрального источника верификации.

Вторая группа относится к жизненному циклу доверенного сегмента: его инициализации, устойчивости во времени, способности адаптироваться к изменению состава узлов и изменению состояния среды.

Третья группа проблем обусловлена отсутствием в открытой системе условий, обеспечивающих корректное, эквивалентное и проверяемое выполнение защищённой информационной технологии.

Исходя из этих трёх групп проблем, формируются базовые требования к модели доверия, которая способна обеспечить устойчивое функционирование групп агентов в открытой системе. Модель должна обеспечивать:

1. проверяемую локальную доверенность каждого узла и целостность межузловых взаимодействий;
2. поддержку полного жизненного цикла сегмента — от безопасного включения и выхода узлов до механизмов восстановления после нарушений;
3. поддержку корректного и проверяемого функционирования защищённой информационной технологии.

Эти базовые требования задают рамку для формирования функциональных требований, которым должна соответствовать модель доверия, чтобы быть практически применимой в открытых системах.

Первая группа проблем, связанная с функционированием системы в неконтролируемом окружении, обусловливает необходимость локальной проверяемости действий узлов, контекстной интерпретации доверия и устойчивости к недостоверным данным. В таких условиях доверие должно формироваться на основе наблюдаемого поведения и характеристик конкретного взаимодействия.

Вторая группа проблем, относящаяся к жизненному циклу доверенного сегмента и динамике состава участников, порождает требования к динамичности формирования доверия и способности модели адаптироваться к изменениям состояния системы. Модель доверия должна обеспечивать обновление оценок во времени, корректно реагируя на включение и исключение узлов, а также на изменения их поведения.

Третья группа проблем связана с отсутствием условий для корректного функционирования защищённой информационной технологии в открытой системе. Это обусловливает необходимость поддержания эквивалентности технологической цепочки, выявления отклонений в выполнении операций доверенного сегмента.

Выделенные группы проблем не являются независимыми и проявляются в системе одновременно. Как следствие, функциональные требования к модели доверия формируются как результат пересечения указанных групп проблем. В частности, динамичность формирования доверия одновременно обусловлена необходимостью поддержки жизненного цикла сегмента и функционирования в неконтролируемой среде; контекстная зависимость и устойчивость к недостоверным данным вытекают из сочетания открытости среды и отсутствия централизованной верификации; способность формировать устойчивые группы надёжных участников и прогнозировать деградацию системы является следствием требований к поддержке корректного выполнения защищённой информационной технологии в условиях динамики и частичной компрометации узлов.

Выработанные функциональные требования представлены в таблице.

Таблица 4.6. Функциональные требования к моделям доверия

Функциональное требование	Смысл
Динамичность формирования доверия	Модель должна работать в условиях изменяющегося состава участников, формируя и обновляя оценку доверия.
Контекстная зависимость	Оценка доверия должна учитывать тип взаимодействия, текущие цели агентов и характеристики среды.
Устойчивость к недостоверным данным	Модель должна учитывать влияние ложных сообщений, задержек, атакующих участников и шумов в каналах связи, обеспечивая фильтрацию или подавление искажённой информации.
Способность группировать надёжных участников	Необходимо наличие механизма выделения устойчивых подмножеств агентов, внутри которых сохраняется согласованность данных, предсказуемость поведения и ненулевой уровень взаимного доверия.
Поддержка самоорганизации	Формирование доверенных сегментов должно происходить на основе локальных наблюдений и распределённых процедур, без централизованного контроля.
Масштабируемость	Модель должна оставаться работоспособной по мере роста числа агентов и увеличения интенсивности взаимодействий.
Прогнозирование деградации системы	Необходима способность выявлять ранние признаки нарушения доверия,

	приводящие к разрушению коллективного поведения.

Далее будет представлен обзор подходов к вычислению доверия в распределённых системах и производиться их оценка с точки зрения соответствия данным требованиям.

4.6.5. Обзор моделей доверия в распределённых системах

За последние два десятилетия в области распределённых систем и P2P-сетей было предложено множество подходов к вычислению доверия: формальные модели, алгоритмы агрегирования репутаций, социально-контекстные схемы. К ним относятся EigenTrust, PeerTrust, REGRET, FIRE, PATROL, а также широкий класс байесовских и репутационных схем. Все они по-разному отвечают на один и тот же вопрос: как по истории взаимодействий и отзывов предсказать, будет ли данный узел вести себя честно при следующем взаимодействии. Ниже кратко рассмотрим каждую модель и её вклад в развитие методов формализованной оценки доверия в распределённых средах.

EigenTrust

EigenTrust [81] относится к классу алгоритмов распределённого вычисления глобального доверия. Он был разработан для P2P-файлообменных сетей, где угроза поддельных файлов требовала механизма их фильтрации. Алгоритм основывается на построении матрицы локальных оценок, нормализации рейтингов и итеративном распространении доверительных значений до сходимости. Для предотвращения сговора авторы предложили механизм предварительно доверенных узлов, чьи оценки служат опорными точками и препятствуют тому, чтобы группа злоумышленников создавала замкнутые циклы ложного доверия.

PeerTrust

В отличие от EigenTrust, PeerTrust [82] представляет собой многофакторную модель вычисления доверия, ориентированную на транзакционные P2P-сервисы и электронные торговые площадки.

PeerTrust использует пять ключевых параметров:

1. оценки партнёров;
2. объём и частоту транзакций;
3. доверие к источникам отзывов;
4. контекст каждой транзакции (например, её стоимость или тип услуги);
5. контекст сообщества (например, наличие стимулов для оставления отзывов).

Особое значение имеет третий параметр: если источник отзыва сам обладает низкой репутацией, его мнение учитывается в меньшей степени. С технической точки зрения PeerTrust строится поверх распределённой хеш-таблицы (DHT), где данные о репутации хранятся распределённо, а каждый узел использует собственный менеджер доверия для сбора сведений и расчёта итогового рейтинга.

Модель уделяет особое внимание динамике поведения. Она позволяет быстро снижать доверие к узлам, которые длительное время вели себя честно, а затем переходят к злоупотреблениям. В экспериментах показано, что PeerTrust лучше распознаёт такие сценарии, чем модели, в которых все старые оценки учитываются равномерно.

REGRET

Модель REGRET [83], разработанная для мультиагентных систем, рассматривает доверие не только как функцию прошлых взаимодействий, но и как социально обусловленное свойство. В отличие от технических моделей уровня P2P, REGRET вводит представление о социальном графе, ролях, группах и отношениях, влияющих на доверие.

REGRET выделяет три группы источников достоверной информации:

1. личный опыт агента с конкретным партнёром;
2. свидетельские оценки — отзывы других агентов;
3. социальное доверие, зависящее от принадлежности к группам, ролям и структурам сообщества.

Одним из ключевых элементов модели является механизм временного забывания: старые положительные впечатления

постепенно теряют влияние, что позволяет быстро реагировать на резкое ухудшение поведения узла. Это особенно важно в сценариях, где участник долго «строит» репутацию, а затем начинает злоупотреблять накопленным доверием.

FIRE

Модель FIRE [84] стремится объединить преимущества различных подходов в единую архитектуру. Она включает четыре независимых компонента:

1. доверие по прямому опыту,
2. свидетельское доверие,
3. доверие, основанное на роли агента,
4. сертифицированную репутацию, подтверждённую внешними механизмами.

Каждый компонент вносит вклад в итоговую оценку, причём вес различных источников может адаптироваться в зависимости от количества доступной информации. На ранних этапах, когда прямой опыт ещё отсутствует, более значимыми становятся свидетельские оценки и сертифицированные репутации. По мере накопления собственных взаимодействий баланс смещается, и модель начинает всё больше опираться на прямой опыт агента, сохраняя при этом вклад остальных компонент.

FIRE имеет выраженную модульную структуру, благодаря чему её можно адаптировать под конкретные сценарии. Например, в одноранговых P2P-сетях, где узлы обмениваются файлами, часто отсутствуют стабильные роли, и тогда больший вес получают свидетельские оценки — отзывы других узлов о том, как вел себя участник в предыдущих сессиях обмена. В командных MAC — доверие может быть основано на ролях. Модель позволяет изменять набор используемых модулей и их относительные веса, подстраиваясь под контекст применения.

Однако оригинальная модель предполагает достоверность и корректность предоставляемой информации и лишь частично рассматривает механизмы фильтрации ложных свидетельств. Это делает необходимость дополнительных защитных средств — таких как выявление недобросовестных рекомендателей или анализ отклоняющихся оценок.

PATROL

Модель PATROL [85] заявлена как универсальный подход к вычислению доверия в распределённых системах. Авторы учли в модели максимально широкий набор факторов:

1. прямой опыт взаимодействия,
2. рекомендательная информация,
3. надёжность рекомендателей,
4. временную динамику, включая эффект «первого впечатления»,
5. активность и популярность узлов,
6. степень кооперативности,
7. сходство между участниками,
8. их место в иерархии сети.

PATROL оценивает не только целевой узел, но и надёжность источников репутационной информации, что делает модель более устойчивой к ложным или манипулятивным отзывам.

Байесовские и вероятностные схемы доверия

Отдельное направление составляют байесовские и вероятностные методы оценки доверия [86], применяемые как в P2P-системах, так и в мультиагентных средах. В этих подходах доверие рассматривается как вероятностная величина, отражающая шансы честного поведения при следующем взаимодействии. Вероятностные модели позволяют аккуратно учитывать неполную информацию, неопределённость и вариативность наблюдений.

Байесовские методы хорошо подходят для ситуаций, когда данных мало или они неоднородны, однако большинство таких методов слабо учитывают социальный контекст, не предлагают механизмов противодействия сговору и не включают аспекты доверия к рекомендателям.

4.6.6. Анализ моделей доверия с точки зрения функциональных требований

Рассмотренные выше подходы к вычислению доверия демонстрируют широкий спектр подходов к формализации доверия, однако их исходные цели, архитектурные принципы и допущения существенно различаются. Чтобы определить, насколько они

пригодны для построения доверенных сегментов в открытых мультиагентных системах, необходимо сопоставить их с функциональными требованиями, сформулированными ранее.

При сопоставлении с функциональными требованиями становится очевидно, что каждая модель охватывает лишь отдельные аспекты необходимой функциональности, в то время как устойчивый доверенный сегмент требует совокупности свойств. Последующие рассуждения позволят выявить сильные и слабые стороны подходов, а также определить, какие элементы могут быть использованы при формализации собственной модели.

Динамичность формирования доверия

Одним из ключевых признаков открытой мультиагентной среды является изменчивость её состава. В этом аспекте удовлетворительно работают PeerTrust, REGRET, FIRE и PATROL:

- они учитывают временную динамику,
- позволяют снижать доверие при изменении поведения узла,
- реагируют на обновление информации.

Особенно выражена динамичность в PeerTrust, где доверие меняется в зависимости от частоты и интенсивности транзакций, и в REGRET благодаря механизму временного забывания.

В отличие от них, EigenTrust обновляет доверие итеративно, но, по сути, формирует стационарное глобальное значение, которое плохо отражает краткосрочные изменения поведения; байесовские модели динамичны математически, но лишь в предельно локальном смысле и без учёта поведения других участников.

Контекстная зависимость доверия

Взаимодействия в реальных распределённых системах неоднородны, что требует учёта контекста — роли узла, типа операции, стоимости транзакции, структуры сообщества. Здесь особенно выделяются REGRET и PeerTrust.

- REGRET опирается на социальный граф и ролевую структуру, позволяя учитывать принадлежность к группам и типы отношений.
- PeerTrust вводит параметры транзакции и контекста сообщества.

• PATROL также оперирует многочисленными факторами, включая активность, кооперативность, иерархию.

EigenTrust и байесовские модели, напротив, полностью игнорируют контекст, что делает их непригодными для сложных систем, где доверительность зависит от типа действия.

Устойчивость к недостоверным данным

В условиях открытой среды узел не может ожидать достоверности сведений от других участников или каналов связи. В этом отношении лучше всего выглядят подходы PeerTrust и PATROL, оценивающие надёжность рекомендателей перед использованием их отзывов.

• PeerTrust снижает влияние источников с низкой репутацией.

• PATROL дополнительно анализирует динамику поведения рекомендателя.

REGRET частично компенсирует недостоверность за счёт личного опыта, свидетельства, социального доверия, но не имеет встроенных механизмов выявления злонамеренных рекомендателей.

EigenTrust ограниченно устойчив благодаря предварительно доверенным узлам, но, если злоумышленники проникнут в этот набор, общий рейтинг будет искажён.

Байесовские модели устойчивы математически, но не защищены от атакующих, которые формируют систематически ложные свидетельства.

Способность формировать устойчивые группы надёжных участников

Это требование является критическим для доверенных сегментов, поскольку именно сегменты образуют «островки предсказуемости» в открытой сети.

Среди рассмотренных моделей лишь REGRET обладает механизмами, которые можно интерпретировать как сегментацию: принадлежность к группам, социальным ролям и сетевым структурам позволяет выделять подмножества агентов с более высоким уровнем доверия.

В остальных подходах доверие вычисляется исключительно на уровне пар «агент–агент» и не приводит к формированию устойчивых коллективов.

Поддержка самоорганизации

Самоорганизация предполагает способность системы формировать структуры доверия без централизованного управления. Ни один из рассмотренных подходов не реализует это свойство в полном виде.

- REGRET частично поддерживает самоорганизацию через социальную динамику, но её граф формируется не автоматически, а предполагается заранее существующим или медленно изменяющимся.
- PeerTrust полагается на DHT — централизованно-распределённую структуру.
- EigenTrust требует глобальной итеративной процедуры сходимости.
- FIRE и PATROL ориентированы на индивидуальные оценки, а не на системное образование сегментов.

Это означает, что существующие модели доверия плохо подходят для сред, где взаимодействия формируются и исчезают непредсказуемо, а доверенный сегмент должен возникать как результат локальных правил.

Масштабируемость

Значительная часть подходов не предназначена для сетей с миллионами узлов.

- EigenTrust и байесовские схемы масштабируются хорошо и могут применяться в очень больших системах.
- PeerTrust масштабируется удовлетворительно, пока стабильна структура DHT.
- REGRET и PATROL масштабируются плохо из-за сложных зависимостей между агентами и роли социального графа.

Таким образом, модели, которые лучше всего удовлетворяют контекстным или социальным требованиям, оказываются недостаточно эффективными при большом числе участников.

Прогнозирование и предотвращение деградации

Ни один из рассмотренных подходов не содержит встроенных средств для прогнозирования системной деградации. Механизмы вроде временного забывания (REGRET) или снижения доверия при «всплеске» негативного опыта (PeerTrust) реагируют постфактум, но не позволяют выявить условия, при которых начинается разрушение сегмента: рост узловой доминанты, нарушение согласованности, накопление ложных свидетельств или снижение разнообразия связей.

EigenTrust, FIRE и байесовские модели не рассматривают структуру системы как объект анализа и не отслеживают её устойчивость.

Это один из ключевых аргументов в пользу того, что рассмотренные подходы к вычислению доверия принципиально не подходят для построения самоподдерживающихся доверенных сегментов.

4.6.7. Результаты анализа моделей доверия

Результат сравнения представлен в сводной таблице 7 и демонстрирует, что существующие модели доверия покрывают лишь отдельные фрагменты необходимого функционала.

Таблица 4.7. Сравнение подходов к вычислению доверия по функциональным требованиям

Функциональное требование	EigenTrust	PeerTrust	REGRET	FIRE	PATROL	Байесовские модели
Динамичность формирования доверия	–	+	+	+	+	+
Контекстная зависимость	–	+	+	±	+	–
Устойчивость к недостоверным данным	±	+	±	±	+	+
Способность выделять	–	–	+	–	–	–

устойчивые группы участников						
Поддержка самоорганизации	–	–	±	–	–	–
Масштабируемость	+	±	–	±	–	+
Прогнозирование деградации системы	–	–	±	–	–	–

Обозначения:
+ — соответствует;
± — частично соответствует;
– — не соответствует.

Из анализа таблицы следует, что ни одна модель не удовлетворяет всем требованиям, необходимым для устойчивого формирования доверенных сегментов.

Таким образом, существующие подходы могут быть использованы при построении новой архитектуры, но сами по себе они неспособны обеспечить устойчивость доверенного сегмента в открытой среде и формируют основу для разработки новой модели доверия, ориентированной на поведенческие и технические механизмы устойчивости системы.

Современные подходы к доверенному ИИ, архитектуре нулевого доверия и проверяемым удостоверениям важны для гибридных ИИ-систем, однако они решают смежные, а не тождественные задачи. Архитектура нулевого доверия усиливает постоянную проверку субъектов, активов и ресурсов; подходы к доверенному ИИ задают рамки управления рисками; проверяемые удостоверения обеспечивают криптографически подтверждаемое происхождение атрибутов. Модель доверенного сегмента должна соединить эти идеи с проверкой технологической цепочки агентных действий.

4.6.8. Архитектура нулевого доверия, доверенный ИИ и доверенный сегмент

Архитектура нулевого доверия переносит фокус с сетевого периметра на постоянную проверку пользователей, активов и ресурсов [87]. Этот подход методологически близок к работе, однако не решает специфическую задачу проверки технологической цепочки агентных действий. Trustworthy AI, в свою очередь, рассматривает безопасность, надёжность, прозрачность, управляемость и подотчётность ИИ-систем[5], но часто остаётся на уровне принципов и процессов управления рисками.

Предлагаемая модель формируется на пересечении трёх направлений: защиты информации, управления доверием и исследований безопасности систем искусственного интеллекта. Она задаёт инженерный и формальный механизм, позволяющий переводить принцип доверенности в проверяемые события.

4.6.9. Проверяемые удостоверения и подтверждаемое происхождение

Трейлеры безопасности близки к подходам проверяемых удостоверений и подтверждаемого происхождения, поскольку фиксируют происхождение и целостность данных. Спецификация W3C Verifiable Credential Data Integrity описывает механизмы обеспечения подлинности и целостности цифровых документов с использованием криптографических доказательств[6]. В настоящей работе эта идея переносится с уровня документов на агентные действия и технологические цепочки.

Принципиальное отличие состоит в том, что трейлер фиксирует не только атрибут или удостоверение, но и операцию внутри технологии: кто выполнил действие, в каком контексте, с какими входными данными, каким инструментом, с каким результатом и как это действие связано с предыдущими шагами.

---

[5] NIST. Artificial Intelligence Risk Management Framework (AI RMF 1.0). — Gaithersburg: NIST, 2023; NIST. Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile (NIST AI 600-1). — Gaithersburg: NIST, 2024.
[6] W3C. Verifiable Credential Data Integrity 1.0. — W3C Recommendation, 2024.

4.7. ФОРМАЛЬНАЯ МОДЕЛЬ ДИНАМИЧЕСКОЙ ДОВЕРЕННОСТИ

Формальная модель оценки доверия должна одновременно отражать семантические атрибуты доверия, поведенческую динамику агента, контекст действия, проверяемость технологической цепочки и агрегированную память взаимодействий. В отличие от бинарной модели допуска, доверие рассматривается как изменяющаяся во времени величина, связанная с конкретным контекстом и конкретным классом действий.

4.7.1. Семантические атрибуты доверия субъекта

Современные подходы к моделированию доверия в распределённых системах исходят из того, что доверие представляет собой контекстно-зависимую и динамическую категорию, формируемую на основе совокупности взаимосвязанных атрибутов [88]. В отличие от сертификационных схем, где уровень доверия фиксируется административно и не зависит от последующего поведения субъекта, в мультиагентных и иных открытых вычислительных средах доверие возникает как результат накопленного опыта взаимодействия, учитывающего контекст выполняемых операций и репутационные сведения, формируемые распределённым образом.

В [88] выделяется ряд характеристик доверенных отношений, отражающих принципы формирования поведения участников в рамках этих отношений в корпоративных и агентных средах. На базе этих характеристик сформирован перечень атрибутов, который далее используется для формализации понятия «доверие».

4.7.2. Вербальная модель доверия

В открытой среде доверие рассматривается как изменяющаяся во времени характеристика, зависящая от действий агента и качества выполненных операций.

Доверие определяется через совокупность атрибутов, отражающих ключевые свойства доверительных отношений. В этой

нотации каждая оценка доверия включает семь взаимосвязанных компонентов:

1. контекст взаимодействия (ctx);
2. асимметричность (A_S);
3. поведенческо-временная динамика (D);
4. субъективность (S);
5. иерархия связей (H);
6. подтверждаемость (F);
7. коннотация (K).

Эти атрибуты определяют, какие аспекты поведения агента оказывают влияние на уровень доверия, и формируют основу для перехода к вычислимой метрике.

Уровень доверия агента A к агенту B рассматривается как функция времени и представляется через динамическую величину:

$$T_{A,B}(t) \quad (7)$$

Изменение этой величины определяется наблюдаемыми факторами, которые реально отражают качество выполнения агентом своих обязательств в рамках конкретного контекста. К таким факторам относятся:

1. результаты последних взаимодействий агентов,
2. задержки, ошибки и нарушения политики,
3. редкие или аномальные поведенческие паттерны,
4. внешние подтверждения или опровержения,
5. изменения в общей политике контекста, влияющие на интерпретацию событий.

Все перечисленные факторы представляют собой наблюдаемые признаки поведения агента. Поскольку эти признаки изменяются постепенно и могут фиксироваться в любой момент времени, оценка доверия должна трактоваться как величина, изменяющаяся непрерывно и способная реагировать на каждое новое наблюдение.

Таким образом, доверие — это непрерывная величина, отражающая степень обоснованной уверенности в способности агента корректно выполнить делегированную задачу.

Наблюдаемое поведение агента B не имеет универсальной интерпретации: разные субъекты оценивают его по-разному. В модели это описывается атрибутами S и K, задающими

индивидуальные фильтры восприятия поведения. Они определяют, какой вклад в доверие вносит каждая наблюдаемая метрика (успехи, ошибки, задержки, нарушения и др.).

Таким образом, доверие вычисляется локально каждым узлом, что позволяет сохранять многообразие оценок.

Как показано выше, в условиях открытой системы принципиально невозможно полагаться на единый удостоверяющий центр. Поэтому модель опирается на распределённый механизм подтверждения доверия. Для этого предлагается механизм аддитивности доверия, основанный на накоплении истории поведения узла во времени. В рамках данного механизма каждый агент непрерывно вносит вклад в общее состояние доверия сегмента посредством некоторого агрегирующего элемента, отражающего результаты его предыдущих взаимодействий. Корректное поведение приводит к постепенному накоплению доверия, тогда как нарушения или аномалии сопровождаются штрафными воздействиями, уменьшающими вклад узла в коллективную метрику.

При достижении суммарного уровня недоверия, превышающего заданный порог, узел автоматически исключается из доверенного сегмента без необходимости проведения отдельной процедуры голосования. Таким образом, решение об исключении является следствием накопленного консенсуса поведения, а не разового акта подтверждения.

Такой подход обеспечивает децентрализацию доверия и исключает единственную точку отказа.

Каждый акт делегирования или значимое взаимодействие фиксируется через трейлер безопасности (ST), содержащий:

1. идентификатор сообщения,
2. описание операции,
3. идентификатор инициатора,
4. хэш цепочки делегирования, иными словами, подтверждение полномочий,
5. политика контекста,
6. описание контекста операции,
7. наблюдаемые метрики поведения,

8. коннотационный профиль эмитента
9. временную отметку,
10. криптографическую подпись.

Трейлер обеспечивает проверяемость, непротиворечивость и трассируемость доверенных решений, создавая связь между поведением агента, актуальной политикой и формализованной метрикой доверия.

В таком случае формальная модель будет предполагать, что метрика доверия выражается в виде функции, объединяющей эти параметры в динамическое значение, изменяющееся в зависимости от опыта взаимодействий. В простейшем виде это можно представить как:

$$T_{A,B}(t) = f(\text{ctx}, A_S, D, S, H, F, K). \quad (8)$$

где $T_{A,B}(t)$ — уровень доверия агента A к агенту B во времени t ;

ctx — контекст взаимодействия;

A_S — асимметрия отношения;

D — ПВХ;

S — субъективность;

H — иерархия связей;

F — подтверждаемость;

K — коннотация.

Такая метрика позволяет адаптивно регулировать взаимодействие между агентами. Рассмотрим несколько из этих метрик.

В первую очередь контекст фиксирует условия задачи, одинаковые для всех агентов. Контекст не зависит от субъективного восприятия и задаёт общую рамку взаимодействия. Например, если контекстом является «передача конфиденциального файла через защищённый канал», то он одинаков для всех узлов, независимо от их внутренней политики или технических особенностей.

Асимметричность доверия. В распределённых системах отношения доверия изначально несимметричны: агент A может высоко оценивать надёжность B, в то время как агент B может не доверять A в той же степени.

Например, в системе автономной доставки робот-курьер (агент A) регулярно запрашивает у навигационного узла (агент B)

актуальные карты проходимости двора и получает корректные маршруты без ошибок. Для агента A навигационный узел B выглядит предсказуемым и надёжным источником информации, поэтому уровень доверия A→B высок.

Однако в обратном направлении ситуация может быть совершенно иной. Навигационный узел B фиксирует, что робот A периодически отклоняется от рекомендованной траектории, задерживается на контрольных точках или совершает резкие коррекционные повороты, повышающие риск столкновений. С точки зрения узла B это признаки нестабильного поведения, поэтому доверие B→A может быть значительно ниже. В такой системе один участник взаимодействия может оцениваться как надёжный, тогда как обратная оценка остаётся сдержанной — типичная для мультиагентных систем асимметрия доверия.

Эта же ситуация демонстрирует отсутствие транзитивности доверия. Навигационный узел B может полностью доверять облачному координатору (агент C), который агрегирует сенсорные данные со всего района и вычисляет глобальные маршруты. Однако робот A взаимодействует только с узлом B и не имеет собственного опыта работы с координатором C. Из факта, что A доверяет B, а B доверяет C, не следует, что A автоматически доверяет C. Для агента A доверие к координатору остаётся неопределённым до появления подтверждённых наблюдений или прохождения независимых процедур аттестации.

Таким образом, асимметричность доверия является естественным параметром мультиагентных систем, и модель должна учитывать его при построении устойчивых доверенных взаимодействий.

Поведенческо-временная характеристика (ПВХ). Доверие должно изменяться не просто с течением времени, а в зависимости от реального поведения субъекта. Например, если агент в течение длительного периода демонстрировал стабильную работу, но затем в рамках одного контекста совершил несколько последовательных ошибок, уровень доверия к нему должен снижаться значительно быстрее. Это позволяет обнаруживать моменты «переключения» — ситуации, когда субъект начинает злоупотреблять ранее накопленной репутацией или меняет модель поведения.

Субъективность оценки. Разные агенты могут по-разному интерпретировать одно и то же событие. Например, задержка передачи файла в 300 мс для узла в локальной сети является значительным отклонением, а для агента, работающего через спутниковый канал связи, такая задержка считается нормальной. Следовательно, в одном случае доверие будет снижено, а в другом — сохранено или даже увеличено. Это подчёркивает необходимость учёта индивидуального восприятия условий и адаптации доверия к контексту функционирования каждого участника.

Коннотация. Даже при одинаковом контексте агенты могут по-разному оценить последствия одного и того же действия. Например, взаимодействие с удалённым узлом, который использует специфический алгоритм сжатия данных, может восприниматься одними агентами как оптимизация и качественное исполнение задачи, а другими — как риск последствий некорректной обработки или несоответствия требованиям политики безопасности. Таким образом, внешние условия едины, но итоговая оценка события различается. Это свойство фиксирует различия в индивидуальных приоритетах, опыте и внутренних критериях оценки риска.

Универсальность предложенной модели заключается в том, что она формирует доверие на уровне пары субъектов, но не ограничивается только бинарными взаимодействиями. Каждый узел системы формирует собственную оценку доверия, основанную на частных наблюдениях и локальной политике, что делает модель независимой от глобального состояния сети и одинаково применимой в системах любого масштаба. Поскольку доверительная функция определяется локальными характеристиками поведения агента, контекстом задачи и подтверждаемыми свидетельствами, она одинаковым образом подходит как для оценки доверия между двумя узлами, так и для формирования доверенных сегментов из произвольного числа участников.

4.7.3. Основы формализации модели доверия

Для практической реализации семантические атрибуты операционализируются через наблюдаемые показатели поведения. При фиксированных A, B и ctx используется вектор наблюдений

$$\mathrm{obs}_{A,B}^{\mathrm{ctx}}(t) = (s_t, f_t, d_t, p_t, r_t), \quad (9)$$

где s_t — доля успешных взаимодействий, f_t — частота сбоев, d_t — нормированная задержка, p_t — частота нарушений политики, r_t — доля редких или аномальных паттернов поведения.

Вычислимая форма доверия задаётся как

$$T_{A,B}^{\mathrm{ctx}}(t) = f(s_t, f_t, d_t, p_t, r_t, \mathrm{ST}, K_A, P_{\mathrm{ctx}}). \quad (10)$$

Здесь ST обеспечивает подтверждаемость и трассируемость, K_A задаёт индивидуальный фильтр интерпретации поведения, а P_{ctx} определяет веса, пороги и правила реакции в данном контексте.

Доверие обновляется по мере поступления новых наблюдений. Для учёта памяти используется экспоненциальное сглаживание:

$$T_{A,B}^{\mathrm{ctx}}(t+1) = \alpha_{\mathrm{ctx}} T_{A,B}^{\mathrm{ctx}}(t) + (1-\alpha_{\mathrm{ctx}})\tilde{\Delta}_{A,B}^{\mathrm{ctx}}(t), \qquad \alpha_{\mathrm{ctx}} \in (0,1). \quad (11)$$

При фиксированных A, B и ctx эта формула записывается короче:

$$T_{t+1} = \alpha T_t + (1-\alpha)\tilde{\Delta}_t. \quad (12)$$

Если α велико, система медленно меняет оценку и сильнее опирается на историю. Если α мало, оценка быстрее реагирует на последние события. В критичных контекстах α_{ctx} может снижаться при всплеске нарушений или аномалий, но только в пределах, заданных политикой P_{ctx}.

Мгновенная оценка агрегирует текущие метрики качества взаимодействий. Базовый вариант — взвешенная аддитивная модель с монотонными преобразованиями:

$$\Delta_{A,B}^{\mathrm{ctx}}(t) = \omega_s \phi_s(s_t \mid K_A) + \omega_f \phi_f(1-f_t \mid K_A) + \omega_d \phi_d(1-d_t \mid K_A) + \omega_p \phi_p(1-p_t \mid K_A) + \omega_r \phi_r(1-r_t \mid K_A). \quad (13)$$

Весовые коэффициенты задаются политикой контекста:

$$\omega_i \geq 0, \qquad \omega_s + \omega_f + \omega_d + \omega_p + \omega_r = 1. \quad (14)$$

Преобразования ϕ_i нормируют и сглаживают вклад метрик:

$$\phi_i : [0,1] \to [0,1]. \quad (15)$$

Например, может использоваться степенное преобразование

$$\phi_i(x) = x^{\gamma_i}, \qquad \gamma_i \in [0.5, 2], \quad (16)$$

или логистическая функция, если требуется подавлять влияние выбросов.

Метрики нормируются в диапазон $[0,1]$. Для окна наблюдений W_t можно использовать следующие определения:

$$s_t = \frac{N_{succ}(W_t)}{N_{total}(W_t)}, \qquad f_t = \frac{N_{err}(W_t)}{N_{total}(W_t)}. \quad (17)$$

Нормированная задержка задаётся как

$$d_t = \min\left(1, \frac{d_{med}(W_t)}{d_{\max}^{\text{ctx}}}\right), \quad (18)$$

где $d_{med}(W_t)$ — медианная задержка в окне наблюдений, а $d_{\max}^{\text{ctx}}$ — предельно допустимая задержка по политике контекста.

Доля нарушений политики и доля аномалий определяются аналогично:

$$p_t = \frac{N_{viol}(W_t)}{N_{total}(W_t)}, \qquad r_t = \frac{N_{anom}(W_t)}{N_{total}(W_t)}. \quad (19)$$

Чем выше s_t, тем больше вклад в доверие. Чем выше f_t, d_t, p_t и r_t, тем ниже вклад в доверие, поэтому в формуле мгновенной оценки используются величины $1 - f_t$, $1 - d_t$, $1 - p_t$ и $1 - r_t$.

При малом числе наблюдений нельзя делать жёсткий вывод о оценке доверии. Поэтому мгновенная оценка смешивается с априорной оценкой:

$$\tilde{\Delta}_{A,B}^{\text{ctx}}(t) = (1 - \beta_t)\Delta_{A,B}^{\text{ctx}}(t) + \beta_t \mu_0(K_A), \quad (20)$$

где

$$\beta_t = \frac{\lambda_0}{\lambda_0 + n_t}, \qquad \lambda_0 > 0, \qquad \mu_0(K_A) \in [0,1]. \quad (21)$$

Здесь n_t — число наблюдений, λ_0 — сила априора, а $\mu_0(K_A)$ — априорная оценка, зависящая от коннотационного профиля агента A. Осторожный агент может задавать более низкое значение $\mu_0(K_A)$, а более риск-ориентированный агент — более высокое.

Делегирование разрешается, если накопленное доверие не ниже порога, заданного политикой контекста:

$$T_{A,B}^{\text{ctx}}(t) \geq \Theta_{\text{ctx}}. \quad (22)$$

Чтобы избежать «дребезга» на границе порога, вводится гистерезис:

$$\begin{cases} T_{A,B}^{\text{ctx}}(t) \geq \Theta_{\text{ctx}}^{\uparrow} & \Rightarrow \text{allow}, \\ T_{A,B}^{\text{ctx}}(t) \leq \Theta_{\text{ctx}}^{\downarrow} & \Rightarrow \text{revoke}, \\ \Theta_{\text{ctx}}^{\downarrow} < T_{A,B}^{\text{ctx}}(t) < \Theta_{\text{ctx}}^{\uparrow} & \Rightarrow \text{keep current state}, \end{cases} \qquad \Theta_{\text{ctx}}^{\downarrow} < \Theta_{\text{ctx}}^{\uparrow}. \quad (23)$$

Если операция охватывает несколько контекстов $\text{ctx}_1, \ldots, \text{ctx}_m$, агрегированная оценка для решения о делегировании задаётся как

$$T_{A,B}^{\text{agg}}(t) = \sum_{\ell=1}^{m} \nu_\ell \, T_{A,B}^{\text{ctx}_\ell}(t), \quad \nu_\ell \geq 0, \quad \sum_{\ell=1}^{m} \nu_\ell = 1. \quad (24)$$

Критичные контексты получают больший вес ν_ℓ.

При критическом нарушении политики применяется штрафование уже вычисленного значения доверия:

$$T_{A,B}^{\text{ctx}}(t+1) \leftarrow \max\left(0,\, T_{A,B}^{\text{ctx}}(t+1) - \delta_{\text{fine}}(\text{ctx}, K_A)\right). \quad (25)$$

Дополнительно политика P_{ctx} может временно увеличивать веса штрафных компонент в формуле $\Delta_{A,B}^{\text{ctx}}(t)$, прежде всего ω_p и ω_r.

Каждый акт делегирования или значимое взаимодействие фиксируется трейлером безопасности:

$$\text{ST}_k = \langle \text{msg}_{\text{id}}, \text{op}, \text{issuer}_{\text{id}}, \text{cert}_{\text{chain\_digest}}, \text{policy}_{\text{ref}}, \text{ctx}, \text{obs}_t, K_A, \text{ts}, \text{sig}_{\text{issuer}} \rangle. \quad (26)$$

Наблюдаемые метрики, включаемые в трейлер, записываются как

$$\text{obs}_t = (s_t, f_t, d_t, p_t, r_t). \quad (27)$$

Хэш цепочки делегирования может быть задан формулой

$$\text{cert}_{\text{chain\_digest}} = H(\text{Node}_A \parallel \text{Node}_B \parallel \cdots \parallel \text{Node}_m), \quad (28)$$

где $H(\cdot)$ — криптографическая хэш-функция, а $\parallel$ — конкатенация.

Трейлер безопасности обеспечивает:

1. различимость взаимодействий через msg_{id};
2. сопоставление действия с контекстом через op и ctx;
3. проверку полномочий через $\text{cert}_{\text{chain\_digest}}$;
4. проверку применённой политики через $\text{policy}_{\text{ref}}$;
5. фиксацию наблюдаемого поведения через obs_t;
6. фиксацию интерпретационного профиля через K_A;
7. упорядочивание событий через ts;
8. аутентичность и неизменность через $\text{sig}_{\text{issuer}}$.

Для технологической цепочки удобно использовать хэш-связанные трейлеры:

$$h_k = H(ST_k \parallel h_{k-1}), \quad sig_k = \text{Sign}_{sk_{\text{issuer}}}(h_k). \quad (29)$$

Такое представление позволяет обнаруживать удаление, вставку или перестановку операций в цепочке.

Цифровой феромон — агрегированная форма памяти о поведении агента. Он свёртывает множество трейлеров в компактный след, доступный другим агентам при оценке доверия.

Феромон агента B в контексте ctx определяется как свёртка трейлеров за временное окно $[t-\tau, t]$:

$$\text{DF}_B^{\text{ctx}}(t) = g(\{\text{ST}_k(B, \text{ctx}) \mid t - \tau \le t_k \le t\}), \quad (30)$$

где $g(\cdot)$ — функция агрегирования, а t_k — время формирования трейлера ST_k.

Динамика феромона задаётся с испарением старых следов:

$$\mathrm{DF}_B^{\mathrm{ctx}}(t+1) = \gamma_{\mathrm{ctx}}\mathrm{DF}_B^{\mathrm{ctx}}(t) + (1-\gamma_{\mathrm{ctx}})\Psi\big(\mathrm{ST}_{t+1}(B,\mathrm{ctx})\big), \quad (31)$$

где

$$\gamma_{\mathrm{ctx}} = e^{-\lambda_{\mathrm{ctx}}\Delta t}, \qquad \lambda_{\mathrm{ctx}} > 0, \qquad 0 < \gamma_{\mathrm{ctx}} < 1. \quad (32)$$

Оператор $\Psi(\cdot)$ извлекает вклад текущего трейлера в агрегированную память. Чем меньше γ_{ctx}, тем быстрее феромон адаптируется к текущему поведению; чем ближе γ_{ctx} к единице, тем дольше сохраняется память о прошлом.

Если существует константа M_Ψ, такая что вклад каждого трейлера ограничен,

$$\| \Psi\big(ST_k(B,\mathrm{ctx})\big) \| \leq M_\Psi, \quad (33)$$

то феромон остаётся ограниченным:

$$\| DF_B^{\mathrm{ctx}}(t) \| \leq \max(\| DF_B^{\mathrm{ctx}}(0) \|, M_\Psi). \quad (34)$$

Феромоны публикуются в распределённом журнале $\mathcal{L}$ и используются для проверки истории поведения, реконструкции доверительных оценок и выявления начала структурной деградации сегмента.

4.7.4. Формализация модели доверия

Доверенный сегмент рассматривается как подмножество узлов и отношений, устойчивых в заданном контексте. С учётом асимметричности доверия удобно описывать сегмент как ориентированный граф:

$$G^{\mathrm{ctx}}(t) = \big(U, E^{\mathrm{ctx}}(t)\big), \quad (35)$$

где

$$E^{\mathrm{ctx}}(t) = \{(A,B) \in U \times U \mid T_{A,B}^{\mathrm{ctx}}(t) \geq \Theta_{\mathrm{ctx}}\}. \quad (36)$$

Локальный доверенный сегмент с точки зрения агента A определяется как

$$TS_A^{\mathrm{ctx}}(t) = \{B \in U \mid verify(B,t) = 1 \wedge T_{A,B}^{\mathrm{ctx}}(t) \geq \Theta_{\mathrm{ctx}}\}. \quad (37)$$

Здесь $verify(B,t) = 1$ означает, что целостность и аутентичность узла B подтверждены.

Доверенный сегмент должен удовлетворять следующим условиям:

1. участники идентифицированы и обладают подтверждённой целостностью;
2. действия сопровождаются трейлерами безопасности;
3. технологическая цепочка проверяема;

4. доверие находится в допустимых пределах;
5. отсутствует критическое доминирование одного агента;
6. журнал $\mathcal{L}$ обеспечивает аудит;
7. политика P_{ctx} задаёт правила включения, ограничения и исключения.

Политика контекста — нормативно-описательная конструкция, задающая, какие аспекты поведения значимы в данном сценарии и как они влияют на решение о допуске, делегировании или ограничении полномочий.

Формально политика контекста задаётся кортежем

$$P_{\text{ctx}} = \langle \Theta_{\text{ctx}}, \Theta^{\uparrow}_{\text{ctx}}, \Theta^{\downarrow}_{\text{ctx}}, \omega, \phi, \alpha_{\text{ctx}}, \delta_{fine} \rangle. \quad (38)$$

В этом кортеже Θ_{ctx} — базовый порог доверия; $\Theta^{\uparrow}_{\text{ctx}}$ и $\Theta^{\downarrow}_{\text{ctx}}$ — пороги гистерезиса; ω — вектор весов метрик; ϕ — набор преобразований метрик; α_{ctx} — параметр памяти; δ_{fine} — функция штрафования.

Политика выполняет функцию макростабилизатора: она ограничивает локальную адаптацию, предотвращает неконтролируемое смещение порогов, регулирует распределение активности между узлами и препятствует деградации доверенного сегмента.

4.7.5. Цифровой феромон

Каждая технологическая операция агента фиксируется трейлером, при росте числа таких операций трейлеров становится слишком много. В распределённой среде количество ST растёт линейно со временем и стремится к бесконечности. Хранение всех трейлеров становится неэффективным, а вычисление доверия требует их агрегации.

Цифровой феромон (DF) — это агрегированная форма памяти о поведении агента, которая свёртывает множество трейлеров в один след. Аналогичная биологическим феромонам, DF создаёт «облако следов» вокруг агента, по которым можно отслеживать поведение агента во времени, подобно следу, оставляемому муравьями с помощью биологических феромонов.

DF — это сигнал среды о поведении субъекта, воспринимаемый другими агентами.

$$DF_B^{\text{ctx}}(t) = g(\{ST_i(B, \text{ctx})\}_{i=1}^{n}). \quad (39)$$

где $g(\bullet)$ — функция свёртки, агрегирующая данные из всех трейлеров агента B в интервале $[\![t - \tau;\ t]\!]$.

DF должен публиковаться в распределённом журнале доверия и доступен другим агентам для оценки $T(A, B|ctx)$. Феромоны нужны для сохранения памяти, в общем случае можно накапливать трейлеры без ограничения, однако это приводит к росту объёма хранимых данных. Феромон агрегирует такие трейлеры и тем самым решает проблему масштабируемости. При этом старые следы должны экспоненциально затухать по аналогии с биологическими феромонамии.

4.7.6. Связь трейлеров безопасности и цифрового феромона

Каждый акт делегирования или значимое взаимодействие сопровождается трейлером безопасности ST . Поток трейлеров создаёт детализированный, но потенциально очень объёмный след поведения. Для обеспечения масштабируемости вводится цифровой феромон $DF_B(t)$ агента B.

Цифровой феромон — это агрегированная форма памяти о поведении агента, свёртывающая множество трейлеров в компактное представление. Формально феромон в момент времени t определяется как свёртка трейлеров за окно $[t - \Delta, t]$, в которой Δ — шаг дискретной временной шкалы:

$$DF_B(t) = \Phi(\{ST_B(\tau) \mid t - \Delta \le \tau \le t\}). \quad (40)$$

где $\Phi(\cdot)$ — функция агрегирования.

Динамика феромона задаётся с учётом «испарения» по аналогии с биологическими феромонами:

$$DF_B(t_k) = e^{-\lambda \Delta t} DF_B(t_{k-1}) + \Psi\big(ST_B(t_k)\big). \quad (41)$$

где $\lambda > 0$ — коэффициент затухания, $\Psi(\cdot)$ — оператор извлечения вклада текущего трейлера. Старые события оказывают на DF_B всё меньшее влияние, что обеспечивает ограниченность памяти и адаптацию к актуальному поведению.

Феромоны публикуются в распределённом журнале доверия $\mathcal{L}$ и доступны другим агентам для:

- проверки истории поведения;
- реконструкции или верификации доверительных оценок;

• анализа начала структурной деградации (например, роста доминантных узлов или снижения разнообразия связей).

Связка оценки доверия $T_{A,B}^{ctx}$, трейлеров безопасности, цифрового феромона и распределённого журнала охватывает два уровня анализа: микроскопический — поведение отдельного агента, и макроскопический — динамику доверенного сегмента в целом.

4.7.7. Применение к гибридной системе ИИ

Гибридная ИИ-система задаётся кортежем

$$\mathcal{H} = \langle U, C, \mathrm{Act}, \mathrm{Tool}, M, D_{\mathrm{src}}, E, \Pi, ST, DF, \mathcal{L}, \mathcal{T} \rangle. \quad (42)$$

где U — множество агентов; C — множество контекстов; Act — множество действий; Tool — множество инструментов; M — множество моделей; D_{src} — множество источников данных; E — среда; Π — множество политик; ST — множество трейлеров безопасности; DF — множество цифровых феромонов; $\mathcal{L}$ — распределённый журнал доверия; $\mathcal{T}$ — семейство функций доверия.

Агент A описывается как

$$a_A = \langle \mathrm{id}_A, \mathrm{type}_A, \mathrm{owner}_A, \mathrm{Cap}_A, \mathrm{Mem}_A, \mathrm{Tool}_A, \mathrm{Goal}_A, \pi_A, \mathrm{state}_A \rangle. \quad (43)$$

Здесь id_A — идентификатор; type_A — тип агента; owner_A — владелец; Cap_A — множество возможностей; Mem_A — память; Tool_A — доступные инструменты; Goal_A — цели; π_A — локальная политика агента; state_A — текущее состояние.

Действие является минимальной проверяемой единицей модели:

$$\mathrm{act}_k = \langle \mathrm{actor,ctx,input,op,tool/model,output,result,ts} \rangle. \quad (44)$$

Контекст задаётся как

$$\mathrm{ctx} = \langle \mathrm{task,resource,} P_{\mathrm{ctx}}\mathrm{,environment} \rangle. \quad (45)$$

4.7.8. Трейлер безопасности агентного действия

Для агентного действия удобно использовать расширенный трейлер:

$$\mathrm{ST}_k^{\mathrm{act}} = \langle h_{k-1}, \mathrm{id}_{\mathrm{actor}}, \mathrm{type}_{\mathrm{act}}, h_{\mathrm{ctx}}, h_{\mathrm{input}}, h_{\mathrm{output}}, \mathrm{tool}_{\mathrm{id}}, \mathrm{model}_{\mathrm{id}}, \mathrm{policy}_{\mathrm{ref}}, \mathrm{ts}, \mathrm{sig}_{\mathrm{actor}} \rangle. \quad (46)$$

Здесь h_{k-1} — хэш предыдущего трейлера; $\mathrm{id}_{\mathrm{actor}}$ — идентификатор исполнителя; $\mathrm{type}_{\mathrm{act}}$ — тип действия; h_{ctx} — хэш

контекста; h_{input} и h_{output} — хэши входа и выхода; tool_{id} и model_{id} — идентификаторы использованного инструмента и модели; $\text{policy}_{\text{ref}}$ — ссылка на политику; ts — временная метка; $\text{sig}_{\text{actor}}$ — подпись исполнителя.

Для агентов на основе больших языковых моделей трейлер должен дополнительно фиксировать хэш системной инструкции, параметры вызова, используемые источники, вызовы инструментов, факт делегирования, результат проверки и уровень автономности. Это позволяет отличать просто корректный ответ от проверяемого действия.

4.7.9. Функция доверия для гибридной ИИ-системы

В гибридной системе мгновенная оценка может задаваться через вектор признаков:

$$\Delta_{A,B}^{\text{ctx}}(t) = \left(\mathbf{w}_{\text{ctx}}^{T}\mathbf{x}_{A,B}^{\text{ctx}}(t)\right), \quad (47)$$

где $\mathbf{x}_{A,B}^{\text{ctx}}(t)$ — вектор наблюдаемых признаков, $\mathbf{w}_{\text{ctx}}$ — веса контекста.

Агрегированное доверие обновляется тем же правилом, что и в базовой модели:

$$T_{A,B}^{\text{ctx}}(t+1) = \left(\alpha_{\text{ctx}}T_{A,B}^{\text{ctx}}(t) + (1-\alpha_{\text{ctx}})\tilde{\Delta}_{A,B}^{\text{ctx}}(t)\right). \quad (48)$$

Вектор признаков может включать долю успешных действий, ошибки, задержки, нарушения политики, аномальные вызовы инструментов, несоответствие трейлеров, жалобы соседних агентов, признаки коллюзии, качество результата и степень воспроизводимости.

4.7.10. Правило делегирования

Делегирование разрешается только при одновременном выполнении условий по доверию

$$delegate(A, B, \text{ctx}, t) = \text{true} \Leftrightarrow \left(T_{A,B}^{\text{ctx}}(t) \geq \Theta_{delegate}(\text{ctx})\right). \quad (49)$$

Если хотя бы одно из условий нарушено, действие не делегируется или выполняется с дополнительными ограничениями.

4.7.11. Доверенный сегмент гибридной ИИ-системы

Доверенный сегмент в контексте ctx в момент времени t задаётся кортежем

$$TS^{\text{ctx}}(t) = \langle U_{TS}(t), E_{TS}^{\text{ctx}}(t), \text{ST}_{TS}(t), \text{DF}_{TS}(t), P_{\text{ctx}} \rangle. \quad (50)$$

Здесь $U_{TS}(t)$ — множество участников сегмента; $E_{TS}^{\text{ctx}}(t)$ — множество доверенных направленных отношений; $\text{ST}_{TS}(t)$ — трейлеры безопасности сегмента; $\text{DF}_{TS}(t)$ — феромоны участников; P_{ctx} — политика контекста.

Множество доверенных отношений определяется как

$$E_{TS}^{\text{ctx}}(t) = \{(A,B) \in U_{TS}(t) \times U_{TS}(t) \mid T_{A,B}^{\text{ctx}}(t) \geq \Theta_{\text{ctx}}\}. \quad (51)$$

Такая запись сохраняет асимметрию доверия: наличие ребра (A,B) не означает наличия ребра (B,A).

4.7.12. Свойства модели

Свойство 1. Ограниченность доверия. Если $T_{A,B}^{\text{ctx}}(0) \in [0,1]$ и $\tilde{\Delta}_{A,B}^{\text{ctx}}(t) \in [0,1]$, то при $\alpha_{\text{ctx}} \in (0,1)$ выполняется

$$T_{A,B}^{\text{ctx}}(t) \in [0,1] \quad \text{для всех } t. \quad (52)$$

Это следует из того, что обновление является выпуклой комбинацией предыдущего доверия и новой оценки.

Свойство 2. Обнаружение систематической деградации. Если в течение n шагов мгновенная оценка ограничена сверху значением $\bar{\Delta} < \Theta_{\text{ctx}}$, то

$$T_{t+n} \leq \alpha^n T_t + (1-\alpha^n)\bar{\Delta}. \quad (53)$$

Следовательно, при достаточно длинной серии нарушений доверие опустится ниже порога делегирования.

Свойство 3. Устойчивость к единичным ошибкам. Вклад одного события ограничен множителем $1-\alpha$:

$$T_{t+1} - T_t = (1-\alpha)\big(\tilde{\Delta}_t - T_t\big). \quad (54)$$

Поэтому при α, близком к единице, единичное негативное событие не приводит к немедленному исключению агента, но повторяющиеся нарушения накапливаются.

Свойство 4. Трассируемость технологической цепочки. Если каждый трейлер содержит хэш предыдущего трейлера и подпись исполнителя,

$$h_k = H(ST_k \parallel h_{k-1}), \quad sig_k = \text{Sign}_{sk_{\text{actor}}}(h_k), \quad (55)$$

то удаление, вставка или перестановка операций обнаруживаются при проверке цепочки, за исключением случаев компрометации ключей или нарушения предположений криптографической стойкости.

Свойство 5. Контекстность. Для разных контекстов оценки одного и того же агента могут различаться:

$$T_{A,B}^{\text{ctx}_1}(t) \neq T_{A,B}^{\text{ctx}_2}(t). \quad (56)$$

Поэтому доверие не переносится автоматически между контекстами без правила агрегации или отдельной политики.

4.8. АРХИТЕКТУРА И МЕХАНИЗМЫ ФУНКЦИОНИРОВАНИЯ

Архитектура доверенного сегмента гибридной ИИ-системы должна обеспечивать непрерывную связь между формальной моделью доверия и инженерными механизмами её реализации. В рамках работы гибридная ИИ-система задаётся как:

$$\mathcal{H} = \langle U, C, \text{Act}, \text{Tool}, M, D_{\text{src}}, E, \Pi, ST, DF, \mathcal{L}, \mathcal{T} \rangle. \quad (57)$$

Здесь U — множество агентов, C — множество контекстов, Act — множество действий, Tool — множество инструментов, M — множество моделей, D — множество источников данных, E — множество связей, Π — множество политик, ST — множество трейлеров безопасности, P — множество цифровых феромонов, L — распределённый журнал доверия, T — множество функций доверия.

Доверенный сегмент в контексте c и в момент времени t определяется как:

$$TS^{\text{ctx}}(t) = \langle U_{TS}(t), E_{TS}^{\text{ctx}}(t), \text{ST}_{TS}(t), \text{DF}_{TS}(t), P_{\text{ctx}} \rangle. \quad (58)$$

В этом выражении $U_{TS}(t)$ — множество участников сегмента, $E_{TS}^{\text{ctx}}(t)$ — множество доверительных и технологических связей между ними, $\text{ST}_{TS}(t)$ — множество трейлеров безопасности, $\text{DF}_{TS}(t)$ — множество цифровых феромонов, а P_{ctx} — политика или набор политик, действующих в контексте c.

Главный архитектурный принцип состоит в том, что доверенность определяется не только принадлежностью агента к сегменту, а проверяемостью его конкретных действий. Поэтому первичным объектом контроля является действие агента:

$\alpha_k = \langle$actor,context,input,operation,tool/model,output,result,time$\rangle$. (59)

Именно действие α_k связывает исполнителя, контекст, входные данные, выполняемую операцию, используемый инструмент или модель, результат и время выполнения. Следовательно, архитектура фиксирует не абстрактное доверие к агенту вообще, а доверие агента i к агенту j в конкретном контексте c:

$$T_{ij}^{c}(t). \quad (60)$$

Такой подход позволяет разделить разные виды доверенности. Идентификация агента не равна его поведенческой надёжности; корректный вызов инструмента не гарантирует корректность всей технологической цепочки; высокий уровень доверия не означает автоматического права на автономное действие. Поэтому решение о допуске, делегировании или ограничении должно приниматься как результат композиции нескольких свидетельств: трейлеров безопасности, цифрового феромона, локальной оценки доверия и политики контекста.

4.8.1. Общая архитектура

Архитектура доверенного сегмента строится как многоуровневая структура.

Нижний уровень образуют инфраструктурные компоненты: вычислительные узлы, модели, инструменты, источники данных, каналы связи и внешняя среда. На этом уровне фиксируется, какие ресурсы доступны агентам и какие ограничения накладываются на их использование.

Следующий уровень образуют агенты:

$$a_i = \langle \mathrm{id}_i, \mathrm{type}_i, \mathrm{owner}_i, \mathrm{Cap}_i, \mathrm{Mem}_i, \mathrm{Tool}_i, \mathrm{Goal}_i, \pi_i, \mathrm{state}_i \rangle. \quad (61)$$

Агент a_i описывается идентификатором, типом, владельцем, набором возможностей, памятью, доступными инструментами, целями, локальной политикой и текущим состоянием. В гибридной ИИ-системе агентом может быть LLM-агент, сервисный агент, edge-агент, агент-наблюдатель, инфраструктурный компонент или человек-оператор.

Над уровнем агентов располагается уровень действий. Действие является минимальной единицей, подлежащей проверке, фиксации и последующей оценке. Это важно потому, что один и тот

же агент может быть доверенным в одном контексте и недоверенным в другом. Например, агент может корректно выполнять операции классификации данных, но не иметь права автономно вызывать внешний инструмент или изменять критический ресурс.

Контекст действия задаётся как:

$$c = \langle \text{task,resource,policy,environment} \rangle. \quad (62)$$

Контекст c определяет задачу, ресурс, применяемую политику и параметры среды. За счёт этого модель доверия становится контекстно-зависимой: оценка $T_{ij}^{c}(t)$ не переносится автоматически на другие задачи и другие режимы работы.

Выше располагается уровень трейлеров безопасности. Каждое значимое действие или акт делегирования сопровождается трейлером:

$$\text{ST} = \langle h_{k-1}, \text{id}_{\text{actor}}, \text{type}_{\text{act}}, h_{\text{context}}, h_{\text{input}}, h_{\text{output}}, \text{tool}_{\text{id}}, \text{model}_{\text{id}}, \text{policy}_{\text{id}}, \text{ts,sig} \rangle. \quad (63)$$

Трейлер ST связывает действие с предыдущим состоянием цепочки, исполнителем, контекстом, входом, выходом, используемым инструментом или моделью, политикой, временем и подписью. Благодаря этому технологическая цепочка становится проверяемой: можно установить, кто выполнил действие, в каком контексте, с какими данными, каким инструментом и в рамках какой политики.

Следующий уровень образует распределённый журнал доверия L. В нём фиксируются трейлеры, события политик, решения о делегировании, ограничения, исключения и обновления цифровых феромонов. Журнал не заменяет функцию доверия и не является единым центром принятия решений; он обеспечивает проверяемую память сегмента.

Над журналом располагается уровень цифровых феромонов. Цифровой феромон $\text{DF}_{TS}(t)$ является агрегированной памятью о поведении агента i в контексте c. Он свёртывает поток трейлеров в компактный след поведения и позволяет агентам не анализировать всю историю взаимодействий заново при каждом решении.

Динамика феромона задаётся выражением:

$$DF_i^c(t+1) = \lambda_c DF_i^c(t) + (1-\lambda_c)\Phi(tr_i(t), c), \quad 0 \le \lambda_c < 1. \quad (64)$$

Здесь λ_c — коэффициент затухания, а Φ — функция извлечения вклада текущего трейлера в феромон агента. При большом λ_c система дольше сохраняет память о прошлом поведении, при малом λ_c быстрее реагирует на новые события.

Верхний уровень архитектуры образуют функции доверия, политики контекста, мониторинг деградации и механизмы восстановления. На этом уровне принимаются решения о допуске, делегировании, снижении автономности, эскалации к оператору или исключении агента из доверенного сегмента.

4.8.2. Распределённый журнал доверия

Распределённый журнал доверия L предназначен для хранения проверяемых свидетельств о функционировании доверенного сегмента. В журнал включаются трейлеры безопасности tr_k , обновления цифровых феромонов $DF_i^c(t)$, события политик, решения о делегировании, ограничения автономности, факты эскалации и решения об исключении агентов.

Журнал не обязан быть глобальным блокчейном. В зависимости от сценария он может быть реализован как реплицируемый журнал, DAG событий, локальная сеть журналов, журнал с независимыми наблюдателями или гибридная схема. Существенным является не конкретный способ реализации, а выполнение следующих требований: неизменяемость записей после фиксации, проверяемость источника, возможность локальной проверки цепочки, устойчивость к потере отдельных узлов и ограничение влияния одного участника на интерпретацию истории.

Каждый трейлер должен быть связан с предыдущим состоянием цепочки через h_{k-1}, а также подписан исполнителем или уполномоченным компонентом. Поэтому удаление, вставка или перестановка действий должны обнаруживаться при проверке цепочки. В этом смысле журнал обеспечивает трассируемость технологического процесса: он позволяет восстановить последовательность действий, проверить соответствие контексту и определить, какая политика действовала в момент принятия решения.

При этом журнал не является единственным источником доверия. Оценка доверия остаётся локальной и контекстной:

$$T_{ij}^{c}(t). \quad (65)$$

Разные агенты могут по-разному интерпретировать одну и ту же историю, поскольку используют собственные локальные политики и коннотационные фильтры. Журнал предоставляет проверяемые свидетельства, но не навязывает единую глобальную оценку доверия.

4.8.3. Политики контекста

Политика контекста задаёт правила интерпретации действий, метрик. В расширенном виде политика может быть представлена как:

$$\pi_c = \langle \theta_{\text{delegate}}(c), \theta_{\text{restrict}}(c), \theta_{\text{exclude}}(c), w_c, \rho_c, \lambda_c, \text{Esc}_c \rangle. \quad (66)$$

Здесь $\theta_{\text{delegate}}(c)$ — порог доверия для делегирования, $\theta_{\text{restrict}}(c)$ — порог ограничения, $\theta_{\text{exclude}}(c)$ — порог исключения, w_c — веса признаков поведения, ρ_c — коэффициент памяти функции доверия, λ_c — коэффициент затухания феромона, Esc_c — правила эскалации.

Мгновенная оценка поведения агента j с точки зрения агента i в контексте c определяется как:

$$\Delta_{ij}^{c}(t) = \sigma\left(w_c^T x_{ij}(t) - \beta_c R_j^c(t)\right). \quad (67)$$

Здесь $x_{ij}(t)$ — вектор наблюдаемых признаков, включающий успешность действий, ошибки, задержки, нарушения политики, аномальные вызовы инструментов, несоответствия трейлеров, жалобы соседних агентов, признаки коллюзии, качество результата и степень воспроизводимости.

Агрегированная оценка доверия обновляется по правилу:

$$T_{ij}^{c}(t+1) = \text{clip}_{[0,1]}\left(\rho_c T_{ij}^{c}(t) + (1-\rho_c)\Delta_{ij}^{c}(t)\right). \quad (68)$$

Коэффициент ρ_c определяет инерционность доверия. Если ρ_c велик, система медленнее реагирует на единичные отклонения и больше учитывает накопленную историю. Если ρ_c мал, доверие быстрее адаптируется к текущему поведению. Поэтому в критических контекстах политика может снижать ρ_c, чтобы ускорить реакцию на нарушения, а в стабильных контекстах — повышать его,

чтобы избежать чрезмерной чувствительности к единичным ошибкам.

Решение о делегировании принимается по доверию:

$$delegate(i, j, c) = \text{true} \Leftrightarrow T_{ij}^{c}(t) \geq \theta_{\text{delegate}}(c) \quad (69)$$

Политики контекста выполняют роль макростабилизатора доверенного сегмента. Они регулируют распределение делегирования, ограничивают концентрацию полномочий у отдельных агентов, задают допустимую скорость восстановления после сбоев, определяют реакцию на аномалии и предотвращают переход сегмента в состояние структурной деградации.

4.8.4. Жизненный цикл доверенного сегмента

Жизненный цикл доверенного сегмента включает инициализацию, самоорганизацию, развитие, деградацию, восстановление и исключение нарушителей (см. рис. 2).

На этапе инициализации задаются множество агентов $U_{TS}(0)$, контексты C, политики Π_c, начальные значения доверия $T_{ij}^{c}(0)$, начальные феромоны $DF_i^c(0)$, допустимые инструменты, модели и источники данных. Также устанавливаются требования к идентификации участников, формат трейлера безопасности и правила записи в журнал L.

На этапе самоорганизации агенты устанавливают первичные связи, обмениваются проверяемыми действиями и начинают формировать локальные оценки доверия. Каждое значимое взаимодействие фиксируется трейлером ST, а накопленная история постепенно отражается в цифровом феромоне $DF_i^c(t)$. В этот момент доверие ещё не является устойчивым, поэтому политики должны учитывать режим холодного старта и не допускать преждевременной концентрации делегирования у одного агента.

На этапе развития сегмент переходит к устойчивому функционированию. Агенты накапливают историю взаимодействий, феромоны становятся информативными, а значения $T_{ij}^{c}(t)$ начинают отражать не разовые события, а повторяющиеся поведенческие паттерны. В этом состоянии сегмент способен распределять задачи, делегировать действия и адаптироваться к изменениям среды без единого центра управления.

На этапе деградации фиксируются признаки ухудшения состояния сегмента: снижение доверия, увеличение числа нарушений политики, появление аномальных вызовов инструментов, несоответствие трейлеров, рост доминирования отдельных агентов или снижение разнообразия связей. Деградация может быть локальной, когда проблема связана с одним агентом или контекстом, либо структурной, когда нарушается баланс всего сегмента.

На этапе восстановления политики контекста перераспределяют делегирование, снижают автономность подозрительных агентов, усиливают мониторинг, меняют веса признаков w_c , ускоряют затухание устаревших феромонов или переводят часть действий в режим подтверждения оператором. Восстановление должно происходить не произвольно, а на основе проверяемых свидетельств, зафиксированных в трейлерах и журнале.

Если восстановление невозможно или агент продолжает порождать нарушения, применяется исключение из делегирования или из доверенного сегмента. Такое исключение является не административным актом, а следствием накопленной отрицательной динамики доверия и феромона.

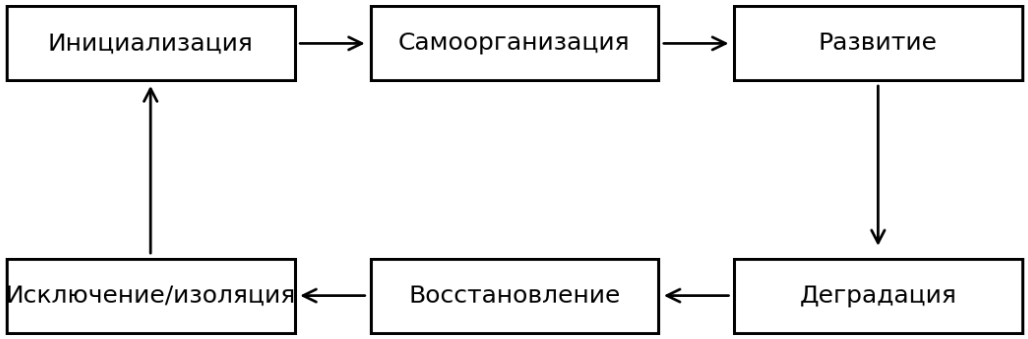


Рис. 4.2. Жизненный цикл доверенного сегмента гибридной ИИ-системы

4.8.5. Механизмы ограничения и исключения

Ограничение агента может быть мягким или жёстким. Мягкое ограничение снижает уровень автономности, запрещает отдельные инструменты, переводит действия в режим предварительного

согласования или усиливает мониторинг. Жёсткое ограничение исключает агента из делегирования и помечает его феромон как деградирующий или вредоносный.

Исключение должно быть проверяемым. Поэтому решение об исключении фиксируется в журнале доверия вместе с набором трейлеров, показавших нарушение, значением функции доверия и политикой, на основании которой принято решение.

4.8.6. Связь с человеческим надзором

В гибридных системах ИИ доверенный сегмент не должен противопоставляться человеческому контролю. Напротив, уровни автономности и правила эскалации должны определять, когда агент действует самостоятельно, когда он предлагает решение, а когда требуется подтверждение оператора. Участие человека в контуре управления (human-in-the-loop) становится не внешним ограничением, а частью политики доверенного сегмента.

Архитектурный вывод состоит в том, что доверенный сегмент не должен иметь единой точки, где концентрируются все решения о доверии. Централизация может быть допустима как вспомогательный механизм аудита или наблюдения, однако сама доверенность сегмента должна формироваться распределённо: через локальные оценки, проверяемые действия, ограничение делегирования и согласованные правила обработки технологических событий.

В практической реализации это означает необходимость проектировать не только интерфейсы агентов, но и формат трейлера безопасности, правила связывания трейлеров с журналом, политику обновления цифрового феромона, условия снижения автономности и критерии исключения агента из сегмента. Эти механизмы должны работать совместно: трейлер фиксирует факт и параметры действия, феромон агрегирует историю поведения, политика определяет допустимость действия, а журнал обеспечивает восстановимость цепочки.

4.9. ЭКСПЕРИМЕНТАЛЬНЫЙ АНАЛИЗ ОЦЕНКИ ДОВЕРИЯ

В этой главе исследуется работа модели доверия в обычных условиях, исключая аномальные паттерны поведения. Целью эксперимента является анализ чувствительности модели доверия к различным вариантам поведения узлов: стабильно хороших, нестабильных, деградирующих и вредоносных.

Эксперименты выполнены на основе реализованной модели, включающей функции вычисления мгновенной оценки поведения Δ(t), итеративного обновления доверия T(t) и параметризуемой политики, регулирующей вклад успешных событий, ошибок, задержек и нарушений политик безопасности.

Мгновенная оценка поведения узла определяется линейной комбинацией наблюдаемых факторов:

$$\Delta_t = \omega_s s_t + \omega_e e_t + \omega_d \phi_d(d_t) + \omega_v v_t. \quad (70)$$

где:

- s_t — успешное действие,
- e_t — ошибка,
- d_t — задержка,
- v_t — нарушение политики.

Агрегированное доверие обновляется рекурсивно:

$$T_{t+1} = \alpha T_t + (1 - \alpha)\Delta_t. \quad (71)$$

где $\alpha \in (0,1)$ определяет «память» системы. В экспериментальной реализации применяется подписанная шкала $T^* \in [-1,1]$, где отрицательная область интерпретируется как недоверие. При необходимости она приводится к нормированной шкале $[0,1]$ преобразованием $T = (T^* + 1)/2$.

Используемая политика оценивания (веса):

- успех: +1.0,
- ошибка: –0.3,
- задержка: –0.15,
- нарушение политики: –0.5.

Порог доверия для делегирования задан как:

$$\Theta_{\text{ctx}} = 0.5. \quad (72)$$

Для моделирования были заданы четыре характерных сценария поведения:

1. Хороший узел — в основном стабильная работа, редкие задержки и единичная ошибка.

2. Нестабильный узел — частые значительные задержки, периодические ошибки и нарушения.

3. Деградирующий узел — плавное ухудшение задержек, накопление ошибок и переход к нарушению политик.

4. Вредоносный узел — ошибки с нарушениями политики, высокий шум в задержках, преднамеренные аномалии.

Для каждого сценария были построены три типа графиков:

• динамика $T(t)$ при разных значениях параметра инерционности α;

• карта чувствительности доверия к ошибкам в плоскости (α, w_error);

• фазовая диаграмма доверия в координатах $\big(\Delta(t), T(t)\big)$.

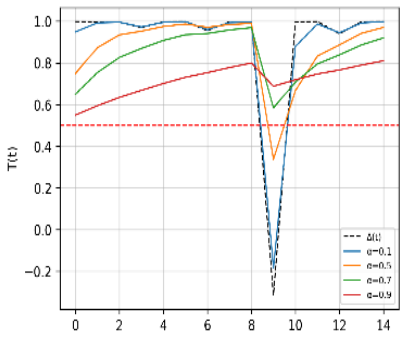
4.9.1. Анализ результатов

Ниже приведено описание результатов для каждого сценария в соответствии с объединённой визуализацией (рис. 4.3).

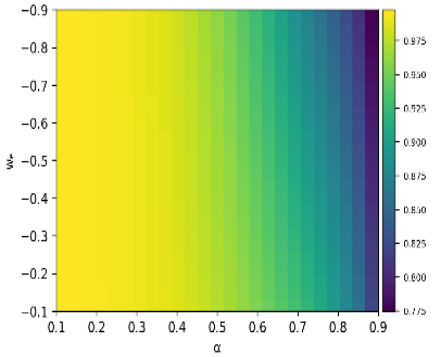
Сравнение динамики доверия для различных типов поведения узлов

Хороший узел

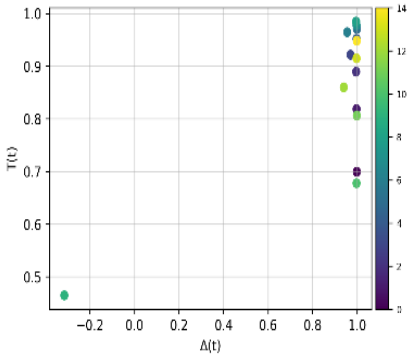
Кривые доверия T(t) при разных α



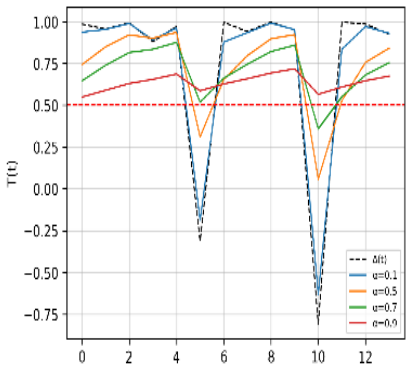
Чувствительность α–w_e

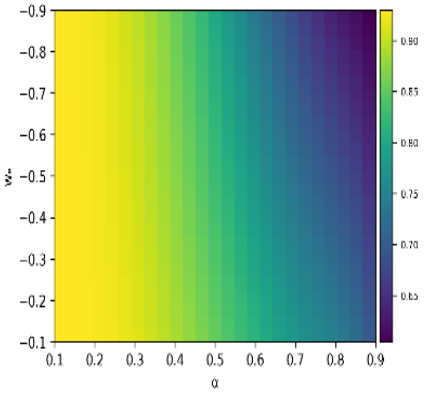


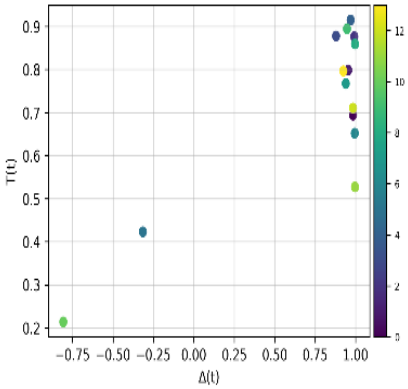
Фазовая диаграмма T–Δ



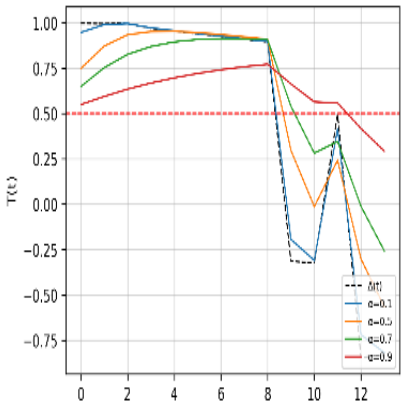
Нестабильный узел

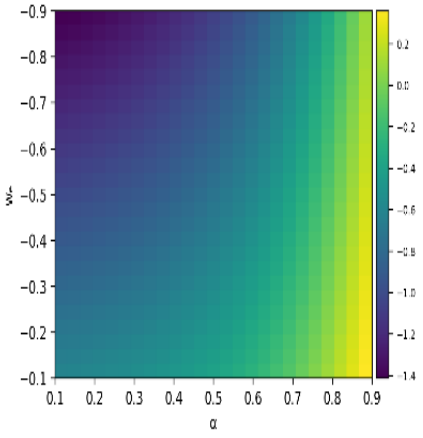


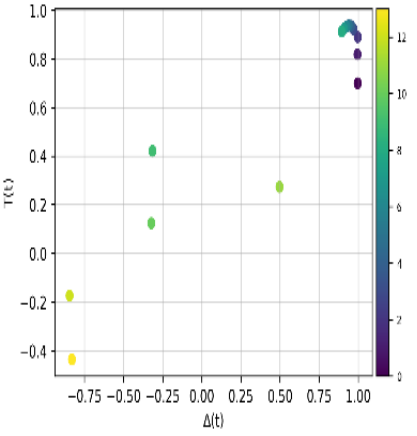




Деградирующий узел







Вредоносный узел

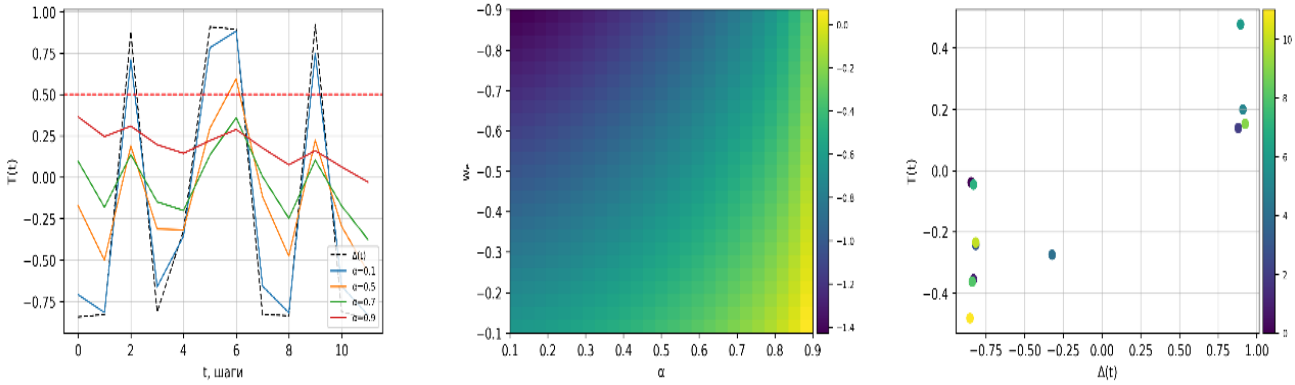


Рис. 4.3. Сравнение динамики оценки доверия для различных типов поведения узлов

Хороший узел

На графике кривые доверия показывают устойчиво высокий уровень доверия, оставаясь выше порога делегирования для всех α. Отмечаются кратковременные провалы, связанные с единичной ошибкой или увеличением задержки, однако:

• при $\alpha = 0.9$ реакция системы на доверие к узлу почти не падает ниже порога $\Theta = 0.5$.

• при $\alpha = 0.1$ модель реагирует мгновенно на редкие выбросы: единичная ошибка вызывает глубокий провал – но доверие к узлу быстро возвращается к высоким значениям.

Карта чувствительности показывает, что даже при сильном штрафе за ошибку $w_{error} \approx -0.9$ итоговое доверие к узлу остаётся положительным.

Это говорит о том, что хороший узел формирует неизменно позитивную оценку доверия, компенсируя редкие ошибки.

На фазовой диаграмме точки расположены преимущественно в правой верхней части плоскости, что визуально подтверждает стабильность поведения.

Нестабильный узел

Для нестабильного узла характерны выраженные колебания доверия — узел периодически «проваливается» ниже порога делегирования:

• при α = 0.1 доверие резко падает до отрицательных значений после больших задержек и ошибок;

• при α = 0.9 колебания сглажены, но доверие всё равно периодически опускается к порогу.

На тепловой карте видно, что при $w_{error} > 0.5$ доверие почти всегда падает ниже порога. Фазовая диаграмма показывает, что облако точек растянуто, что отражает нерегулярность поведения узла.

Деградирующий узел

Кривые доверия демонстрируют устойчивый нисходящий тренд. Независимо от α:

- сначала доверие соответствует хорошему узлу,
- затем уменьшение задержек и накопление ошибок снижают $T(t)$,
- в финале доверие стабильно падает ниже порога.

На тепловой карте видно, что существенные различия между зонами $\alpha < 0.5$ и $\alpha > 0.5$:

• при малых α итоговое доверие становится отрицательным при любом штрафе;

• при больших α систематическая деградация становится очевидной только при $w_{error} \leq -0.3$.

Таким образом, высокая инерционность может временно скрывать деградацию, но не предотвращает её распознавание.

На фазовой диаграмме облако смещается от правой части графика влево вниз, отражая переход к негативному поведению.

Вредоносный узел

Для вредоносного узла доверие быстро уходит в отрицательную область и остаётся там. Даже:

- колебания имеют высокую амплитуду, что связано с чередованием ошибок, нарушений и аномальных задержек;
- даже при $\alpha = 0.9$ доверие остаётся отрицательным после первых нескольких шагов.

Модель быстро выявляет вредоносный узел, независимо от параметров α.

На тепловой карте видно просадку: итоговое доверие остаётся в неположительной зоне при любом сочетании параметров.

На фазовой диаграмме видно, что большинство точек сместилось в левую нижнюю зону: поведение вредоносного узла предсказуемо негативно.

4.9.2. Результаты эксперимента

Проведённый экспериментальный анализ показал, что предложенная модель доверия адекватно реагирует на различные варианты поведения узлов и формирует устойчивую, интерпретируемую оценку их надёжности. На основе четырёх характерных сценариев поведения было продемонстрировано, что модель не только корректно отражает мгновенные изменения наблюдаемых метрик, но и формирует стабильную долгосрочную оценку доверия, чувствительно реагирующую на специфику поведения узла.

Для хорошего узла модель обеспечивает устойчивое поддержание доверия выше порога делегирования независимо от значения параметра инерционности α: редкие ошибки и задержки компенсируются преобладающими положительными событиями. Нестабильный узел характеризуется выраженными колебаниями доверия, что приводит к регулярному пересечению порога и позволяет корректно идентифицировать его как ненадёжный. Для деградирующего узла доверие закономерно снижается по мере ухудшения поведения, что обеспечивает своевременное выявление критических изменений динамики. Вредоносный узел во всех случаях вызывает быстрое и устойчивое падение доверия в отрицательную область, что подтверждает способность модели надёжно и предсказуемо выявлять опасные узлы.

Тепловые карты чувствительности показывают, что параметры политики, в частности штраф за ошибку, оказывают заметное влияние на итоговую оценку доверия преимущественно для промежуточных типов поведения — нестабильных и деградирующих узлов. В то же время для хороших и вредоносных узлов итоговое решение остаётся устойчивым при любых разумных вариациях параметров. Фазовые диаграммы дополнительно подтверждают

различимость классов поведения в пространстве $(\Delta(t), T(t))$, позволяя визуально и формально разделять надёжные, нестабильные и негативно ведущие себя узлы.

Полученные результаты демонстрируют, что предложенная модель доверия обладает высокой интерпретируемостью и предсказательной точностью. Она различает различные типы поведения узлов и может служить основой для механизмов делегирования, самоорганизации и поддержания доверенной среды в открытых распределённых системах.

4.10. ЗАКЛЮЧЕНИЕ

Таким образом, предложенная модель доверия формирует методологическую и архитектурную основу для построения доверенных сегментов в открытых распределённых системах. В отличие от существующих подходов, она сочетает поведенческую оценку, проверяемость технологической цепочки, механизм стабилизации и децентрализованное управление доверием, что делает её практичным инструментом для применения в мультиагентных системах, edge/fog-инфраструктурах, децентрализованных платформах и системах удалённого взаимодействия.

Перспективными направлениями работ в этом направлении являются экспериментальная реализация распределённого журнала доверия, исследование устойчивости сегмента при больших масштабах, разработка алгоритмов групповой самоорганизации и интеграция модели с реальными системами передачи и обработки данных. Эти направления позволят расширить практическую применимость модели и приблизить создание полноценных самоорганизующихся доверенных сегментов, способных функционировать в условиях неопределённости и отсутствия единого центра управления.

Предложенная модель не исчерпывает всю проблематику доверенности гибридных ИИ-систем, однако задаёт основу для дальнейших исследований. В частности, необходимы расширенный симулятор, экспериментальная проверка на сценариях

взаимодействия агентов на основе больших языковых моделей, моделирование коллективных атак и проверка устойчивости распределённых политик. Именно эти направления образуют следующий этап развития модели динамической доверенности.

Глава 5: РАСШИРЕНИЕ ПРОСТРАНСТВА ПРИЗНАКОВ В ИНС ДЛЯ ЗАДАЧ АНТИФРОДА В ДБО

С. Г. Ищанова

Внедрение искусственного интеллекта становится все более распространенным способом повышения качества средств защиты информации. О необходимости такого решения заявили 64% представителей компаний, пострадавших от атак на инфраструктуру еще в 2023 году, такое решение, по мнению опрошенных, повысит эффективность защиты, так как традиционные средства не позволяют обеспечить полноценную защиту при возрастающих затратах на закупку таких средств [89]. Нейронные сети являются одним из самых эффективных способов для решения задач, в которых алгоритм решения не формализован или неизвестен [90], поэтому внедрение нейронных сетей в средства защиты информации может повысить качество этих средств. В 2025 году методы машинного обучения, в том числе нейронные сети являются частью множества средств защиты информации, в том числе модулями средств антифрода для банковской сферы[1], сферы автострахования, сферы, разрабатывающей решения для информационной безопасности [91–92].

[1] Применение алгоритмов машинного обучения к задаче выявления мошенничества при использовании пластиковых карт [Электронный ресурс]. – Режим доступа: https://www.sberbank.ru/ru/person/kibrary/articles/primenenie-algoritmov-mashinnogo-obucheniya-k-zadache-vyyavleniya-moshennichestva-pri-ispolzovanii-plastikovykh-kart (дата обращения: (16.04.2026).

В современном секторе банковских услуг обслуживание физических лиц преимущественно реализуется через дистанционные каналы управления счетами, переводными операциями и депозитами. Крупные кредитные организации предоставляют клиентам удобный и интуитивно понятный формат доступа к финансовым сервисам посредством программных продуктов – мобильных банковских приложений. Высокая степень доступности, эргономичность и оперативность выполнения операций объясняют массовую установку указанных приложений на персональные мобильные устройства пользователей банковских сервисов.

Согласно результатам опроса, проведённого в 2025 году, 85% опрошенных идентифицировали себя как активных пользователей банковских приложений; 62% опрошенных указали на ежедневное использование подобного инструментария; в среднем на одном мобильном устройстве российского гражданина можно найти три таких приложения[2].

В 2025 году основной функциональностью банковских приложений оставались платёжные операции и денежные переводы. Широкое распространение мобильных банковских приложений выступило катализатором роста видов кибермошенничества, ориентированных на совершение несанкционированных транзакций – так называемого фрода. Термин «фрод» (от англ. fraud – обман) в банковском контексте обозначает операции, сопряжённые с противоправными мошенническими действиями.

По итогам 2025 года Банк России сообщил, что объем хищений, совершенных мошенниками в контексте операций без добровольного согласия, составил 29,3 млрд руб.[3]. По сравнению с 2024 годом показатель вырос на 6,4%. Количество операций,

---

[2] Чего не хватает россиянам в банковских приложениях: исследование [Электронный ресурс] // Hi-Tech Mail. 13.11.2025. Режим доступа: https://hi-tech.mail.ru/articles/137344-rossiyane-ozhidayut-ot-finansovyh-servisov-novyh-funkcij-keshbeka-i-stabilnoj-raboty/ (дата обращения: 16.03.2026)

[3] ЦБ: объем похищенных мошенниками средств в 2025 году превысил 29 млрд рублей [Электронный ресурс] // Коммерсантъ. 2026. Режим доступа: https://www.kommersant.ru/doc/8438895 (дата обращения: 11.05.2026).

совершенных без согласия клиента (далее – ОБДС, операции без добровольного согласия), по сравнению с 2024 годом увеличилось на 31,2%, до 1,6 тыс. Банки возместили пострадавшим клиентам ущерб на сумму 1,7 млрд руб., что составляет 5,9% всех операций без добровольного согласия. Рост показателя объема хищений можно объяснить внедрением в мобильные приложения банков специального сервиса для пострадавших от фрода, который позволяет клиентам банка оперативно заявить об ОБДС, а также получить документальное подтверждение произошедшего для дальнейших действий. Этот сервис мог увеличить количество обращений, за счет обращений от пострадавших с небольшим размером ущерба, которые ранее не обращались за помощью, а также позволил получить больше информации о реквизитах мошеннических счетов[4].

На рисунке 5.1 представлен график динамики объема хищений ОБДС с 2020 по 2025 годы по данным Банка России[5]. Отметим, что в период с 2020–2023 характеризуется умеренным, но при этом устойчивым ростом объема хищений, за 3 года этот показатель увеличился примерно на 60%. За 2024 год произошел резкий скачок, объем хищений вырос на 74% по сравнению с 2023, достигнув показателя в 27,53 млрд рублей, что составило самый масштабный прирост за весь представленный период. В 2025 году рост объема значительно сократился и составил всего 6,5%, относительное увеличение составило 1,78 млрд рублей, что близко по значениям к показателям 2022–2023 года.

[4] Обзор операций, совершенных без добровольного согласия клиентов финансовых организаций // Банк России. 2025. [Электронный ресурс]. Режим доступа: https://www.cbr.ru/analytics/ib/operations_survey/2025/ (дата обращения: 11.05.2026).

[5] Информационная безопасность // Банк России. [Электронный ресурс]. Режим доступа: https://www.cbr.ru/analytics/ib/ (дата обращения: 11.05.2026).

Рис. 5.1. График динамики объема хищений ОБДС с 2020 по 2025 г.

Рост показателя в 2024 году может быть объяснен внедрением новых, более полных методик учета случаев фрода и появлением новых мошеннических схем. Пик хищений приходится на 2025 год, но в 2025 году наблюдается минимум прироста с 2020 года, что позволяет предположить эффективность проведения организационных мер по повышению осведомленности банковских клиентов о различных методах социальной инженерии и внедрения средств антифрода. Абсолютный уровень остается крайне высоким (почти в 3 раза выше, чем в начале анализируемого периода), что подтверждает актуальность дальнейшего совершенствования методов обнаружения и предотвращения мошеннических операций.

Несмотря на то, что средства антифрода постоянно совершенствуются, выявляются новые признаки мошеннических схем, проводятся новые эксперименты по повышению качества методов машинного обучения, обеспечивающих работу средств антифрода, по выводам статистики, в 2025 году ущерб от действий мошенников составил 275–295 млрд руб., по оценкам «Сбера»[6]. В

[6] «Сбер» раскрыл, сколько средств мошенники похитили у россиян за год // РБК. 2026. 29 января. [Электронный ресурс]. Режим доступа:

эту статистику вошел ущерб, нанесенный пострадавшим от действий мошенников, реализовавших сложные схемы, которые позволили оказать на владельца аккаунта психологическое давление, в результате которого пользователь сам предоставил мошенникам доступ к управлению банковскими счетами или же сам в банковском отделении снял наличные деньги и отдал курьерам. При этом «Сбер» заявляет, что благодаря активным мерам, включая внедрение средств антифрода, удалось остановить рост объема похищенных средств – в 2024 году было украдено 250–300 млрд. руб. Приведённые статистические данные подтверждают, что, хотя процессы создания и интеграции методов антифрода сопряжены с объективными трудностями, достигнутый на сегодняшний день положительный результат (выражающийся в сдерживании роста объема хищений) делает дальнейшее развитие указанных методов не только целесообразным, но и необходимым. Иными словами, даже при наличии отдельных ошибок и недочётов в функционировании систем антифрода, их применение уже обеспечивает измеримый сдерживающий эффект, что служит достаточным основанием для продолжения научно-технических работ в данном направлении.

При этом по сведениям Управления по организации борьбы с противоправным использованием информационно-коммуникационных технологий МВД в 2025 году объем похищенных средств в результате дистанционных хищений вырос на 5% несмотря на то, что общее число пострадавших снизилось[7]. Следует отметить, что в 2025 году число несовершеннолетних, ставших жертвами фрода, выросло почти вдвое, при том, что число хищений у пенсионеров значительно сократилось, что говорит об эффективности принимаемых мер. В ведомстве отмечают, что теперь характер мошеннических действий стал более «точечным» – по всей

https://www.rbc.ru/finances/29/01/2026/697b2a379a7947671694187e (дата обращения: 11.05.2026).

[7] Средний ущерб от киберпреступлений в России вырос на 5% в 2025 году // РБК [Электронный ресурс]. Режим доступа: https://www.rbc.ru/rbcfreenews/694bf7d69a794756d832ace0 (дата обращения: 11.05.2026).

видимости, действия злоумышленников усложняются и движутся по направлению развития к действиям, подобным «Advanced Persistent Threat» («продвинутым устойчивым угрозам»). APT-атаки — это долгосрочные, тщательно спланированные кибератаки, направленные на конкретную организацию или отрасль. Они характеризуются длительным скрытым присутствием злоумышленников в скомпрометированной информационной системе, многоэтапностью и использованием сложных методов для достижения стратегических целей. Например, злоумышленники узнают личную информацию о потенциальной жертве через украденные базы данных, в том числе информацию о месте проживания, родственниках, семейном положении и даже о примерном распорядке дня. Эту информацию в дальнейшем используют, чтобы повысить уровень уязвимости человека перед мошеннической схемой. Единичные случаи более адресного и сложного подхода среди банковских мошенников уже были известны, но изменение подхода мошенников от массовых телефонных звонков к персональным стратегиям обмана жертвы создает перспективы, в которых существующие средства антифрода будут полезны только для детектирования базовых случаев фрода, например, получение СМС-кода от банка мошенниками.

Помимо представленных выше сведений, возникает проблема ошибочных блокировок при проведении самим пользователем операций, похожих на банковские переводы, совершаемые мошенниками, например, многократное пополнение счетов на одном из маркетплейсов[8]. Эксперты оценивают количество блокировок, совершенных за первые три недели 2026 года, в 2–3 миллиона. По сравнению с более ранним периодом, в котором количество блокировок в среднем составляло 330 000 в месяц, количество срабатываний заметно выросло, при этом многие банки не объясняют клиентам причину блокировок, что осложняет снятие

[8] Банки заблокировали до 3 млн карт и счетов физлиц в первые недели 2026 года // Ведомости. 2026. 19 января. [Электронный ресурс]. Режим доступа: https://www.vedomosti.ru/finance/news/2026/01/19/1170089-banki-zablokirovali-do-3-mln (дата обращения: 11.05.2026).

ограничений для клиентов банка, в результате чего у клиентов возникают жалобы и недовольство сервисом.

Подобные ситуации также возникали в 2025 году. По данным Росфинмониторинга, в 2025 году от физических и юридических лиц поступило 840 жалоб, связанных с блокированием банковских счетов; при этом количество заявлений о действиях злоумышленников сократилось[9]. Причиной блокировок в большинстве случаев служит нетипичное поведение пользователя в банковском приложении (например, высокая частота переводов)[10], однако подобные признаки не всегда позволяют достоверно квалифицировать операцию как мошенническую. Наличие ложных блокирований во фрод-мониторинге, инициируемых на основе *простейших* аномалий поведенческих паттернов (в частности, частоты транзакций), способно наносить значительный ущерб как добросовестному клиенту, так и самой кредитной организации.

Последствия выражаются, прежде всего, в падении доверия клиентов к банковским сервисам, увеличении операционной нагрузки на контакт-центры при необходимости верификации законных операций и, как следствие, в снижении общей результативности защитного механизма вследствие эффекта «привыкания» к регулярным ложным срабатываниям. Для клиентов банка возможны репутационные и финансовые потери, обусловленные невозможностью совершить или получить платёж, принять перевод, что может представлять собой «упущенную выгоду». Частые ложные срабатывания обесценивают предупреждения в интерфейсе банковского приложения, формируя у пользователей устойчивую привычку игнорировать такие уведомления либо воспринимать сервис как некомпетентный.

---

[9] *Курносёнкова Д.* Заблокировали счет в банке: причины и что делать для разблокировки // РБК. 01.01.2026. [Электронный ресурс]. Режим доступа: https://www.rbc.ru/quote/news/article/691d7cb99a79473e8bce967a (дата обращения: 16.03.2026)

[10] *Добрунов М., Кошкина Ю.* «…Одновременно участились сообщения о необоснованных блокировках счетов…» // РБК. 18.11.2025. [Электронный ресурс]. Режим доступа: https://www.rbc.ru/finances/18/11/2025/691c27d39a7947259dfa1e11 (дата обращения: 16.03.2026)

Таким образом, средства антифрода в 2026 году настроены детектировать признаки мошеннических схем, такие как хаотичные или учащенные переводы. Мошеннические схемы развиваются в сторону персонализации, кроме того, мошенники постоянно адаптируют схемы под новые правила, чтобы обойти мониторинг, например, просят обманутого пользователя снять наличные средства вместо переводов. В итоге специалистам приходится дополнительно расширять список признаков мошеннических схем, вследствие чего возникает проблема ошибочных блокировок. Предлагается изменить подход работы средств антифрода: вместо методов классического машинного обучения применить нейронные сети, в том числе нейронные сети глубокого обучения. Предлагается обучать НС не на изменяющихся мошеннических схемах, а на эталонном поведении каждого пользователя банковского приложения, при этом по возможности учитывая психологические характеристики пользователя. Предполагается, *что такой подход более соответствует характеру поведенческих данных, характеризующих взаимодействие пользователей с банковским приложением*.

5.1. УЧЕТ ПСИХОЛОГИЧЕСКОГО ФАКТОРА ПРИ РЕШЕНИИ ЗАДАЧИ АНТИФРОДА

В 2024 году Центральный банк опубликовал портрет пострадавшего от кибермошенничества[11], в 2023 году ученые СПБГУ определили психотипы, обладатели которых склонны доверять мошенникам[12]. Проведение данных исследований говорит о

[11] Кибермошенничество: портрет пострадавшего // Центральный банк Российской Федерации. [Электронный ресурс]. Режим доступа: https://cbr.ru/statistics/information_security/cyber_portrait/2024/ (дата обращения: 11.05.2026).

[12] Ученые СПбГУ определили психотипы доверчивых жертв мошенничества // Санкт-Петербургский государственный университет. [Электронный ресурс]. Режим доступа: https://spbu.ru/news-events/novosti/uchenye-spbgu-opredelili-psikhotipy-doverchivykh-zhertv-moshennichestva (дата обращения: 11.05.2026).

научном интересе к выявлению психологических характеристик, по которым можно оценивать уровень уязвимости пользователя к мошенническим схемам. Но на апрель 2026 года в открытых источниках отсутствует информация о прикладном применении анализа информации о психологических чертах пользователей в системах антифрода для банков.

В научных работах прошлых лет можно встретить различные дифференциации пользователей компьютерных систем по склонности к реализации угроз внутреннего нарушителя. В статье А. П. Федюниной «Выявление характерологических признаков и составление психологического портрета возможного нарушителя и лояльного сотрудника в сфере информационной безопасности» представлено описание четырех признаков, свойственных потенциальным нарушителям: лидерские качества, черты диссидентства, девиантные черты, обладание манипулятивными навыками [93]. Данная научная статья освещает поведенческую сторону вопроса, рассматривает психологические черты потенциального нарушителя.

В докторской диссертации «Модели и методы адаптивного риск-ориентированного управления доступом в распределенных информационных системах» [94] предложен способ управления доступом, базирующийся на анализе психологических реакций пользователя при взаимодействии с интерфейсными элементами с помощью измерения времени ответа на контрольные вопросы, данный метод был внедрен на предприятии.

Также в работе представлен метод непрерывной аутентификации, использующий психологические реакции пользователей для формирования дополнительного защитного фактора в условиях повышенного риска реализации угроз информационной безопасности. Разработанный подход предусматривает анализ физиологических и поведенческих характеристик пользователей, что позволяет существенно повысить уровень защищённости за счёт учёта уникальных психофизиологических параметров каждого субъекта доступа и снизить вероятность несанкционированного проникновения.

В работе к.т.н. С. В. Корниенко, А. В. Пантюхиной «Методика выявления потенциальных внутренних нарушителей информационной безопасности» [95] предлагается проводить оценку психологического состояния сотрудников предприятия по их клавиатурному подчерку. Данная методика позволяет выявлять потенциальных нарушителей по динамике изменения следующих параметров: интервал между нажатиями клавиш, количество опечаток. В данной работе представлена прикладная программа, функционал которой при выявлении значимых отклонений от эталонных «нормальных» значений администратору безопасности передается сообщение. Данная прикладная программа работает без применения нейронных сетей, но применяется эталонный профиль клавиатурного почерка пользователя.

Все три вышеназванные работы были последовательно опубликованы в 2007, 2022, 2023, 2025 годах соответственно, что говорит о растущих перспективах анализа психологического фактора в сфере информационной безопасности. Сам факт устойчивого интереса к данной проблематике на протяжении столь длительного периода (с 2007 г. по настоящее время, 2026 год) свидетельствует о высоком потенциале и растущих перспективах учёта психологического фактора в сфере обеспечения информационной безопасности. Помимо этого, следует отметить, что в 2022 году средство защиты информации, реализующее анализ психологических характеристик индивидуально для каждого пользователя, было успешно внедрено в работу предприятия, а в 2025 году была защищена докторская диссертация, в которой был представлен и реализован метод с элементами анализа психологических данных.

5.2. ОБЗОР НАУЧНЫХ ПУБЛИКАЦИЙ ПО ВОПРОСАМ АНТИФРОДА

В работе «Методика обеспечения безопасности транзакций на основе использования антифрод-системы» предложена методика поддержки защиты от мошеннических транзакций с применением инструментов машинного обучения. Новизна исследования заключается в схеме взаимодействия подсистем антифрод-

механизмов с использованием ручной проверки экспертом. В работе авторы представили детальную схему взаимодействия подсистем антифрод-сервиса, включающую клиента, глобальные фильтры, модуль машинного обучения и системы правил, модуль логирования транзакций. Авторы предлагаю методику, схема которой содержит еще один элемент – проверку транзакции экспертом (человеком) [96].

В работе «AI-Driven Fraud Detection in Financial Transactions – Using Machine Learning and Deep Learning to Detect Anomalies and Fraudulent Activities in Banking and E-Commerce Transactions» представлен детальный обобщающий обзор методов искусственного интеллекта для решения задачи антифрода [97]. В начале работы авторы раскрывают информацию о состоянии вопроса решении задач антифрода, в том числе описывают применяемые методы. Авторы применили три открытых набора данных о транзакциях в своем исследовании эффективности методов машинного обучения и глубокого обучения, включая набор о транзакциях, совершенных с помощью мобильных приложений, затем произвели поэтапную предобработку данных для повышения их качества с помощью статистических методов, выявили и извлекли ключевые признаки, применили метод анализа компонент. Такой подход позволил произвести более детальный обзор. Результаты данного исследования подтвердили эффективность применения методов машинного обучения и глубокого обучения для решения задачи антифрода, но авторы отмечают, что сверточные нейронные сети и LSTM нейронные сети показали более высокую точность при выявлении сложных мошеннических схем в сравнении с методами классического машинного обучения. Авторы отмечают проблемы, связанные с применением ИИ-методов в решении задачи антифрода, такие как с непрозрачностью принятий решений и возможностью переобучения на небольших наборах данных. Авторы подчеркивают важность объединения различных подходов с целью применения преимуществ нескольких методов при обработке датасета, отмечают проблему ложного блокирования. Помимо этого, авторы отмечают, что большинство исследований по

решению задачи антифрода затрагивают либо банковские транзакции, либо электронную коммерцию, игнорируя межотраслевые проблемы обнаружения мошенничества [97].

В работе «Методы машинного обучения в задаче оценки риска мошенничества в автостраховании» представлен разработанный метод, который позволяет улучшить разделяющую способность классификатора с помощью повышения качестве данных. Повышение качества данных осуществляется посредством нейронной сети, в отличие от большинства подобных работ, где применяют такие методы как SMOTE [92]. Данная работа показывает значимость предварительной обработки данных.

В автореферате диссертации «Исследования по разработке методов противодействия мошенничеству в финансовых организациях с применением машинного обучения» продолжается вектор исследований по теме повышения качества данных с помощью нейронной сети, предложен новый способ комбинирования экспертного подхода, традиционного для сферы страхования, и методов машинного обучения [98]. Автором предложен подход повышения качества систем фрод-мониторинга путем расширения анализируемого признакового пространства: банковские операции обогащаются за счет интеграций с историей покупок, претензии обогащаются за счет построения графа связей между участниками страховых событий [98]. В работе представлен метод, снижающий количество ложных срабатываний, разработаны методики оценивания данных методов.

Датасет, содержащий сведения о клавиатурном почерке и поведенческих особенностях

Ни одно из представленный выше исследований не предполагало обучения нейронных сетей, решающих задачу антифрода, на эталонном профиле пользователя по его биометрическому клавиатурному почерку. Все выше представленные работы включали обработку банковских транзакций, информации о частоте переводов, то есть обработка была направлена на поиск признаков мошеннических схем, но такие схемы характеризуются постоянными адаптациями и

изменениями, предлагается обучать нейронные сети на эталонном поведенческом профиле пользователей банковской системы.

В рамках настоящего раздела выполнен обзор публикаций, авторы которых осуществляли сбор характеристик клавиатурного почерка пользователей, поведенческих особенностей, и использовали их в целях совершенствования обеспечения информационной безопасности.

В публикации [99] «Применение искусственных нейронных сетей для выявления аномального поведения пользователей центров обработки данных» описан способ обнаружения несанкционированного доступа к базам данных, базирующийся на анализе SQL-запросов, инициируемых администратором. Нейросетевая модель сопоставляет поступающие запросы с ранее выполненными и легитимными для данной базы данных. При отсутствии сходства текущего запроса с типичными для этой базы транзакциями последняя классифицируется как аномальная. В набор учитываемых признаков входят: дата и время поступления запроса, источник запроса, а также сведения о самом событии (запросе).

В работе под названием «Метод обнаружения инцидентов информационной безопасности по аномалиям в биометрических поведенческих чертах пользователя» [100] предлагается подход, базирующийся на применении методов машинного обучения. В рамках данного подхода нейросетевая модель анализирует такие поведенческие характеристики пользователя, как особенности клавиатурного почерка, временные паттерны активности и предпочтения в выборе программного обеспечения. Фиксация отклонений перечисленных параметров от эталонных значений интерпретируется системой как признак компрометации учётной записи. В ходе экспериментальных исследований авторами была сформирована выборка данных с участием 138 пользователей, однако полученный массив данных не был выложен в открытый доступ.

В 2022 году был опубликован в открытом доступе датасет, собранный группой исследователей, описание представлено в работе «Набор данных поведенческой биометрии для идентификации и аутентификации пользователей» [101]. Авторы

работы предлагают использовать характеристики клавиатурного почерка пользователей в качестве дополнительного фактора аутентификации, так как такие факторы, как пароль, код из сообщения могут быть украдены. Возникает препятствие в виде отсутствия публичных датасетов, содержащих необходимые данные, публикуемый авторами датасет данную проблему решает. Сбор данных о характеристиках клавиатурного почерка проводился с помощью формы ввода реквизитов банковской карты: номер, ФИО владельца, срок действия. Исследователи смогли аутентифицировать и идентифицировать пользователей по собранным данным, используя всего 3 признака, полученных путем вычисления среднего. Характеристики данных признаков представлены в таблице 5.1.

Таблица 5.1. Характеристики признаков в датасете

Признак	Характеристика
dwell_avg	Средняя продолжительность нажатия клавиш на клавиатуре.
flight_avg	Средняя продолжительность удержания клавиш на клавиатуре.
traj_avg	Среднее расстояние, пройденное компьютерной мышью за одно движение.

Датасет собран при участии 88 добровольцев, процедура сбора данных была следующей: каждый участник 10 раз вводил данные вымышленных карт, которые якобы принадлежали ему, и 10 раз — данные вымышленных карт, владельцем которых он не был. Общий объем датасета составил 1760 экземпляров (88 пользователей × 20 итераций). Авторы исследования доказали применимость собранного датасета, обучив дерево решений.

К сожалению, подобный способ для повышения качества систем антифрода не подходит, так как детектирование фрода происходит после ввода карты. Из-за высокого развития банковской системы переводы выполняются практически мгновенно, поэтому ситуация, в которой мошенник уже имеет доступ к реквизитам карты, является для нас неприемлемой.

Необходимо расширить признаковое пространство датасета таким образом, чтобы выявлять действия злоумышленников в режиме реального времени.

Подобное замечание касается некоторых других средств антифрода — пост-транзакционная обработка может привести к ущербу, несмотря на выявление действий злоумышленника. В условиях современных высокоскоростных платежных систем, пост-транзакционный подход может быть недостаточным. Именно эта причина требует перехода от единоразовой статической аутентификации к непрерывной аутентификации, из чего следует необходимость расширения признакового пространства датасета, на котором будут в дальнейшем обучаться нейронные сети.

Суть непрерывной аутентификации заключается в постоянном анализе не отдельных транзакций, а всего потока действий пользователя, начиная с момента установления сеанса связи [102]. Применительно к поставленной задаче это означает, что алгоритм выявления аномалий должен начать функционировать до того, как платёжное поручение будет авторизовано. Анализ действий злоумышленника — самого факта ввода карточных данных, навигации по форме платежа, характерных пауз и ошибок — должен дать сигнал к блокировке операции на ранней стадии её инициирования. Такой подход снижает зависимость от «чистоты» истории транзакций и переводит антифрод в плоскость прогностической аналитики, и, возможно, поведенческой экономики.

5.3. ОБЗОР НАУЧНЫХ ПУБЛИКАЦИЙ ПО ВОПРОСАМ АНАЛИЗА ПОВЕДЕНИЯ И ПСИХОЛОГИЧЕСКИХ ФАКТОРОВ В ИССЛЕДОВАНИЯХ

5.3.1. Психология уязвимости и принятия решений

Большинство мошеннических схем, реализованных за последние годы, основаны на психологическом воздействии на жертву: мошенники активно применяют различные методы социальной инженерии, чтобы побудить жертву предоставить доступ к финансам, например, через передачу кода из СМС. Атаки

мошенников становятся более «целевыми» и продуманными, злоумышленники все чаще изучают особенности, семейное положение жертвы, воздействуя на более уязвимых представителей близкого окружения[13]. Мошенники также используют фишинговые атаки, основанные на дополнительных данных об интересах или положении жертвы.

В связи с этим очень интересным является текстовое описание доклада доцента СПбГУ О. В. Медяник на Уральском форуме «Кибербезопасность в финансах — 2025»[14]. На форуме был проведен анализ звуковой записи телефонного разговора мошенника и пользователя банковской системы, в результате которого пользователь, являющийся дееспособным, выполнил все указания злоумышленника добровольно. Эксперт утверждает, что мошенники действуют по специальным текстовым программам, называемым «скриптами», которые вероятно были разработаны профессионалами в области психологии и психолингвистики. Злоумышленник, используя эти тактики и механизмы, воздействуют на когнитивное состояние человека, с целью вызвать эмоциональный отклик, который затруднит критическое мышление. Поэтому, утверждает эксперт, практически каждый человек при нужной последовательности действий, может поддаться уговорам мошенников. Эксперт отметила, что в ходе анализа большого количества скриптов, выявлены основные стратегии злоумышленников, такие как начало разговора с создания ложного чувства доверия, а затем настаивание на срочности совершения какого-либо действия. Эксперт также отметила эффективность введения технических, организационных, правовых мер против фрода.

---

[13] Средний ущерб от киберпреступлений в России вырос на 5% в 2025 году // РБК. [Электронный ресурс]. Режим доступа: https://www.rbc.ru/rbcfreenews/694bf7d69a794756d832ace0 (дата обращения: 11.05.2026).

[14] Как не стать жертвой телефонного мошенника: советы эксперта СПбГУ https://spbu.ru/news-events/krupnym-planom/kak-ne-stat-zhertvoy-telefonnogo-moshennika-sovety-eksperta-spbgu

Вопросы фрода также рассматриваются учеными и специалистами в области права[15]. Участники обсудили, кто чаще всего становится жертвой мошенников, отметили развитие мошеннических схем: если раньше злоумышленники использовали различные просьбы о помощи от лица родственников, то сегодня все чаще называются специалистами служб безопасности, используют спам-рассылки, зараженное компьютерными вирусами программное обеспечение. Для подготовки к фроду теперь используются новые технологии: с помощью ИИ мошенники подделывают голоса. Отмечается, что банки постепенно адаптируют модели кредитного скоринга для детектирования заявок на кредитование, отправленных под влиянием мошенников. *Участники обсудили возможность проведения комплексных мер, направленных на определение особого психологического состояния человека, находящегося под влиянием мошенников, например, при оформлении сделок с недвижимостью.*

Специалисты в области психологии и социологии из СПбГУ определили психотипы, обладатели которых могут оказаться более уязвимыми перед воздействием мошенников[16]. Исследованиями подтверждено, что привлекательная внешность правонарушителей способствует формированию у потерпевших повышенного доверия, причём данная тенденция не зависит от социально-демографических характеристик (возраста, пола, профессионального статуса, уровня дохода) и места жительства обманутых лиц. Решающими факторами, определяющими подверженность влиянию привлекательных злоумышленников, выступают типологическая структура личности индивида и степень сформированности его критического мышления. Наибольшую

[15] «Стать жертвой мошенника может абсолютно любой». В СПбГУ состоялся круглый стол о киберпреступлениях // Санкт-Петербургский государственный университет. [Электронный ресурс]. Режим доступа: https://spbu.ru/news-events/novosti/stat-zhertvoy-moshennika-mozhet-absolyutno-lyuboy-v-spbgu-sostoyalsya-kruglyy (дата обращения: 11.05.2026).

[16] Ученые СПбГУ определили психотипы доверчивых жертв мошенничества // Санкт-Петербургский государственный университет. [Электронный ресурс]. Режим доступа: https://spbu.ru/news-events/novosti/uchenye-spbgu-opredelili-psikhotipy-doverchivykh-zhertv-moshennichestva (дата обращения: 11.05.2026).

уязвимость перед манипуляциями демонстрируют индивиды с доверчивым типом поведения и те, кто ориентируется на первое впечатление, не проводя углублённого анализа. Менее восприимчивыми признаны лица тревожного типа (доверие зависит от эмоционального фона) и недоверчивого типа (склонны верифицировать информацию). Рациональный психотип характеризуется критическим мышлением, вниманием к невербальным сигналам и деталям внешности собеседника, что минимизирует риск быть обманутым. Наличие экономического образования может уменьшать уязвимость перед манипуляциями мошенников. При этом привлекательная внешность мошенников ошибочно интерпретируется как признак надёжности, что провоцирует необдуманные финансовые решения.

В тезисах диссертации от 2012 «Willing to be scammed: how self-control impact Internet scam compliance» [103] представлены психологические факторы, способствующие подверженности мошенническим схемам в Интернете. Утверждается, что наиболее значимым предиктором согласия на обман является уровень самоконтроля индивида. Исследование также выявило значимую роль таких признаков, как подверженность социальному влиянию (например, какого-либо авторитета, признанного обществом), потребность в последовательности совершаемых действий (видимо, это означает склонность человека при выполнении простой просьбы выполнить после нее другую, более сложную просьбу из-за сложности выразить отказ сразу после согласия, например, согласие прослушать рекламное сообщение, а затем, после недолгих уговоров, все же перейти по присланной ссылке). Также исследование созвучно выводам О.В. Медяник в упомянутом выше докладе – мошенники используют эмоциональные и когнитивные предубеждения, делая жертву более восприимчивой к обману.

С учетом того, что в 2025 году заметно возросло количество несовершеннолетних жертв мошенников, полезной может быть статья «Scam Susceptibility: Determining the Dominant Factor for an Adolescent's Decision-making» от 2019 года, посвященная изучению уязвимости подростков перед мошенническими схемами [104]. В исследовании приняли участие 73 ученика средней школы, 52 из

них прошли психологический тест на тип личности. Затем участникам были представлены различные электронные письма, которые нужно было проанализировать и отнести к определенной категории: отправленные мошенниками или настоящими пользователями. Исследователи утверждают, что значимой характеристикой, определяющей склонность к уязвимости к фроду, оказался уровень дохода, но в выборке было представлено всего несколько человек с низким уровнем дохода, поэтому сложно сделать однозначные выводы. Затем был сделан вывод, что тип личности не влияет на общий результат, но влияет на то, как человек рассуждает. Все участники при анализе писем рассуждали в правильном ключе, но обладателей типа личности INFP(интроверт (I), интуитивного типа (N), ориентирующийся при принятии решений на других людей (F), открыт к новому опыту(P)) рассуждения привели к неверным решениям, например, при наличии деталей в письме, участники могли ошибочно пометить его как «настоящее». Несмотря на небольшую выборку (всего 73 участника), работа даёт обоснование для дифференцированного подхода в профилактике фишинга.

5.3.2. Типы личностей и «Бриллиант мошенничества»

Полезной для исследования может быть популярная модель психологии MBTI или ее версия NERIS Type Explorer, примененная в работе [104]. MBTI подразумевает кодирование типов личности с помощью букв: первая буква описывает, является ли человек экстравертом(E) или интровертом(I), вторая буква в коде описывает, как человек анализирует информацию, если он доверяет фактам, конкретике, то это сенсорный тип (S), если ориентируется на абстрактные связи, паттерны, генерирует гипотезы и ищет скрытые смыслы, то это интуитивный тип(N), третья буква описывает способ принятия решений, особенно в ситуациях морального выбора, людям, отдающим предпочтение логическим доводам, ставят букву T (thinking, мышление), но если человек ориентируется на субъективные ценности, гармонию в отношениях, принимает решение с точки зрения того, как последствия повлияют на других,

то в кодировку ставят букву F (feeling, чувства). Последняя шкала описывает предпочитаемый стиль организации внешней жизни и отношение к дедлайнам, структуре и гибкости, если человек предпочитает структурированную, запланированную среду, то в кодировке ставится буква J (Judging, суждение), при предпочтении человека гибкости, открытости новому опыту ставится буква P (Perceiving, восприятие). Всего в этой модели существует 16 типов личности.

Специалисты в области психологии, социологии и криминологии также исследуют явление мошенничества со стороны самого злоумышленника, в том числе обстоятельства, которые подталкивают человека к совершению мошеннических действий. В 1953 году социологом и криминологом Дональдом Р. Кресси была представлена концепция, получившая название «Треугольник мошенничества» [105]. В данной работе показано, что для совершения мошеннических действий у злоумышленника должен быть стимул (мощный внутренний или внешний мотив, толкающий человека на путь обмана), должна существовать возможность совершения действий, а также рационализация, позволяющая оправдать собственные действия.

В 2004 году была представлена модель «Бриллиант мошенничества», которая представляет собой расширение модели с тремя условиями. В отличие от «Треугольника», в «Бриллианте» добавляется четвертое условие – способность, под которой подразумеваются знания, навыки, уверенность в себе человека, который потенциально может совершить мошеннические действия. Добавленный элемент показывает, что мошеннику необходимо обладать дополнительными характеристиками, которые отличают его от других людей, находящихся в схожих условиях, но не совершивших мошенничество.

5.3.3. Синтез данных

Возможно ли заменить реальные данные на синтезированные в исследованиях в сфере информационной безопасности? Целесообразно ли повышать качество методов синтеза и

разрабатывать новые? Не являются ли синтезированные данные временным «костылем» для исследований, ожидающих сбора реальных данных?

В статье «Генерация синтетических данных для систем интеллектуального анализа в задаче обнаружения вредоносного программного обеспечения» представлен ход работ по синтезу данных для систем обнаружения вредоносного программного обеспечения [106].

На данный момент качество синтезированных данных уступает качеству реальных. Но синтез данных целесообразен для исследований, в которых в силу внешних обстоятельств сложно получить реальные данные. Также синтез данных может быть полезен как средство моделирования поведения пользователей или субъектов экономики.

5.4. СБОР ДАТАСЕТА С РАСШИРЕННЫМ ПРИЗНАКОВЫМ ПРОСТРАНСТВОМ

5.4.1. Сбор данных – разработка собственного симулятора банковского приложения и алгоритма сбора данных

Разработано программное средство, предназначенное для сбора эмпирических данных, на основе которых строится датасет с расширенным признаковым пространством. Помимо этого, разработан алгоритм, обеспечивающий регистрацию клавиатурного почерка (уточнение – здесь и далее под клавиатурным почерком подразумевается не только работа с физическим устройством – клавиатурой, но и с сенсорным экраном смартфона, в том числе операции без использования экранной клавиатуры смартфона, например, простые нажатия на элементы пользовательского интерфейса, например, кнопки) как одной из разновидностей поведенческой биометрии пользователей. На рисунке 5.2 представлен внешний вид пользовательского интерфейса разработанного приложения.

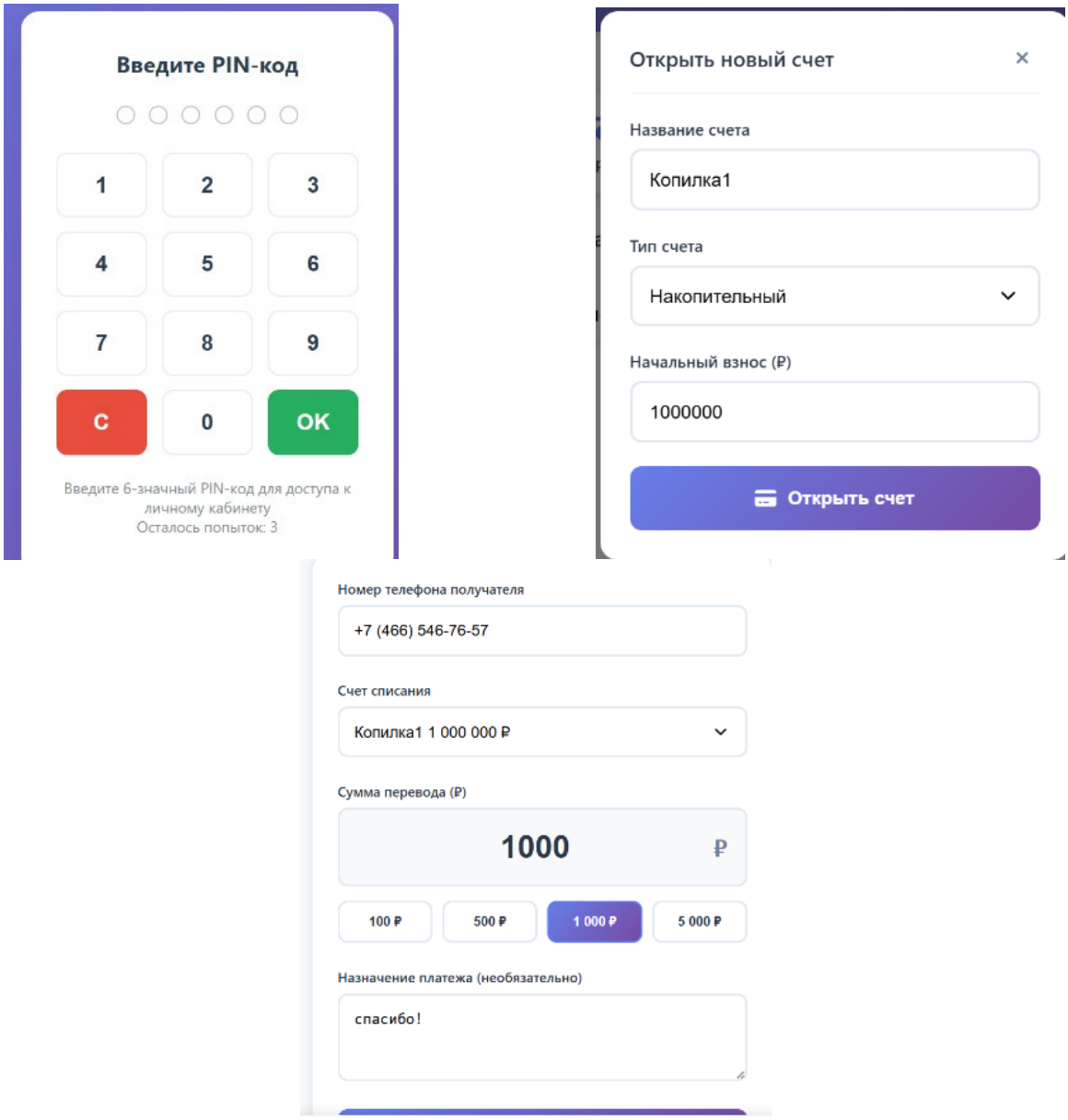


Рис. 5.2. Симулятор банковского приложения

Частью приложения является трекер, реализующий разработанный алгоритм сбора данных. Трекер реализован на языках программирования PHP, JavaScript. Приложение-симулятор разработано на языках программирования PHP, JavaScript с применением HTML и CSS.

В результате анализа нескольких банковских приложений был изучен способ работы пользователей с банковскими приложениями. Первое, что обычно делает пользователь при работе с банковским приложением – это вводит PIN-код. После успешного ввода кода пользователь получает доступ к управлению

банковскими счетами, может совершать переводы и просматривать различные банковские приложения. Но самым первым этапом взаимодействия считается ввод PIN-кода. При этом страница ввода кода отличается от страницы, в которой пользователю предоставляется доступ к банковскому функционалу: страница с вводом кода практически никогда не меняется, на ней расположены только кнопки с цифрами от 0 до 9. Очевидно, что алгоритм сбора данных о работе пользователя на странице ввода кода будет сильно отличаться от алгоритма для страницы со счетами, картами и вкладами. Статичность экрана ввода и малое число интерактивных элементов способствуют быстрому формированию устойчивых моторных паттернов при наборе PIN-кода. Большинство пользователей демонстрируют небрежную манеру ввода, не опасаясь ошибок, поскольку система обычно поддерживает возможность повторного ввода. Исключением являются ситуации нахождения пользователя в неблагоприятной среде, когда возникает потребность скрыть последовательность вводимых символов от окружающих, при этом некоторые пользователи могут изменять угол наклона экрана, тем самым меняя характер работы с экраном, или понижают яркость экрана, затрудняя считывание символов, что также может изменить характеристики нажатий. Именно поэтому алгоритм сбора данных о вводе PIN-кода отличается от алгоритма сбора данных на странице с банковским функционалом.

В таблицах 5.2 и 5.3 представлены признаки, которые регистрируются при вводе пользователем кода. Отметим, что помимо общих для любого события типа «клик», при завершении каждой попытки ввода кода, неважно успешной или нет, собираются также усредненные характеристики, как в работе [102], характеристики, указывающие на то, ошибся ли пользователь при вводе кода, и сколько времени ушло на данную попытку ввода. Курсивом в таблице отмечены характеристики, которые расширяют признаковое пространство датасета – то есть, новые признаки, которые автор работы решил добавить.

Добавление новых признаков позволит повысить качество решения задачи антифрода, позволив выявлять действия злоумышленника в режиме реального времени.

Разработанный алгоритм, внедренный в приложение-симулятор, позволяет собирать данные в независимости от устройства, с помощью которого пользователь осуществляет вход в банковскую систему – смартфон, настольный компьютер или ноутбук. Интерфейс разработанного приложения-симулятора является адаптивным, что обеспечивает корректное отображение на экранах как стационарных компьютеров, так и мобильных устройств. В таблице 2 представлены признаки, характеризующие моторные паттерны пользователя при вводе PIN-кода, а также содержащие информацию об успешности/неуспешности попытки входа в приложение по введенному набору цифр.

Таблица 5.2. Описание действий пользователя при вводе PIN

Признак	Характеристика
dwell_avg	Среднее время удержания кнопки за все клики текущей попытки ввода PIN.
flight_avg	Средний межкликовый интервал (исключая первый клик, где интервал не определён).
traj_avg	*Средняя длина траектории перемещения между последовательными кликами.*
time_to_complete	*Полное время, затраченное на ввод всей последовательности цифр от первого клика до момента отправки запроса.*
pin_attempt	*Последовательность введенных цифр.*
remaining_attempts	*Количество попыток ввода, оставшихся до блокировки ввода PIN.*
blocked_until	*Дата и время, до которого ввод PIN заблокирован при трех неудачных попытках ввода. Сохраняется при событии pin_blocked.*
pin_success / pin_failure	*Успешная попытка ввода/ неуспешная попытка ввода.*

Для событий типа «click», выполняемых в личном кабинете и при вводе PIN, сохраняются сведения, представленные в табл. 5.3.

Таблица 5.3. Признаки, собираемые при событиях типа «pin_click», «click»

Признак	Характеристика
dwell_time	*Время удержания пальца/курсора на кнопке от момента касания до отпускания.*
flight_time	*Временной промежуток между текущим и предыдущим кликом. Для первого клика в сессии равно 0.*
trajectory_distance	*Евклидово расстояние между координатами предыдущего и текущего клика:* $\sqrt{((x_2-x_1)^2 + (y_2-y_1)^2}$*. Если предыдущего клика нет, равно 0.*
coordinates_x, coordinates_y	Координаты точки касания относительно левого верхнего угла окна браузера. Округление до целых.
click_time	Дата и время клика с учетом миллисекунд.
click_element	*Идентификатор элемента пользовательского интерфейса.*
way_of_work	Способ работы с приложением – настольный компьютер или мобильное устройство.
page	Название применяемой страницы.

Приложение также собирает сведения о событиях типа «скроллинг» или «вращение колесом мыши», представленные в табл. 5.4. Добавление в признаковое пространство датасета характеристик о событиях прокрутки позволяет сформировать уникальный поведенческий профиль пользователя, который будет сложнее подделать. События прокрутки происходят во время всего периода работы с приложением, даже если пользователь запускает сеанс работы с банковской системой с целью просмотра состояния счетов и не совершает никаких кликов, поэтому так важно обрабатывать информацию об этом типе событий.

Таблица 5.4. Признаки, собираемые при событиях типа «scroll», «wheel»

Признак	Характеристика
Вращение колесом мыши	
deltaX *deltaY*	*Величина горизонтальной прокрутки. Отрицательное значение означает прокрутку влево, положительное – вправо.*
deltaMode	*Браузер определяет единицы измерения, в которых заданы deltaX и deltaY:* *0 – пиксели (WheelEvent.DOM_DELTA_PIXEL). Это наиболее распространённый режим. при обычном вращении колёсика deltaY часто равно ±120 пикселям (зависит от настроек мыши и ОС).* *1 – строки (DOM_DELTA_LINE). Величина соответствует количеству строк текста, на которое предполагается прокрутить содержимое.* *2 – страницы (DOM_DELTA_PAGE). Прокрутка на целую страницу (например, при использовании клавиш PageUp/PageDown).*
Скроллинг	
scroll_speed	*Время остановки в одной локации. Показывает, сколько времени в среднем мышь или стилус, человеческий палей проводят в одной точке перед следующим перемещением.*
scroll_distance	*Изменение позиции прокрутки за текущий интервал в пикселях.*
scroll_position	*Позиция window.pageYOffset на момент события.*
scroll_direction	*Направление движения вверх или вниз.*
total_scroll_distance	*Сумма всех scroll_distance с начала сессии.*
way_of_work	Способ работы с приложением – десктоп или мобильное устройство.
page	Название применяемой страницы.

Выполняется сбор данных о событиях, связанных с набором текста или нажатием специальных клавиш. Событие типа «key_press» формируется в трекере приложения-симулятора при отпускании клавиши («keyup»). Регистрируется факт нажатия клавиши на клавиатуре (физическом устройстве или клавиатуре телефона), включая время удержания и контекст ввода. Для данного типа событий собираются стандартные признаки, характерные для событий типа «click»: dwell_time, flight_time, trajectory_distance, timestamp, event_type. Дополнительно собираются признаки, представленные в таблице 5.5.

Таблица 5.5. Признаки, собираемые при событиях типа «key_press», «key_special»

Признак	Характеристика
«key_press», «key_special»	
key	Символ или название клавиши (например, 'a', 'Enter', 'ArrowUp')/ часто не распознается.
key_code	Код клавиши в стандарте event.code (например, 'KeyA', 'Digit1')
key_code_legacy	Устаревший числовой код клавиши (event.keyCode)
key_location	Положение клавиши на клавиатуре: 0 – стандартное, 1 – левая, 2 – правая.
special_key_type	Тип специальной клавиши (если определена как специальная). Например «Enter», «Ctrl».
modifiers.ctrl	Показывает, была ли зажата клавиша Ctrl во время нажатия. Логический тип.
modifiers.shift	Показывает, была ли зажата клавиша Shift. Логический тип.
modifiers.alt	Показывает, была ли зажата клавиша Alt. Логический тип.
modifiers.meta	Показывает, была ли зажата клавиша (Win / Cmd). Логический тип.
is_composition	Показывает, является ли нажатие частью композиции ввода (например, иероглифы). Логический тип.

element_type	Тег элемента, на котором произошло событие (например, 'input', 'textarea')
element_input_type	Тип поля ввода (например, 'text', 'number', 'tel', 'password')

Одновременное применение «key_code_legacy» и «key_code» позволяет собирать информацию в новых и более ранних версиях браузеров. Строка с типом события «key_special» содержит почти те же поля, что и «key_press», но с dwell_time = 0 и с обязательным заполнением special_key_type. «Key_special» генерируется на keydown, а не на keyup. Такой подход необходим для оперативной фиксации нажатий специальных клавиш, которые могут не генерировать keyup (например, при комбинациях с Ctrl, Alt, или при блокировке повтора).

Помимо перечисленных выше типов событий и их характеристик, собирается также информация о событиях, несвязанных с движениями пользователя. Такие события характеризуют конкретные действия, такие как открытие модальной формы, перенос фокуса с одного элемента пользовательского интерфейса на другой, начало и конец сеанса работы с приложение.

То есть таким событиям предшествуют события типа «клик», но они регистрируются отдельно, так как несут дополнительную информацию о работе пользователя с приложением.

Таблица 5.6. Дополнительные типы событий

Тип события	Характеристика
События начала и конца активного сеанса работы с приложением (transfer_session_start, account_session_start, transfer_session_end, account_session_end)	Содержат информацию о точной дате начала и конца, вплоть до секунды, активного сеанса работы с приложением. Позволяют уточнить привычное время работы с приложением для конкретного пользователя.
modal_open – открытие любого	Содержит информацию о точной дате открытия модального окна, конкретном

модального окна (общий для всех модальных окон).	пользовательском элементе, который был открыт.
deposit_modal_open – открытие окна пополнения счёта.	Информация о точной дате открытия модального окна пополнения счета.
События focus_in и focus_out	Регистрация момента, когда элемент получает или теряет фокус ввода – состояние, при котором элемент пользовательского интерфейса готов принимать ввод с клавиатуры или готов к взаимодействию, например, нажатие кнопки.
Событие ввода текста input	Событие input срабатывает каждый раз, когда пользователь изменяет значение элемента <input>, <textarea> или любого элемента с атрибутом contenteditable. Оно генерируется после каждого ввода символа, вставки текста и удаления. Особенно необходимо на мобильных устройствах, где события keydown/keyup могут срабатывать с задержкой или не срабатывать вовсе для экранной клавиатуры.
Событие отправки формы form_submit	Регистрирует отправку любой формы.
Событие отправки формы transfer_submit	Регистрирует отправку формы с переводом средств другому пользователю банковской системы. Сохраняет информацию о счете отправителя, телефон получателя, сумме перевода, описание перевода (например «На подарок»).

При анализе поведенческих данных важен не только сам факт совершения действия, но и его смысл, для более детального понимания действий пользователя, а также для возможности

восстановления последовательности действий необходимо собирать информацию о событиях, представленных в таблице 5.6.

В собранных данных существуют некоторые проблемы, такие как наличие большого количества незначащих нулей. Данная проблема возникает из-за того, что для большинства событий, представленных в таблицах 5.4–5.6, не собирается информация о таких параметрах как dwell_time, flight_time, trajectory_distance, эти параметры в таблице имеют значение 0. Так трекер, реализующий алгоритм сбора данных позволяет разделить физические события (нажатия), и логические (открытие модальных форм, отправка перевода).

Для решения данной проблемы необходимо детально проанализировать собранные данные, произвести качественную предобработку данных, так как наличие большого количества незначащих нулей может негативно повлиять на качество обучения нейронных сетей, решающих задачу антифрода.

5.5. РАЗВЕДОЧНЫЙ АНАЛИЗ СОБРАННЫХ ДАННЫХ

Исходный датасет, сформированный по данным из банковского приложения, содержит 41 признак. Необходимо проанализировать признаки на предмет мультиколлинеарности и выявить комбинации, пригодные для объединения. Результатом станет изменение структуры итогового датасета и повышение эффективности обучения нейронной сети, получение полной информации о собранных данных. Проведем поэтапно разведочный анализ полученного массива данных.

5.5.1. Анализ пропусков в данных

Анализ пропусков (missing data analysis) — это совокупность методов выявления, количественной оценки, диагностики механизмов возникновения и статистической обработки отсутствующих значений в наборе данных. В контексте построения систем, применяющих методы поведенческой биометрии и

обнаружения аномалий анализ пропусков является критическим этапом предобработки данных, поскольку игнорирование или некорректная обработка пропусков приводит к смещению оценок, потере статистической мощности и снижению обобщающей способности моделей машинного обучения.

Данные о поведении пользователей неравномерны по определению: некоторые признаки не существуют для определенных типов событий, поэтому наличие пропусков в поведенческих данных является закономерным явлением. Также пропуски могут возникать даже для тех событий, для которых признак должен заполняться – в этом случае необходимо выяснить механизм возникновения пропусков.

Выделяют три основных механизма возникновения пропусков: MCAR (Missing Completely at Random) – для каждой записи набора данных вероятность пропуска одинакова, MAR (Missing at Random) – данные пропущены не случайно, а из-за определенных факторов, MNAR (Missing Not at Random) – пропуск данных вызван неизвестными факторами [107].

В таблице 5.7 представлены первые 10 признаков с наибольшим количеством пропусков. Признаки «key_code», «operation_amount», «operation_type» не заполняются для большинства типов событий, поэтому в них больше 90% пропусков.

Таблица 5.7. Признаки с наибольшим количеством пропусков

Признак	Процент пропусков
session_end	100%
device_type	100%
screen_resolution	100%
session_duration	100%
browser_info	100%
click_accuracy	100%
key_code	99.916388
operation_amount	97.881828
operation_type	95.624303
session_start	94.816054

На рисунке 5.3 представлена матрица пропусков для первых 20 признаков, отметим, что визуально наблюдается значительное количество пропусков для признаков, заполняющихся только для типов событий «scroll», «input», так как доля этих событий значительно меньше доли событий типа «click». Возможно, что анализ и обработку событий в дальнейшем придется разделить по типам.

На рисунке 5.4 представлена дендрограмма пропусков для всех признаков, представленных в массиве данных. Дендрограмма позволила визуализировать иерархическую кластеризацию признаков на основе схожести паттернов пропусков: подтверждается гипотеза об отсутствии некоторых признаков для распространенных типов событий как причины большого количества пропусков (более 90%). Отметим, что у признаков screen_resolution, device_type, session_duration, session_end, click_accuracy, browser_info наблюдается практически идентичная структура пропусков, что говорит о едином механизме отсутствия. У этой группы признаков наблюдается связь с признаком key_code. В отдельную группу по идентичности паттернов пропусков объединяются признаки scroll_speed, scroll_dictance, scroll_position, scroll_direction, total_scroll_distance, wheel_delta_x, wheel_delta_y характеризующие движения скроллинга. Аналогично, объединяются в группы по схожести паттернов такие признаки, как dwell_time, flight_time, click_time, trajectory_distance, click_time, coordinates_y, coordinates_x, input_type, относящиеся к событию типа click, у этой группы признаков наблюдается слабая связь с признаком click_element.

Группа признаков key_code_legacy, key_code, key, key_location, special_key_type, modifiers_ctrl, modifiers_shift, modifiers_alt, modifiers_meta, is_composition обладают идентичностью паттернов пропуска данных, образуют слабые связи с группой признаков element_type, element_input_type. Обе группы характеризуют нажатие клавиши на клавиатуре.

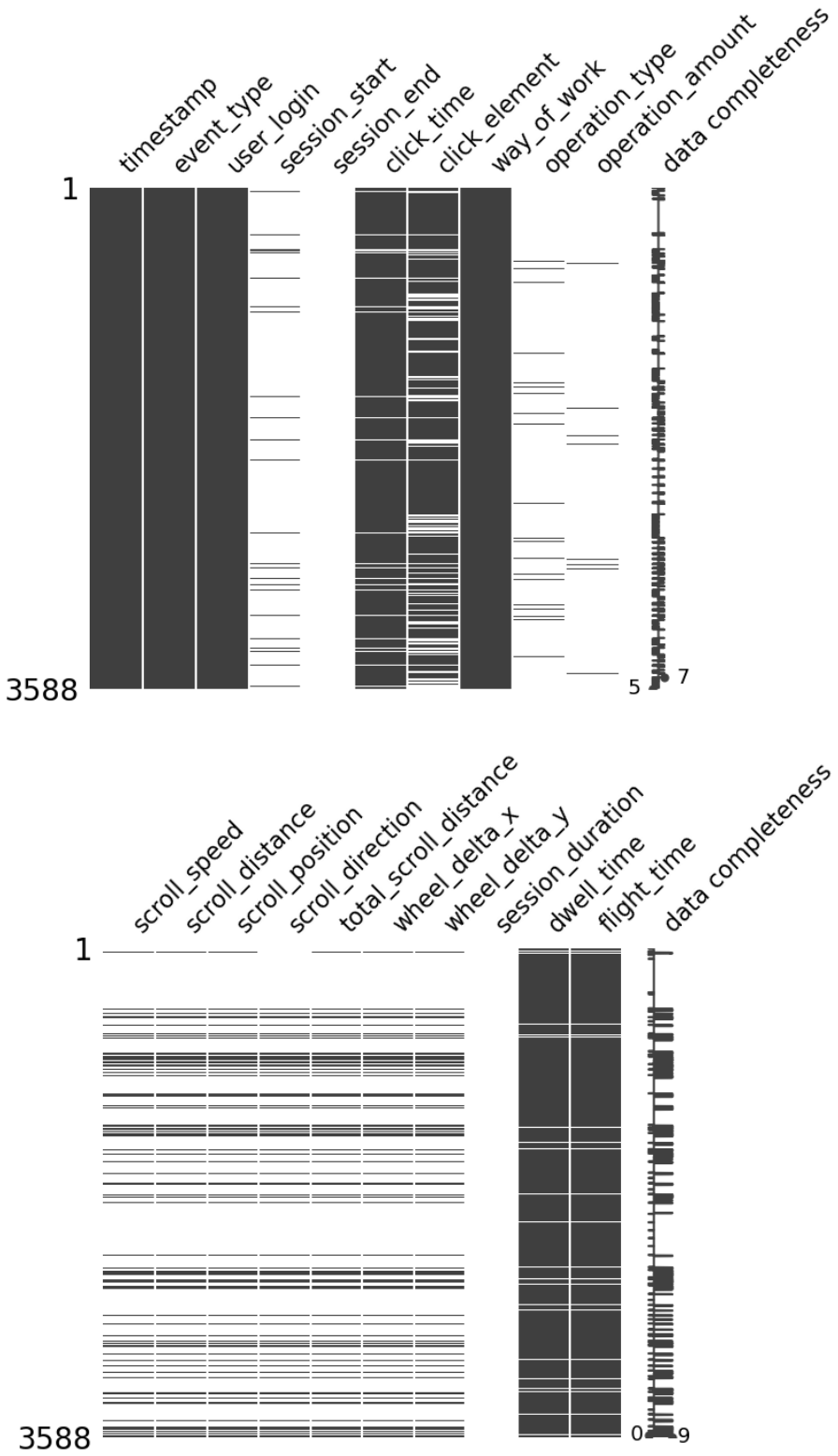


Рис. 5.3. Матрица пропусков для первых 20 признаков

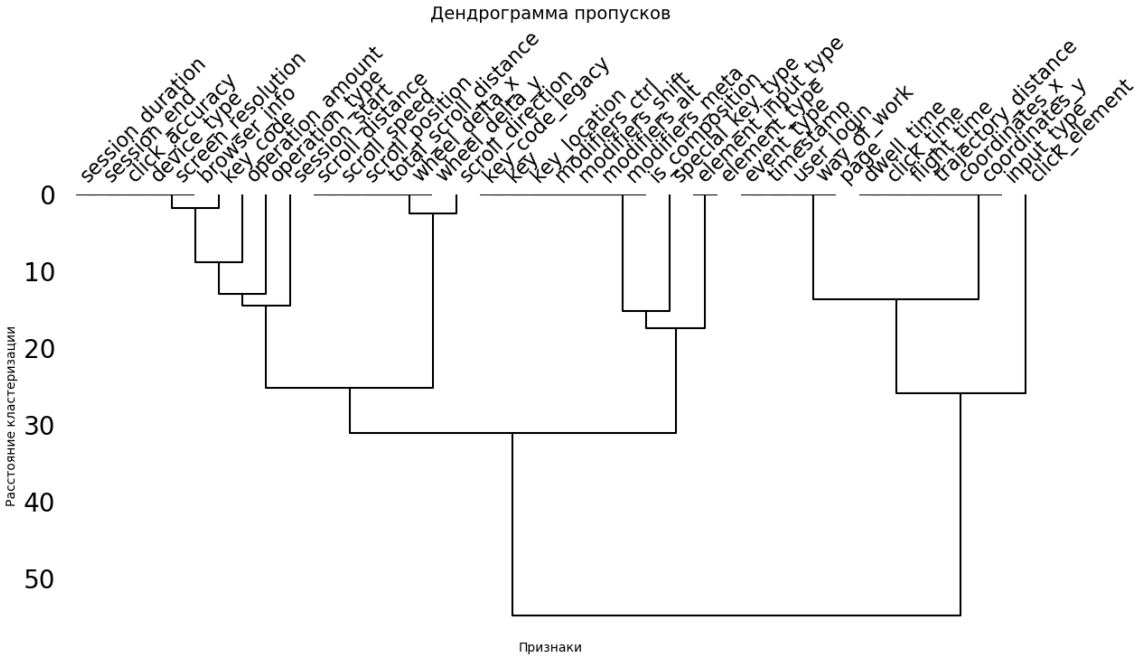


Рис. 5.4. Дендрограмма пропусков для всех признаков из массива данных

Анализ пропусков в данных показал, что следующие признаки содержат большое количество пропусков (>90%) из-за возможных настроек браузера на стороне пользователя (например, отключение JavaScript): click_accuracy, device_type, screen_resolution, browser_info, session_duration. Возможно, следует вычислить их на этапе предобработки данных или удалить из итогового датасета. Механизм возникновения пропуска в представленных признаках относится к типу MNAR, поскольку конфигурация браузера пользователя является неизвестным и ненаблюдаемым фактором. Пропуски в таком случае могут являться информативными сами по себе: мошенники часто используют приватные режимы браузера, повышают уровень конфиденциальности с помощью различных инструментов, что увеличивает вероятность пропусков для технических признаков, таких как тип устройства или информация о браузере. Возможно добавление дополнительного признака, характеризующего отсутствие технических признаков в данных, что может указывать на деятельность злоумышленника, особенно, если раннее в данных таких пропусков не наблюдалось.

Так как пропусков в представленных выше признаках более 90%, а в некоторых, таких как browser_info их 100%, возникает вопрос о

целесообразности включения представленных признаков в итоговый датасет, предназначенный для обучения нейронной сети.

Проанализируем характер пропусков для каждого признака, возможность восстановления данных и их потенциальную информативность, результаты представлены в таблице 5.8.

Таблица 5.8. Анализ пропусков

Признак	Оценка возможности восстановления	Потенциальная информативность
click_accuracy	Непосредственно вычислить невозможно без исходных координат клика и координат целевого элемента. Вычислить возможно, но сложно, возможно ошибочные вычисления, так как неизвестен формат отображения пользовательского интерфейса на стороне клиента, из-за чего сложно получить координаты целевого элемента.	Высокая
device_type	Можно восстановить из базы данных при условии сохранения информации об устройстве пользователя.	Высокая
screen_resolution	Пропуски вызваны невозможностью получить данные (например, из-за ограничений браузера). Восстановить из других данных нельзя.	Высокая
browser_info	Можно восстановить из базы данных приложения	Средняя

	при условии сохранения информации о браузере пользователя.	
session_duration	Можно вычислить по признаку «timestamp», который почти не содержит пропусков.	Высокая

По результатам анализа можно сделать вывод, что необходимо вычислить признак «session_duration», также по возможности восстановить признаки, характеризующие способ работы с приложением, такие как тип устройства, характеристики браузера. Остальные признаки необходимо удалить из датасета или произвести дополнительную обработку, в виде добавления дополнительных бинарных признаков, маркирующих пропуск или незначащее нулевое значение. К сожалению, представленные пропуски обусловлены неконтролируемыми факторами, такими как настройки браузера на стороне пользователя, поэтому в будущем необходимо продолжить обработку пропусков данных, так как исключить факторы полностью невозможно.

5.5.2. Построение и анализ графиков для временных рядов

Для проведения анализа данных необходимо визуализировать временные ряды, чтобы выявить возможные проблемы в данных до того, как будет сформирован датасет. При своевременном выявлении проблем проще выбрать методы очищения и предобработки данных.

На рисунке 5.5 представлен график значений dwell_time для каждого события. Данные не были предобработаны, значения собраны для всех типов событий, в том числе для типов, для которых dwell_time автоматически заполняется незначащими нулями. При визуальном анализе графика заметно, насколько много незначащих нулей находится в данных. Аналогичный анализ был проведен для графиков flight_time, trajectory_distance. Значения временного ряда были скорректированы с целью защитить

информацию о поведении пользователя 3 в приложении-симуляторе.

Рис. 5.5. График значений dwell_time по всем типам событий (с коррекцией)

График значений dwell_time по событиям типа «click», представленный на рисунке 5.6, представляет динамику значений признака за промежуток с 22 января по 15 февраля 2026. При визуальном сравнении графиков на рисунках 5.3 и 5.4 заметно, насколько меньше нулевых значений признака dwell_time в данных для событий типа «click».

Для собранных данных характерна перенасыщенность незначащими структурными нулевыми значениями, которые могут негативно повлиять на качество обучения моделей нейронных сетей. Необходимо учитывать данный фактор при предобработке данных, подготовке архитектуры модели, разработав или применив новые способы работы с данными и моделями НС.

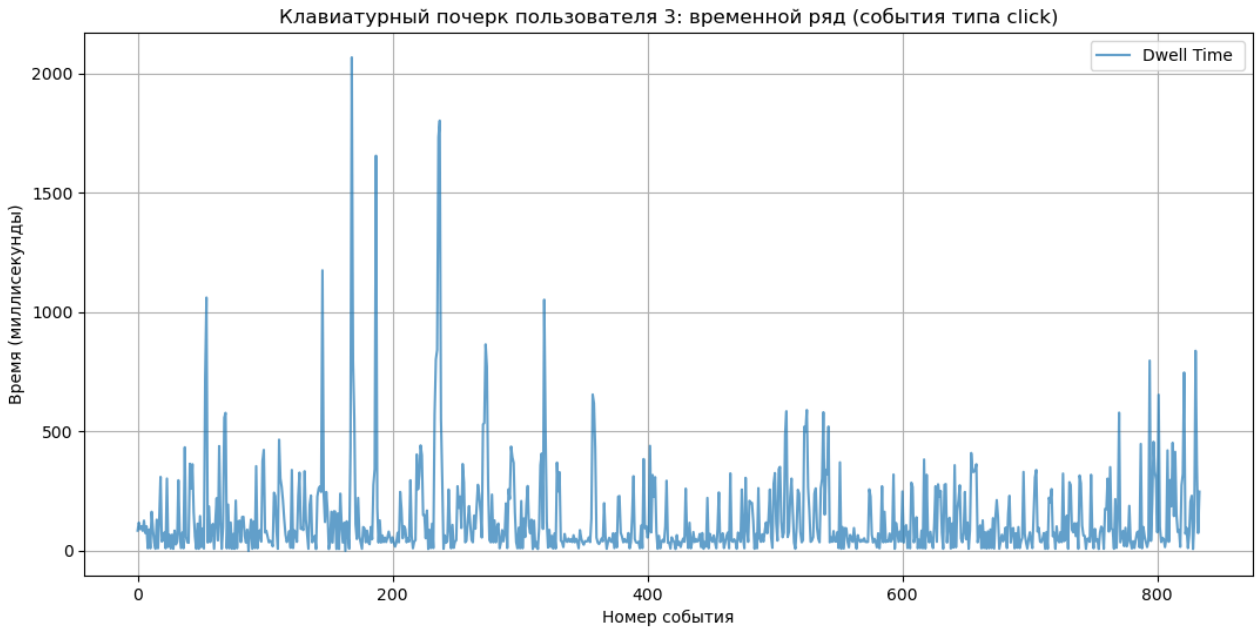


Рис. 5.6. График значений dwell_time по событиям типа click (с коррекцией)

Поиск линейных и нелинейных связей

Для поиска линейных связей между парами признаков в собранных данных применим матрицу корреляции.

Для числовых признаков dwell_time, flight_time, trajectory_distance, coordinates_x, coordinates_y была построена матрица корреляции. Коэффициенты, представлены на рисунке 5.7, сохраняются в приближенном виде для всех пользователей, данные которых собраны в датасете.

В результате анализа матрицы корреляции, представленной на рисунке 5.7, сделаны следующие выводы. Признаки «coordinates_x» и «coordinates_y» сильно коррелируют (коэффициент равен 0.79), что ожидаемо, так как при движении пальца или мыши оба эти признака изменяются одновременно.

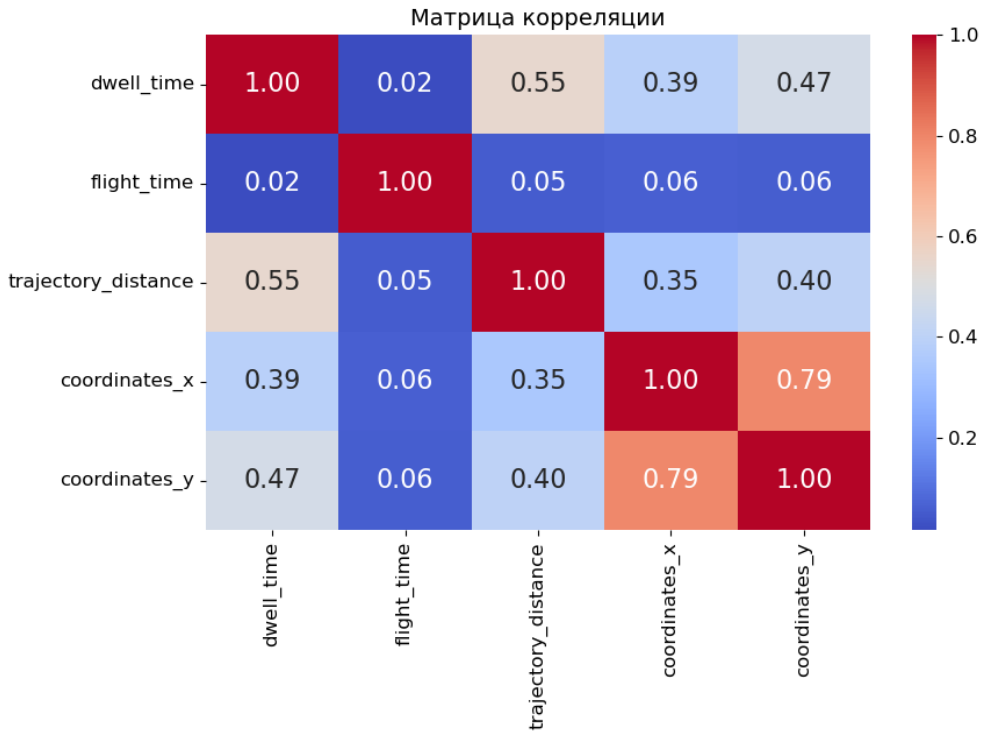


Рис. 5.7. Матрица корреляции для избранных признаков для всех типов событий

Высокий уровень корреляции говорит о том, что эти признаки могут дублировать друг друга, а значит, нужно заменить их новым признаком, который зависит от значения обоих, или же просто оставить один из них. Умеренно положительные корреляции демонстрируют признаки «dwell_time» и «trajectory_distance», что вполне логично: чем большее расстояние проходит палец или мышь по экрану, тем больше времени указатель (далее – синоним пальца пользователя или мыши) задерживается на целевом элементе пользовательского интерфейса. Это может отражать одновременную моторную активность (например, пользователь совершает длинное движение мыши и одновременно удерживает клавишу, или же не отрывает палец от экрана, описывает траекторию).

Естественная связь прослеживается между признаками «trajectory_distance» и «coordinates_x», «trajectory_distance» и «coordinates_y»: чем большую дистанцию проходит указатель, тем большее смещение у координат. Уровень корреляции последних

трех пар признаков не настолько высок, чтобы считать признаки избыточными.

Признаки dwell_time и flight_time почти не коррелируют (коэффициент равен 0.016) – значит, время удержания клавиши и время между нажатиями клавиш измеряют разные аспекты клавиатурного почерка. Данные признаки не дублируют друг друга, их одновременное нахождение в датасете может быть полезным.

Интересен признак flight_time – он практически не коррелирует с остальными признаками (все коэффициенты не более 0.06), представленными на рисунке 5.7.

Подобный фактор дублируется у остальных пользователей, принимавших участие в формировании датасета.

Можно предположить, что данный признак ортогонален остальным, а значит, его значения практически не зависят от значений других признаков, представленных в матрице. Следовательно, такой признак обладает наиболее высоким потенциалом при обучении моделей для решения задачи антифрода, так как он может нести уникальную информацию о поведении каждого пользователя.

Но данный анализ не является точным, так как анализируются все представленные в собранных данных типы событий, кроме событий ввода PIN-кода, но как сказано выше, для многих типов событий числовые признаки заполняются незначащими нулями, что может повлиять на результат анализа. Для более полного и детального анализа вычислим матрицы корреляций для событий типа «click», «key_press».

На рисунке 5.8 представлена матрица корреляции для двух типов событий «click» и «key_press».

Интерпретация парных коэффициентов данной матрицы показала, что корреляция между dwell_time и flight_time практически отсутствует, как и в общей матрице.

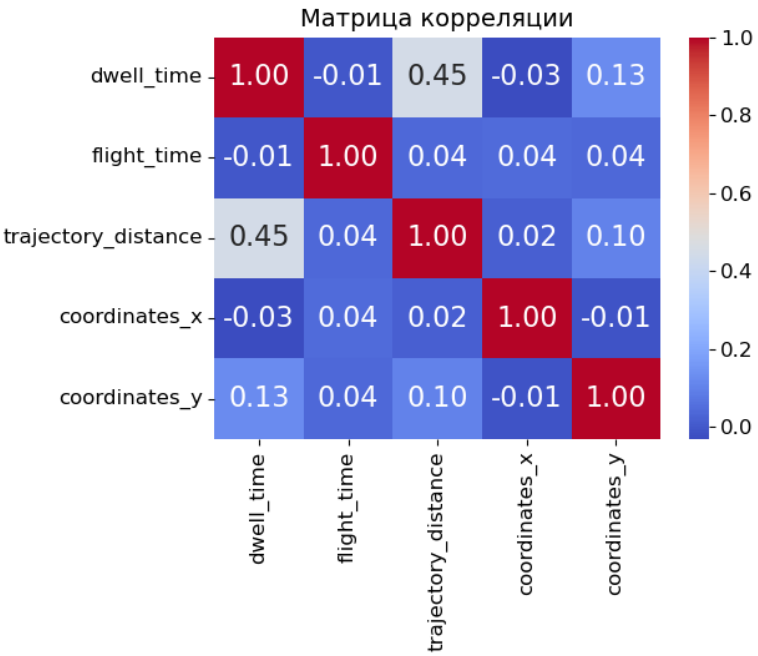


Рис. 5.8. Матрица корреляции для событий типа «click» «key_press»

Для dwell_time и trajectory_distance сохраняется умеренная корреляция, при этом наблюдается слабая корреляция dwell_time с coordinates_y (координаты по оси Y) и нулевая корреляция с координатами по оси X (-0.03). Признак flight_time практически ортогонален всем остальным признакам, что подтверждает наблюдения по матрице, представленной на рисунке 5.5. Признак **flight_time является очень полезным признаком** для поведенческой аутентификации, поскольку он несёт уникальную информацию, не дублируемую другими параметрами.

Длина траектории за одно передвижение указателя (trajectory_distance) почти не коррелирует с конкретными координатами, как и сами координаты между собой.

У представленной матрицы корреляции есть недостатки – она описывает линейные связи для двух несвязанных типов событий. Необходимо провести анализ корреляционных матриц для каждого события отдельно. Также хранение значений координат как отдельных значений, а не непрерывной траектории является возможной причиной слабой корреляции с длиной траектории, хотя по логике данные признаки должны коррелировать.

На рисунке 5.9 представлена матрица корреляции для событий типа «click».

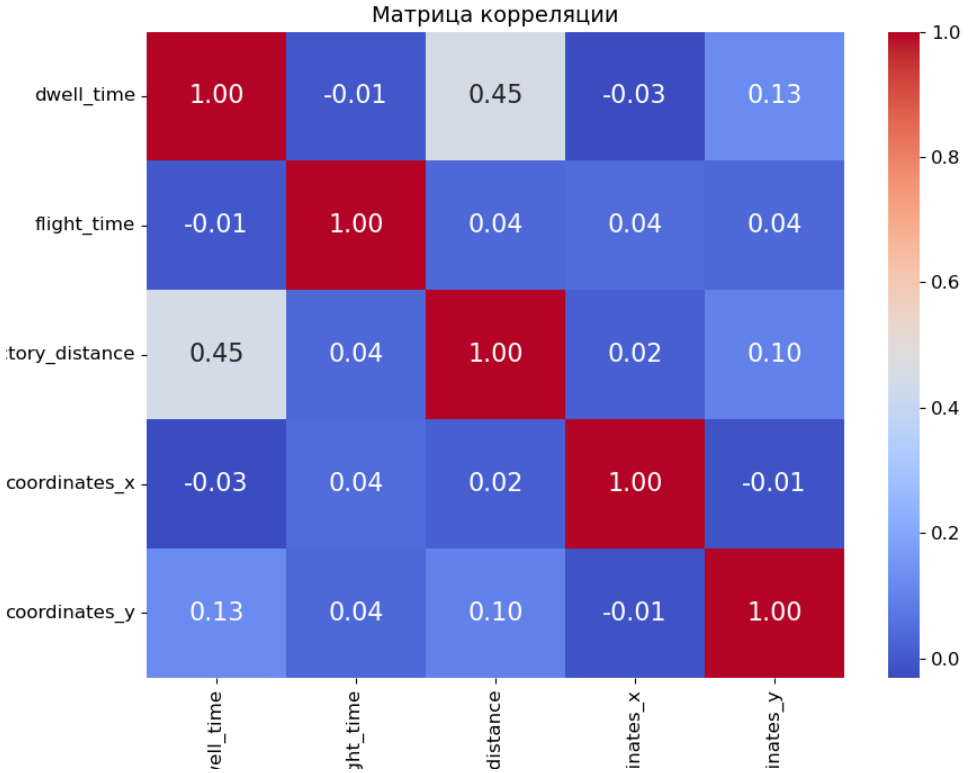


Рис. 5.9. Матрица корреляции для событий типа «click»

В результате анализа матрицы корреляции только для событий, связанных с нажатиями, подтверждается **ортогональность признака «flight_time» с остальными признаками**, представленными в матрице. Возможно, это очень ценный признак, так как он несёт информацию, которую нельзя получить ни из одного другого параметра – **«flight_time» необходимо включить в итоговый вариант датасета**. Для dwell_time trajectory_distance выявлена положительная корреляция (0.45), что может описывать устойчивый паттерн движения пользователя. Практически отсутствует корреляция между dwell_time и координатами, что подтверждает наблюдения выше. Отсутствует корреляция между признаками coordinates_x, coordinates_y, что позволяет включить оба эти признака в итоговый датасет, не опасаясь мультиколлинеарности, но возможно *провести эксперименты по замене обоих признаков новым усредненным признаком.*

На рисунке 5.10 представлена матрица корреляции для событий типа «key_press» с характерными для этого типа событий

признаками. По результатам анализа можно сделать вывод, что flight_time ортогонален остальным признакам, что подтверждает приведенные выше наблюдения.

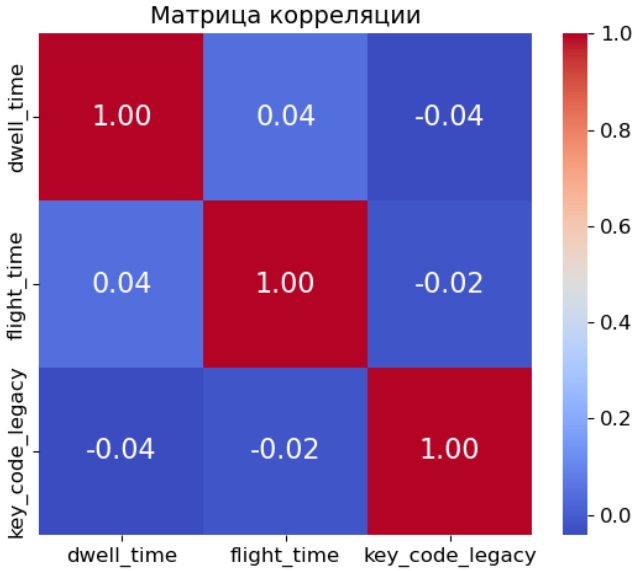


Рис. 5.10. Матрица корреляции для событий типа «key_press»

На рисунке 5.11 представлена матрица корреляции для данных, собранных при работе со страницей ввода PIN-кода.

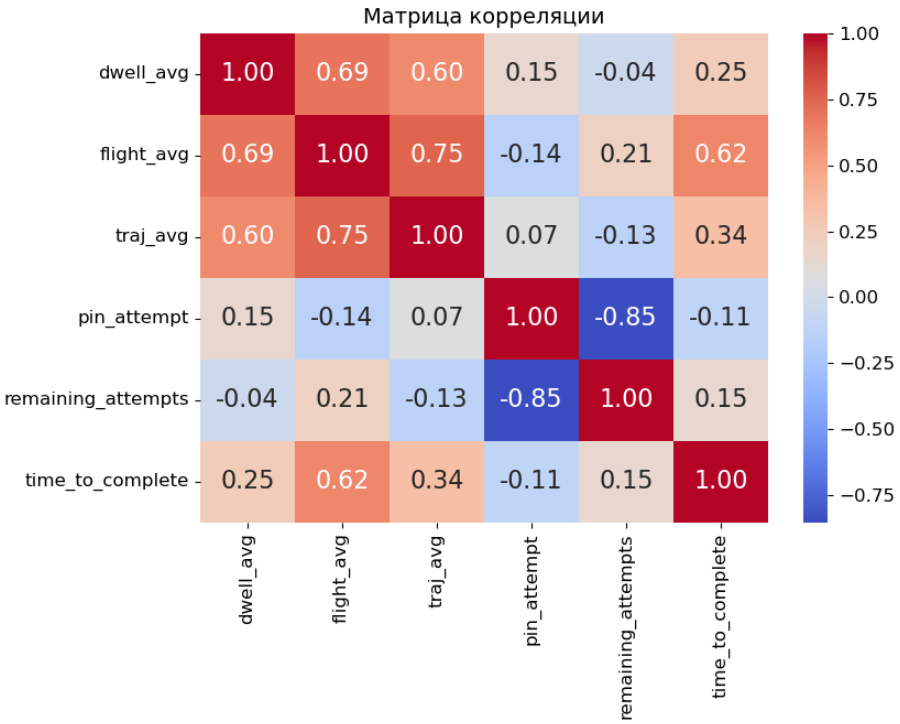


Рис. 5.11. Матрица корреляции для избранных признаков, собранных при работе со страницей ввода PIN-кода

Между flight_avg и traj_avg (коэффициент равен 0,75) наблюдается высокая положительная связь, что свидетельствует о согласованном изменении временных пауз между нажатиями клавиш и средней длины траектории движения указателя. Пара dwell_avg и flight_avg (0,69) также демонстрирует выраженную положительную корреляцию, указывающую на взаимозависимость времени удержания клавиши и интервалов нажатия клавиш. Значение коэффициента 0.85 между pin_attempt и remaining_attempts является ожидаемым, поскольку переменные связаны детерминированным соотношением (сумма потраченных и оставшихся попыток ввода равна константе). Данная пара признаков информационно избыточна, необходимо для обучения нейронной сети выбрать один признак. Коэффициенты связей pin_attempt с динамическими признаками не превышают по модулю 0,2, что говорит о практической независимости количества попыток от моторных паттернов пользователя. Возможно, при построении датасета, необходимо будет выбрать один из трех усредненных признаков: dwell_avg, flight_avg или traj_avg. Наибольшей потенциальной ценностью для задач обнаружения аномалий обладают признаки, слабо коррелирующие с остальными, – в частности, pin_attempt и, возможно, производные от него (например, количество ошибок или темп их нарастания).

Для данных, собранных со страницы ввода PIN-кода, существует возможность создания новых признаков, заменяющих сразу несколько исходных:

1. dwell_flight_ratio = dwell_avg / (flight_avg + ε) – характеризует тип моторики.
2. total_attempts_used – сколько попыток уже было потрачено.
3. avg_time_per_attempt = time_to_complete / pin_attempt – отражает эффективность ввода PIN-кода.

Таким образом, анализ корреляционных матриц показал, что структура *числовых признаков собранных данных состоит как минимум из двух семейств признаков* – характеристик моторных паттернов пользователя и характеристик конкретных действий пользователя, таких как попытки ввода PIN-кода или сумма

переведенных средств. В ходе развития исследования добавится третье семейство признаков – психологические характеристики пользователя, определяющие возможную уязвимость к мошенническим схемам.

5.5.2.1. Поиск нелинейных зависимостей

В представленном выше анализе применялся расчет матриц корреляции (Пирсона) для выявления линейных связей между парами признаков. Если связь между признаками будет нелинейной – например, квадратичной, то данный метод не позволит ее обнаружить. Поэтому для выявления нелинейных зависимостей применим другие инструменты.

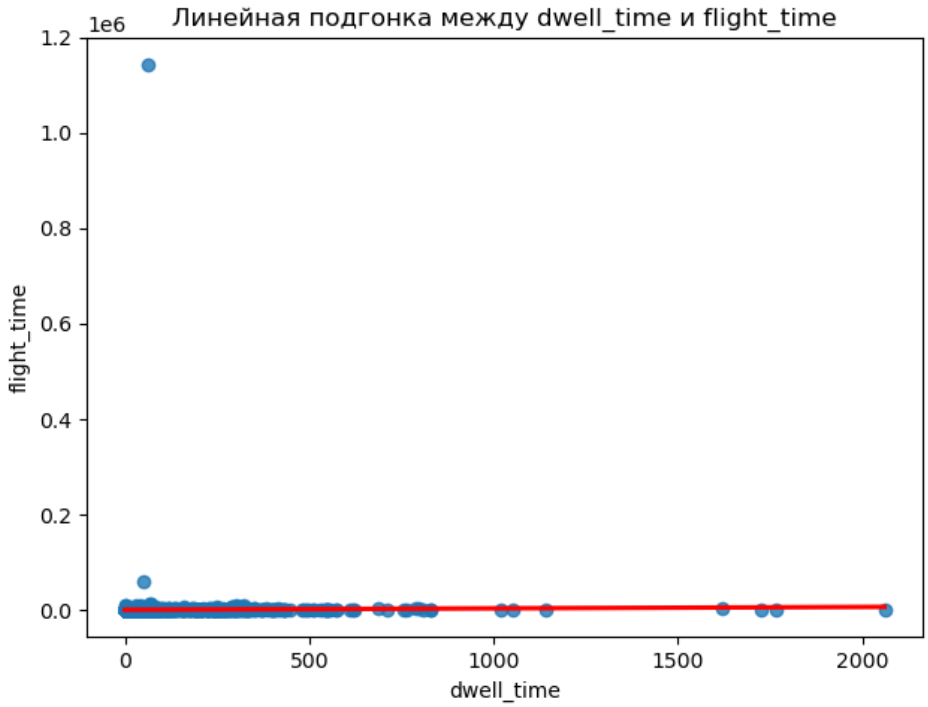


Рис. 5.12. График разброса значений для flight_time в зависимости от dwell_time, для всех типов событий

Видим, что график на рисунке 5.12 неинформативен, так как признаки flight_time, dwell_time содержат пропуски, если анализировать все типы событий сразу. Необходимо искать нелинейные зависимости подобно поиску линейных – с учетом особенностей поведенческих данных, в которых разные типы событий обладают различным набором признаков.

Для поиска нелинейных связей между парами признаков применим показатель предсказательной силы (PPS). Данная метрика используется в разведочном анализе данных для поиска любых значимых взаимосвязей между признаками, включая линейные, нелинейные связи. Признаки могут быть и числовыми, и категориальными, что полезно для массива собранных данных, содержащих категориальные признаки. Показатель принимает значения от 0 до 1, где 0 – это невозможность предсказания значения x по y, полное отсутствие какой-либо связи, а 1 – это идеальная предсказательная способность. В результате проведенного анализа выявлено большое количество нелинейных и категориальных связей между парами признаков.

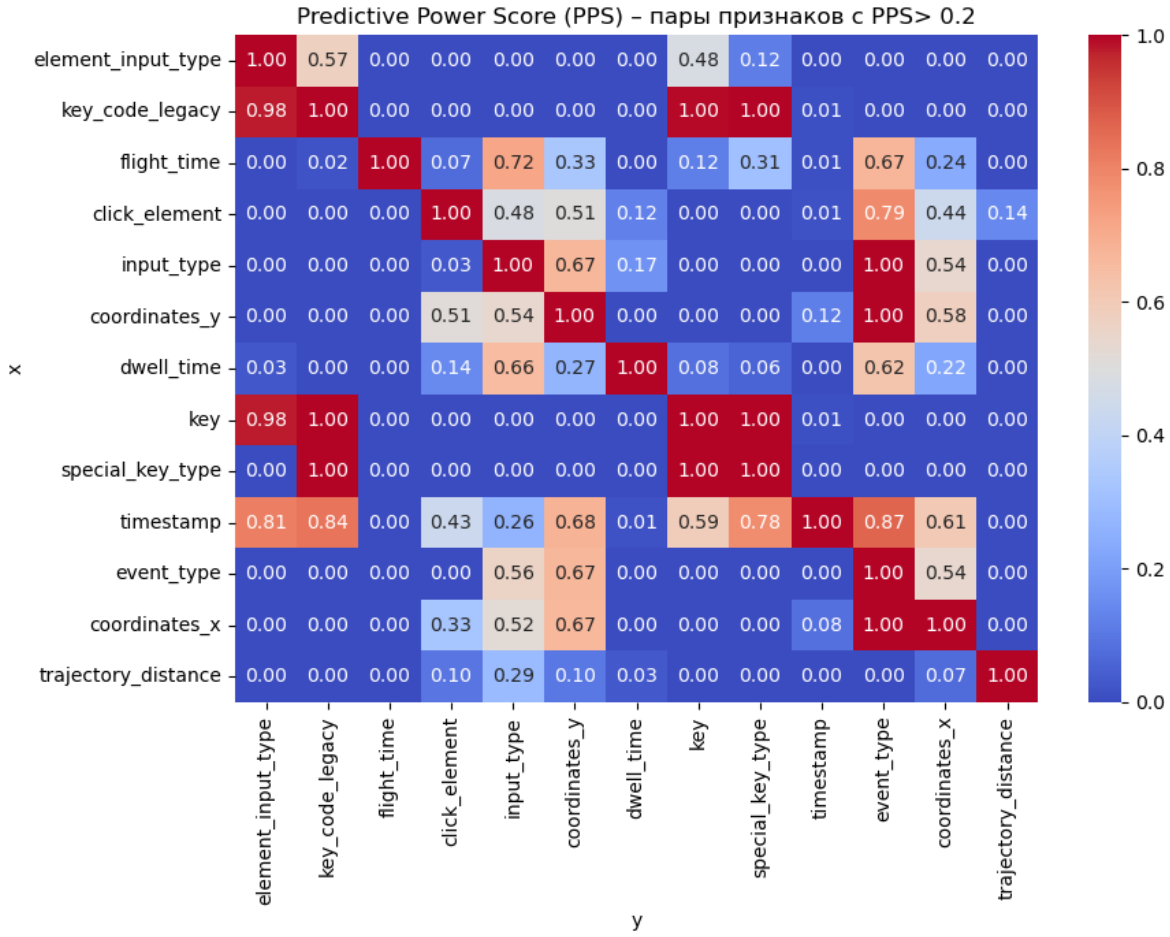


Рис. 5.13. Тепловая карта PPS для признаков с PPS> 0,2 событий click, key_press

Анализ тепловой карты показал, что признак flight_time не зависит от значений других признаков – в столбце с подписью

flight_time на пересечении с предикторами (ось X) показатели равны 0. Данный факт подтверждает ценность признака flight_time, необходимо включить данный признак в итоговый датасет. Строка flight_time (признак как предиктор) содержит много ненулевых значений: признак связан с input_type (0,72), event_type (0,67), coordinates_y (0,33), special_key_type (0,31), следовательно, flight_time является сильным предиктором для многих других признаков.

Следующие группы признаков предсказывают друг друга практически идеально: key_code_legacy, special_key_type, key, element_input_type (показатель выше 0,98, для некоторых пар равен 1,0), возможно, стоит оставить в итоговом датасете только 1 из этих признаков; event_type и input_type предсказывают друг друга с коэффициентами 0,56 и 1,00, но PPS (input_type, event_type) = 1.00, то есть input_type полностью определяет event_type. Рекомендовано дополнительно проанализировать эту пару для событий, описывающих клик, возможно, в итоговом датасета использовать только один из представленных.

Признак timestamp имеет высокие значения PPS (0,78–0,87) для предсказания key_code_legacy, element_input_type, special_key_type, e vent_type. Это может связано с порядком событий, однако использование абсолютного времени опасно из-за переобучения на конкретный диапазон дат. Возможно, в итоговом датасете стоит заменить абсолютные временные метки относительным временем, например, оставить только часы, минуты, секунды с начала суток или начала сессии.

Признаки coordinates_x и coordinates_y предсказывают друг друга с PPS 0,67 и 0,58 – это умеренная, но не полная зависимость. Оба хорошо предсказывают event_type и input_type (0,5–1,0). В итоговом датасете возможна замена признаков на новый, основанный на двух составляющих.

Признак dwell_time, как и flight_time, не предказывается ни одним из признаков, значит, потенциально является достаточно полезным, необходимо включить в итоговый датасет. При этом dwell_time имеет связи с признаками input_type (0,66), event_type (0,62), coordinates_y (0,27).

Признак trajectory_distance имеет слабую предсказательную силу, при этом сам также плохо предсказывается. Возможно удаление из итогового датасета с обязательной проверкой – не будет ли ухудшения качества обучения модели нейронной сети при удалении данного признака, так как у него существует линейная связь с dwell_time.

Таблица 5.8. Итоговые рекомендации для данных по типам событий click, key_press

Признаки	Рекомендация
special_key_type, key, element_input_type, event_type	Удаление, избыточны.
trajectory_distance	Удаление.
timestamp	Замена на относительное время.
flight_time, dwell_time, key_code_legacy, input_type, coordinates_x, coordinates_y, click_element	Не удалять, наиболее информативное «ядро» датасета

Все рекомендованные действия должны подтверждаться экспериментами по сравнению уменьшенного датасета по сравнению с полным массивом данных при обучении модели нейронной сети, так как существует риск, уменьшив размерность массива данных, потерять важные сведения. Если будет доказана избыточность некоторых признаков, то это может ускорить сбор данных и обучение и работу нейронной сети, что ускорит процесс выявление действий злоумышленника при решении реальных задачи антифрода в банковских системах.

5.5.2.2. Добавление психологических характеристик в датасет

На данный момент планируется добавление двух признаков – самостоятельная оценка пользователя, считает ли он себя подверженным к фроду, вторая оценка – попадался ли пользователь ранее на уловки мошенников (любых, в том числе мошенников, действующих оффлайн, использующих фишинг по

СМС или электронным письмам). Сейчас количество участников эксперимента мало, и нельзя достоверно сказать, действительно ли добавление именно таких признаков улучшает качество антифрода, но необходимо проводить эксперименты в этом направлении и сотрудничать с учеными в области социологии и психологии.

Среди участников эксперимента двое участников не пострадали от действий мошенников, так как вовремя приняли меры, но все же выполнили некоторые из требуемых мошенниками действий: перешли по ссылке, присланной в СМС, или совершили иные действия, которые требовали от них злоумышленники.

5.6. ПРЕДОБРАБОТКА ДАННЫХ

Вопрос предобработки данных в контексте построения нейросетевых моделей относится к числу фундаментальных, однако зачастую ему уделяется меньше внимания, чем выбору архитектуры или гиперпараметров. Некачественная предобработка может привести к существенному искажению при обучении нейронной сети. Ниже приведено систематическое обоснование необходимости проведения тщательной и детальной предобработки данных, основанной на выводах, полученных в результате разведочного анализа данных.

Большинство классических активационных функций — сигмоида, гиперболический тангенс, а также более современные, но имеющие режимы насыщения — демонстрируют нелинейное поведение лишь на ограниченном интервале входных значений. Для сигмоиды это диапазон [-3, 3]; при аргументе, превышающем по модулю 5, производная стремится к нулю.

Если на вход сети подаются сырые данные (например, значения яркости пикселей от 0 до 255), после суммирования с весами первого слоя аргумент функции активации легко оказывается в зоне насыщения. Градиент в этой области близок к нулю — явление «умирающих нейронов» или исчезающего градиента. Предобработка, приводящая данные к нулевому среднему и единичной дисперсии либо к интервалу [-1, 1],

гарантирует, что большая часть значений попадет в рабочую область с ненулевой производной.

Эмпирическое распределение признаков в реальных выборках почти всегда содержит аномальные выбросы — например, сбой в работе датчика, ошибка транскрипции или редкое, но легитимное событие. Нейронные сети, особенно с квадратичными функциями потерь (MSE), чувствительны к таким точкам. Выброс вызывает пропорциональный квадрату отклонения вклад в градиент, что приводит к смещению оценок весов в сторону аномалии.

Замена выбросов на медиану с последующей стандартизацией, существенно снижает это влияние. В случае работы с категориальными признаками, где кодирование может породить ложную численную близость, предобработка включает ортогональное кодирование (one-hot) или целенаправленное введение эмбеддингов, что исключает навязывание сетью упорядоченности неупорядоченным классам.

Таким образом, предобработка данных является одним из основополагающих этапов работы по созданию итогового датасета, который может существенно повлиять на качество обучения модели. Предобработка данных не менее важна, чем разработка архитектуры нейронной сети.

Для предобработки полученного массива данных произвели следующие шаги: удаление незначащих признаков, преобразование чисел с разделителем «запятая» в числа с разделителем «точка» (3,21 преобразован в «3.21»), удаление признаков, содержащих более 90% пропусков с последующей их заменой на новые признаки с восстановленными значениями (для восстановления провели дополнительный опрос среди пользователей), заполнение пропусков медианным значением, one-hot encoding для категориальных признаков. В результате предобработки количество признаков увеличилось с 41 до 123.

Были удалены следующие признаки: 'key','timestamp', 'session_start', 'session_end', 'click_time', 'screen_resolution', 'browser_info','session_duration', 'click_accuracy', так как часть из них содержала 90% и более пропусков и не подлежала восстановлению ('screen_resolution', 'click_accuracy'), часть могла привести к

переобучению ('timestamp', 'session_start', 'session_end', 'session_duration', 'click_time'), и часть была заменена новыми признаками ('screen_resolution', 'browser_info'). Были добавлены новые признаки: «is_mobile», характеризующий, использовал ли пользователь в данной сессии мобильное устройство, «browser», содержащий название браузера, с которого пользователь осуществлял доступ в приложение.

Все признаки были разделены на две группы: числовые и категориальные. К числовым отнесены все признаки, содержащие целые или действительные числа, в том числе признаки, которые содержали булевские типы. К категориальным отнесены такие признаки, как тип события (event_type).

Пропуски в числовых признаках были заполнены медианным значением для всех типов событий, но такой подход не отражал смысл датасета. Поэтому после замены пропусков на новое значение, для событий, тип которых не предполагает заполнения определенных признаков (например, scroll_speed для событий типа click), значение таких признаков изменили на 0. Также были добавлены дополнительные признаки-маркеры, например, is_scroll_event.

После обработки пропусков в числовых признаках выполняем стандартизацию – приведение каждого признака к нулевому среднему и единичной дисперсии. Так как в массиве собранных данных используются различные единицы измерения, например, пиксели и миллисекунды, необходимо привести их к общему виду. Это необходимо для корректного обучения нейронной сети – признаки с большими значениями могут ошибочно получить большую значимость, чем действительно важные признаки с меньшими значениями. После стандартизации массив данных принимает вид, сильно отличающийся от исходного.

Далее выполняется one-hot encoding для категориальных данных. Например, вместо заполнения столбца event_type такими значениями как click, key_press, model_open, и так далее, создаются новые признаки, такие как cat__event_type_click, cat__event_type_key_press. В результате данного этапа значительно

увеличивается количество признаков, соразмерно количеству значений категориальных признаков.

В исходной выборке насчитывался 41 признак. В результате выполнения процедур предобработки размерность признакового пространства возросла до 125, но это позволило добиться повышения информативности и репрезентативности набора данных.

5.7. ПОИСК АНОМАЛИЙ

5.7.1. Метод главных компонентов

Метод главных компонент (Principal Component Analysis, PCA) — это статистическая процедура, использующая ортогональное преобразование для перехода от исходного набора коррелированных переменных к новому набору некоррелированных переменных, называемых главными компонентами. Каждый последующий компонент выбирается таким образом, чтобы его дисперсия (разброс значений) была максимальной при условии ортогональности предыдущим.

В многомерных данных часто наблюдается избыточность: признаки могут быть связаны друг с другом (например, время удержания клавиши и пауза между нажатиями). PCA ищет направления в пространстве признаков, вдоль которых дисперсия данных максимальна. Первая главная компонента ориентирована вдоль направления наибольшей изменчивости. Вторая — перпендикулярно первой и вдоль направления следующей по величине дисперсии, и так далее.

Целью применения является снижение размерности данных перед обучением моделей машинного обучения (нейронные сети, SVM, линейная регрессия) для ускорения вычислений, уменьшения шума и предотвращения переобучения.

Для применения PCA необходимо, чтобы исходный массив данных был предобработан: состоял только из числовых значений

без пропусков, причем числовые значения должны быть стандартизированы.

В данном исследовании метод главных компонент был применен к предобработанному массиву данных. Рассмотрим детально результаты применения данного метода к исходному массиву данных.

Рисунок 5.14 состоит из двух графиков: столбчатой диаграммы индивидуальной объяснённой дисперсии и линейного графика накопленной (кумулятивной) дисперсии. Оба графика помогают оценить, насколько хорошо главные компоненты «сжимают» информацию из собранного массива данных.

Дисперсия – это мера разброса данных. Индивидуальная объяснённая дисперсия – доля общей дисперсии, которую объясняет одна конкретная компонента (PC1, PC2, ...). Накопленная (кумулятивная) дисперсия – суммарная доля дисперсии, объяснённая первыми k компонентами.

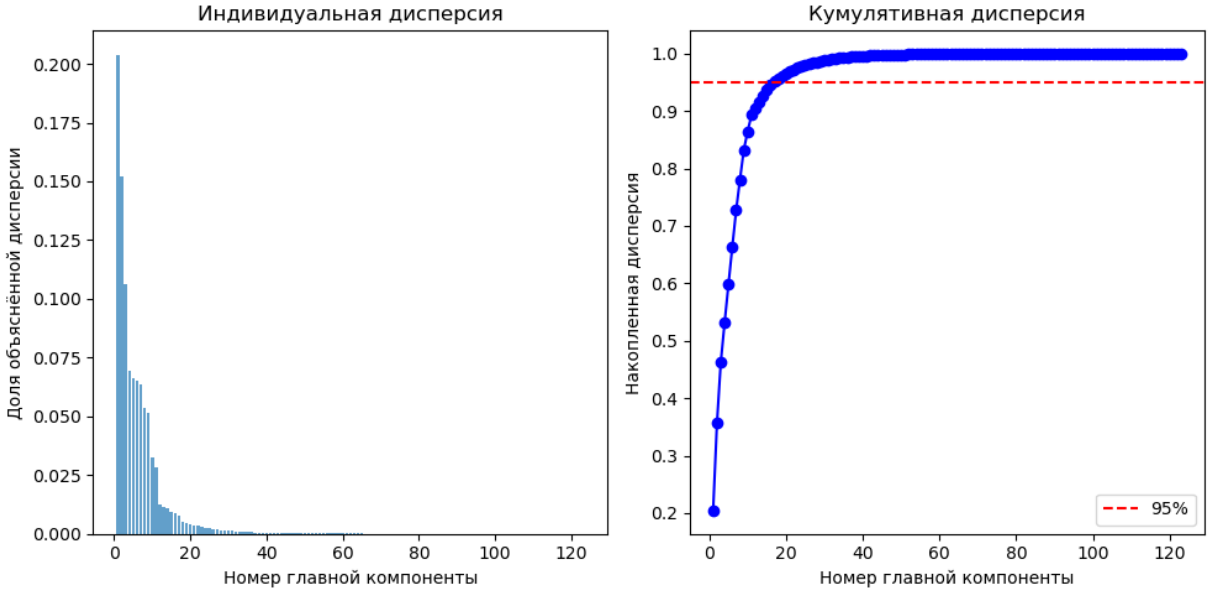


Рис. 5.14. Индивидуальная и кумулятивная дисперсия

На рисунке 5.14 (левый график) по оси X расположены номера компонент от 1 до 125 (всего компонент 125 – по числу признаков после обработки), а по оси Y представлена доля объяснённой дисперсии (от 0 до 1). Видим, что первая компонента (PC1) объясняет 20% всей изменчивости данных, вторая – 15%, третья – 12,5%, далее значение начинает неуклонно снижаться, и для PC40

практически равно 0. Следовательно, компоненты после 40 номера объясняют лишь незначительную долю дисперсии данных, возможно, их не стоит включать в итоговый датасет.

Из правой части рисунка 14 можно сделать вывод, что только 16 первых компонент в сумме объясняют 95% дисперсии, следовательно, остальные компоненты могут быть отброшены.

Исходные 125 признаков (после one-hot encoding) были преобразованы в 16 главных компонент, сохранивших 95% дисперсии. Такое сжатие улучшает обучение нейронной сети для обнаружения аномалий, снижая риск переобучения и ускоряя расчёты.

График, представленный на рисунке 5.15, отображает каждое наблюдение (строку исходных данных) в виде точки в новой системе координат, где оси — это первая (PC1) и вторая (PC2) главные компоненты. Цвет точки соответствует порядковому номеру пользователя. Такой график позволяет визуально оценить, насколько сильно различаются пользователи (или группы наблюдений) по своей поведенческой структуре. Проанализируем данный график.

При визуальном анализе графике можно отметить, что кластеры точек, относящиеся к разных пользователям, лежат на одних и тех же отрезках, располагаясь на некотором удалении от них, но все же, отрезки являются «осью» кластеров. Если мысленно продолжить данные отрезки, то можно заметить, что лучи, на которых лежат данные три отрезка, пересекутся в одной точке в центре между кластерами точек.

Наличие подобной лучеобразной структуры на графике PC1–PC2 свидетельствует о том, что первые две главные компоненты в основном кодируют категориальные различия типов действий, а не непрерывные поведенческие характеристики пользователей. Это ожидаемо, так как среди исходных признаков в основном представлены бинарные индикаторы типа события.

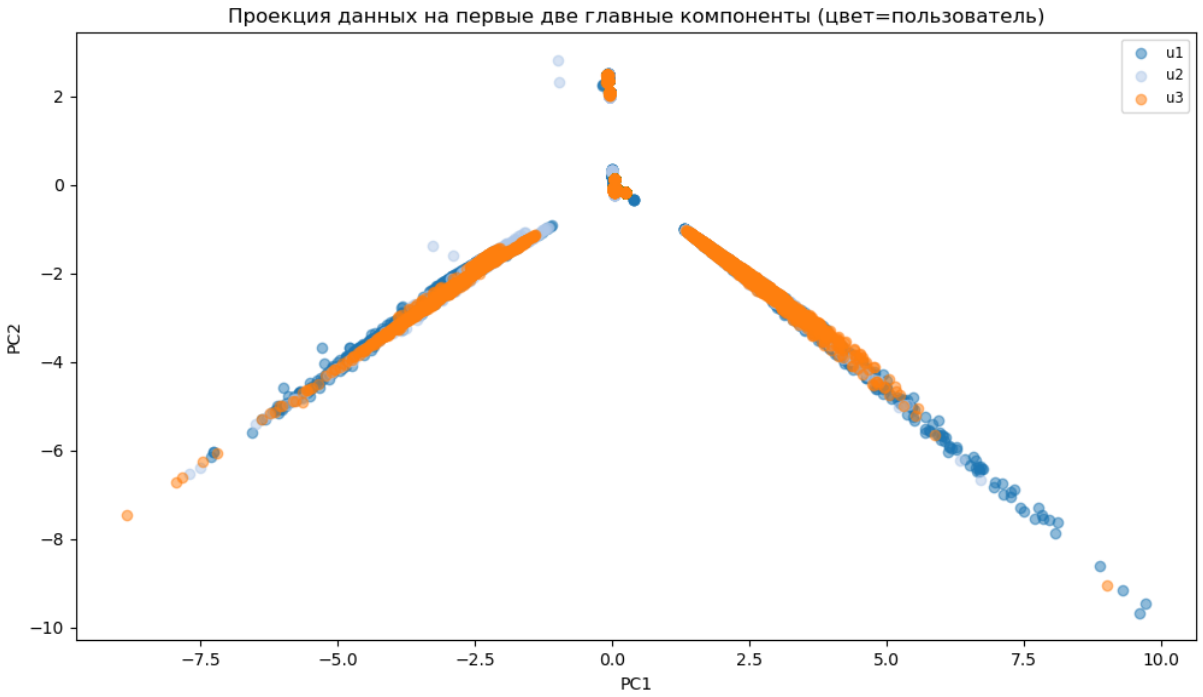


Рис. 5.15. Проекция данных на первые две компоненты, раскраска по пользователю

Для подтверждения гипотезы о том, что первые две главные компоненты в основном кодируют **категориальные различия типов действий**, а не непрерывные поведенческие характеристики пользователей, изменим раскраску точек по типу событий, а не по типу пользователей. На рисунке 5.16 представлен график с раскраской точек по типу события. Видим, что основные кластеры в данном случае соответствуют различным типам событий, нижний левый отрезок окрашен в соответствии с типом события click, правый нижний отрезок сформирован событиями типа scroll, но верхний отрезок сформирован сразу несколькими типами событий – modal_open, wheel, input, account_session_start, также частично добавлены события scroll, при этом в правом нижнем отрезке заметны включения событий типа key_special. *Стоит построить трехмерную модель для более детального визуального анализа.*

Для обнаружения аномалий в поведении пользователей такая структура данных является потенциально полезной. Аномалией в данной случае будет считаться наблюдение (точка на графике), которое не попадает ни в один из лучей, например, лежит далеко

от всех трёх линий или находится в необычной точке внутри луча, например, событие типа скроллинг с характеристиками, типичными для клика. Факт наличия кластеров в виде трех «лучей» показывает, что PCA сохранил структурную информацию, то есть сжатие до 16 компонентов прошло успешно.

Но *для идентификации пользователей такая структура данных может не подойти, или быть неполной*. Внутри каждого «луча» (например, на зеленом луче кликов) точки разных пользователей перемешаны, потому что все пользователи совершают клики. Различия между ними могут проявляться только внутри луча (например, по положению вдоль самого луча или по отклонению от оси). Если эти внутри-лучевые различия малы, то идентификация пользователей по первым двум компонентам будет невозможна. Для решения задачи идентификации можно использовать последующие компоненты (PC3, PC4 и т.д.), где могут быть закодированы индивидуальные поведенческие нюансы (время удержания клавиш, скорость скролла, точность кликов и т.п.) *уже для каждого типа событий*.

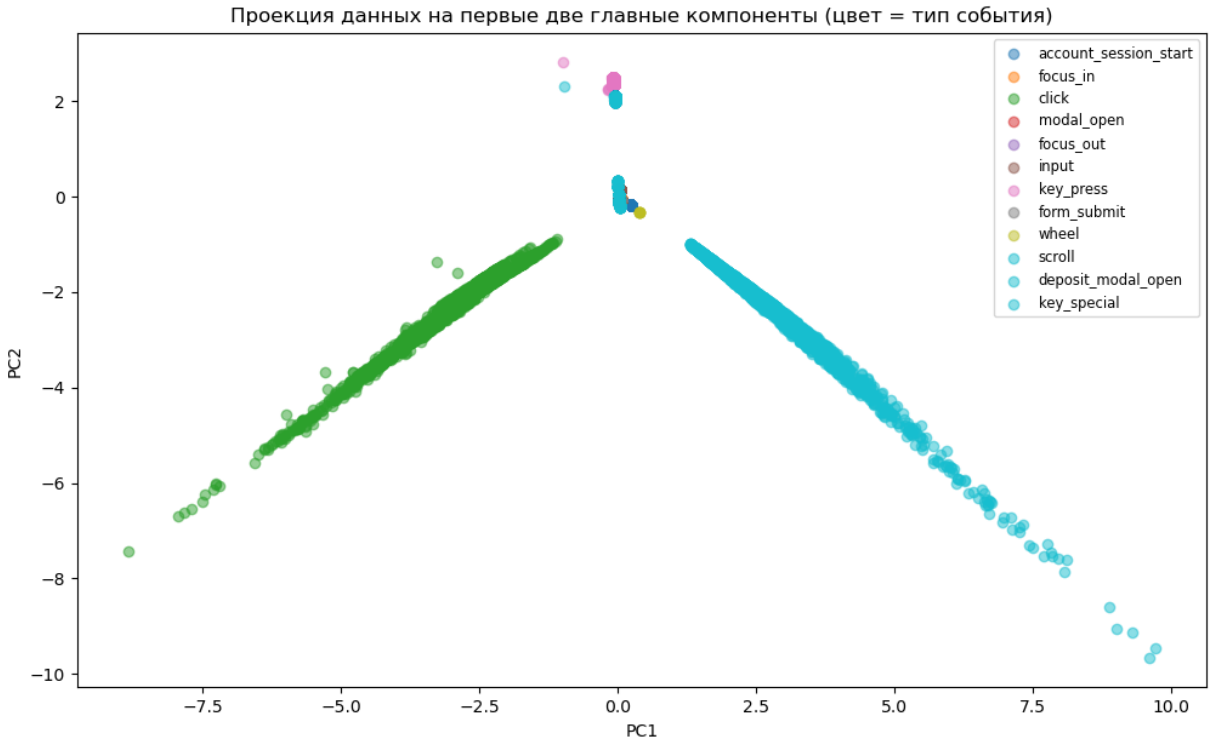


Рис. 5.16. Проекция данных на первые две компоненты, раскраска по типу события

Построим трехмерную модель для более детального анализа. На рисунке 5.17 представлен трехмерный график проекции данных на первые три компоненты с раскраской точек по типу события. Такой график позволит лучше оценить, разделаются ли разные типы событий в третьей компоненте, или же они находятся в одном кластере.

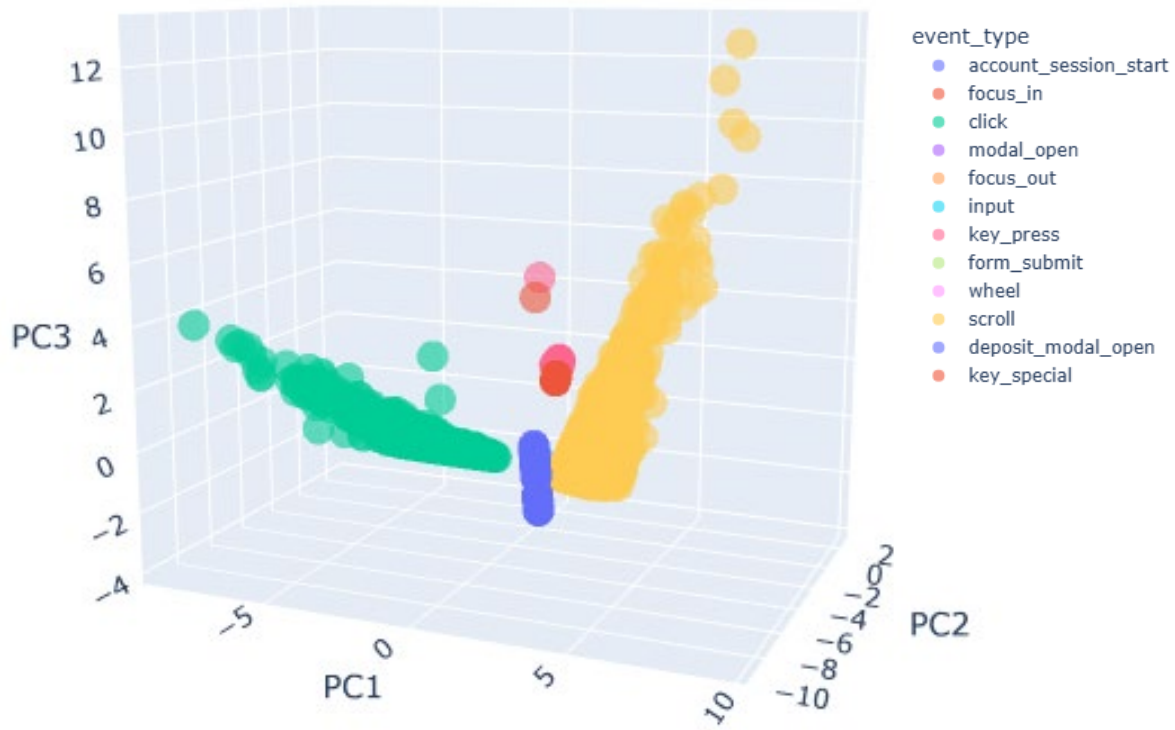


Рис. 5.17. 3D-график проекции на PC1, PC2, PC3 с раскраской по типу события

Для лучшей детализации изменим ракурс обзора на трехмерную модель, чтобы оценить, находятся ли разные точки, являющиеся разными типами событий в различных кластерах – такой график представлен на рисунке 5.18.

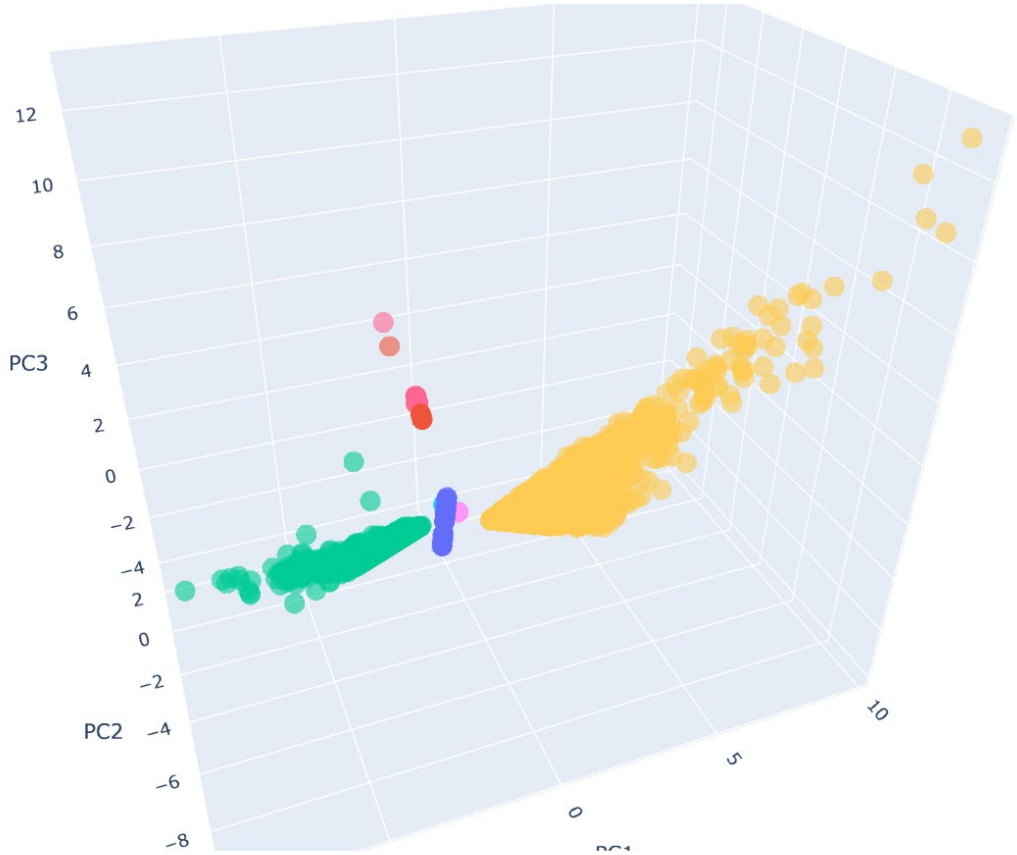


Рис. 5.18. Трёхмерная проекция данных на компоненты PC1, PC2 и PC3 с изменённого ракурса для демонстрации кластерной структуры

Из графиков, представленных на рисунках 5.17–5.18, видно, что кластеры схожих по физическому смыслу событий находятся на небольшом удалении друг от друга, например, события wheel (вращение колеса мыши, в центре между основными кластерами, точки отмечены бледно-розовым цветом) и scroll (события скроллинга, желтый кластер), а кластеры событий, имеющих существенные различия, находятся на удалении от друг друга, например, события клика (зеленый кластер) и события открытия модального окна перевода (фиолетовый кластер). Метод главных компонент позволил не просто разделить события по категориям, но и расположить их относительно друг друга в пространстве первых трёх компонент, отражая их сходство или различие. В контексте обнаружения аномалий визуальный анализ графиков, представленных на рисунках 5.17–5.18, принес положительные

результаты: наблюдаемые на графиках выбросы (например, точки событий типа click, расположенные на удалении от основного кластера кликов) могут быть потенциальными аномалиями, которые модель НС легко обнаружит. Однако основные облака (события типа click и scroll) очень плотные – внутри них модель может не найти аномалий, если только они не выходят за границы облака. Для редких событий (например, событие wheel) любое появление может быть аномалией, если они не свойственны данному пользователю.

Построение трёхмерной проекции на PC1–PC2–PC3 с раскраской по типу события подтверждает, что PCA успешно разделил данные по категориальной принадлежности, выделив пять кластеров, соответствующих разным типам взаимодействия (click, scroll, key_press, modal_open, wheel). Это указывает на то, что основные компоненты отражают структурные различия между типами событий, а не вариабельность внутри одного типа. Для задач идентификации пользователей необходимо либо перейти к анализу на более высоких компонентах (PC4–PC17), либо *строить отдельные модели для каждого типа событий*. Обнаружение аномалий (выбросов) на этом этапе возможно, так как точки, покидающие основные облака, легко идентифицируются.

Следовательно, применение метода главных компонент позволило не только уменьшить размерность датасета, но и предложить *новый подход к обучению модели нейронной сети – отдельно для каждого типа событий*.

На основе анализа результатов применения метода главных компонент были сделаны выводы, что даже в массиве данных, собранных за период, в котором у пользователей не было обнаружено случаев фрода, то есть в массиве данных, являющихся эталонным, возможно наличие аномалий. То есть, в эталонном поведении пользователя встречается небольшая доля аномалий, которые моделью НС могут быть ошибочно распознаны как нежелательная активность. Следовательно, у каждого пользователя бывают случаи аномального поведения, которое все же не приводит к финансовому ущербу и потерям, то есть, по сути, наличие аномалий (например, внезапный перевод родственнику

крупной суммы или крупная покупка, что сопровождается изменениями в моторике из-за стрессовой ситуации) *является нормальным, является частью эталонного поведенческого профиля пользователя банковской системы.* Но для каждого пользователя характерны разные аномалии, поэтому модели НС, работающие одинаково для всех пользователей, могут быть причиной ошибочных блокировок. То есть, внедрение в системы антифрода одинаковых для всех пользователей правил, которые помечают, например, крупные переводы для оплаты на маркетплейсах как нежелательную активность, может быть причиной ложных блокировок и недоверия к системам антифрода в целом.

Следовательно, модель НС должна работать индивидуально для каждого пользователя и обучаться на эталонном профиле пользователя отдельно для каждого клиента банковской системы.

Возможно, в будущем получиться соединить и описать всевозможные аномалии, возникающие в массиве эталонных данных и построить модель НС, единую для всех пользователей, которая при этом бы не вызывала ложных блокировок, но на данном этапе это не представляется возможным ввиду малого периода наблюдений.

Для получения большей информации о структуре и характере собранного массива данных вычислили матрицу нагрузок. Матрица нагрузок (loadings) в методе главных компонент (PCA) — это матрица коэффициентов линейных преобразований, которая используется для перехода из исходного пространства переменных в пространство главных компонент. Она позволяет определить, как исходные переменные связаны с новыми главными компонентами.

Анализ нагрузок позволил интерпретировать компоненты: первая разделяет скроллинг и клики, третья отражает интенсивность прокрутки, четвёртая — сумму перевода и характеристики события типа scroll, например, скорость. Четвертая компонента интересна тем, что имеет отрицательную связь с dwell_time и is_scroll_event. Это может означать, что при больших суммах пользователь меньше скроллит и быстрее нажимает клавиши. Четвертая компонента может быть полезна для детекции

аномалий в платежах. Пятая компонента зависит только от flight_time, остальные признаки незначительны. Это подтверждает вывод, сделанный на этапе анализа данных, параметр flight_time ортогонален остальным и вносит независимый вклад в дисперсию. Остальные компоненты не дают сравнимого вклада в общею дисперсию, но могут быть интересны как отражение второстепенных поведенческих нюансов, как например, PC2.

В результате анализа нагрузок было выявлено, что из 125 признаков только 54 значимо влияют на компоненты. Самыми информативными признаками являются: dwell_time, flight_time, trajectory_distance, scroll_speed, scroll_distance, scroll_position, total_scroll_distance, coordinates_x, coordinates_y, operation_amount, is_scroll_event, is_click_or_key, категориальные признаки, такие как типы событий, тип ввода, детектор нажатия специальной клавиши. Избыточные признаки, которым соответствуют нулевые или близкие к нулю нагрузки на всех компонентах: wheel_delta_x, wheel_delta_y, is_mobile.

В результате анализа матрицы нагрузок сделаны следующие выводы: в результате применения метода главных компонент устранена мультиколлинеарность в массиве собранных данных, все компоненты ортогональны; flight_time является важной поведенческой характеристикой, не связанной с другими характеристиками моторных паттернов пользователя.

5.7.2. Isolation Forest

Isolation Forest (iForest) — это алгоритм машинного обучения без учителя, предназначенный для выявления аномалий (выбросов) в многомерных данных. В отличие от традиционных методов, которые описывают «нормальное» поведение и затем ищут отклонения, iForest напрямую изолирует аномалии. Концептуально алгоритм опирается на тот факт, что аномальные наблюдения, например, фрод, характеризуются малым количеством признаков, по которым их можно отделить от основной массы данных. iForest строит ансамбль бинарных деревьев решений (isolation trees). Для каждого дерева случайным образом выбирается признак,

случайным образом выбирается пороговое значение в диапазоне этого признака, данные разделяются на две части: объекты со значением признака ниже порога и выше порога. Процесс повторяется до тех пор, пока каждое наблюдение не окажется в отдельном узле или не будет достигнута заданная глубина дерева.

Здесь данные представляют собой логи действий пользователей банковского приложения: клавиатурный почерк, движения мыши, прокрутку, типы событий, временные характеристики и финансовые параметры. Основная цель применения — создать систему обнаружения аномального поведения в эталонном профиле пользователя, чтобы получить информацию о характере «легитимных» аномалий. Такая информация позволит оценить характер аномалий, возникающих в массиве данных, собранных в течение периода, в котором не было случаев фрода, что поможет выбрать подход к построению архитектуры модели нейронной сети и ее обучению.

Применение метода главных компонентов (PCA) перед iForest не является обязательным, но в данном исследовании благодаря этому получаем следующие преимущества: уменьшение шума (главные компоненты отбрасывают малые собственные значения); устранение мультиколлинеарности, что улучшает случайный выбор признаков в деревьях; сокращение вычислительных затрат (16 вместо 125 признаков).

В ходе экспериментов iForest выделил ≈4,1% наблюдений как аномальные. Среди них преобладают события scroll и click (около 87%), а также deposit_modal_open (12,6%). Распределение по пользователям показало, что один пользователь даёт 56% от общего числа наблюдений. Это указывает на нестандартное поведение (данный пользователь действительно отличается повышенной скоростью действий), а также на необходимость персонализации модели нейронной сети, решающей задачу антифрода.

Применение Isolation Forest к 16 главным компонентам позволило выявить 4% наблюдений, отклоняющихся от типового паттерна. Обнаружено, что один пользователь генерирует непропорционально много аномалий, в основном за счёт скроллов

и переводов по номеру телефона. Метод показал свою эффективность как первый этап в пайплайне обнаружения аномалий, но для практического применения в антифроде необходима дополнительная калибровка с учётом бизнес-правил и постепенное накопление размеченных инцидентов, но наилучшим вариантом будет применение новых моделей нейронных сетей (например, KAN) с индивидуальным обучением для каждого пользователя, так как доказано, что у каждого пользователя существуют индивидуальные аномалии в поведенческих данных, не приводящие к финансовому ущербу.

5.8. ЗАКЛЮЧЕНИЕ

Так, мы видим необходимость применения психологического фактора для решения задачи антифрода.

С помощью разработанного приложения-симулятора онлайн-банка был собран массив данных о поведении пользователей. С помощью разведочного анализа данных получены важные сведения о характере полученных сведений. Полученный массив данных был предобработан: анализ нулевых значений, обработка пропусков, удаление незначащих признаков, one-hot encoding категориальных признаков (увеличили количество признаков с 41 до 123). В ходе разведочного анализа данных выяснили, что для собранного массива данных характерны линейные, нелинейные и другие типы связей между парами признаков.

После этого был применен метод главных компонентов, его применение позволило подтвердить сведения, полученные в ходе разведочного анализа данных: признак flight_time является потенциально самым ценным поведенческим признаком, его значения не зависят от значений остальных признаков. С помощью метода главных компонентов сократили количество признаков в массиве данных с 125 до 16 компонент, что в перспективе уменьшит время обучения модели нейронной сети. В ходе анализа матрицы нагрузок выявили самые значимые признаки, их 54. Также метод главных компонентов позволил выдвинуть гипотезу о том, что

необходимо разделение моделей НС, решающих задачу антифрода, по пользователям и типам событий.

В ходе исследования собранный массив «сырых» данных был проанализирован, обработан, из него удалось получить несколько «промышленных» датасетов, которые могут быть применены для обучения нейронных сетей.

Применение Isolation Forest позволило доказать, что аномалии, встречающиеся в эталонных поведенческих данных, характерны для всех пользователей, но при этом их количество и характеристики индивидуальны для всех пользователей. Следовательно, подтверждается необходимость обучения моделей НС на данных каждого пользователя отдельно.

Перспектива проведенного исследования заключается в том, что полученные результаты (скрипты для обработки сырых данных и выводы, сделанные в ходе анализа) могут обеспечить более эффективное решение задачи антифрода с помощью предложенного алгоритма обработки сырых данных. Доказано, что данные, получаемые в ходе работы пользователя с банковским приложением, необходимо обрабатывать индивидуальными моделями НС, обученными для каждого пользователя отдельно. Подобная система антифрода может быть внедрена в банковские приложения следующим образом: организация сбора эталонных поведенческих данных за ограниченный период времени для каждого пользователя, проведение психологического тестирования пользователя для оценки уязвимости к мошенническим схемам, обучение модели НС отдельно для каждого пользователя, встраивание обученной модели в работу банковского приложения для анализа каждой активной сессии работы пользователя, что позволит сократить время анализа в сравнении с клиент-серверной моделью работы НС, так как не будет затрачиваться время на доставку данных. Подобная структура накладывает определенные ограничение на архитектуру модели НС – ее работа должна быть оптимальна для запуска на телефоне или ином устройстве пользователя. Нейронные сети и способы их создания постоянно совершенствуются, возможно, в продолжении исследования будет

представлена новый подход к построению модели НС для оптимальной работы на телефонах пользователей.

Ученые, занимающиеся исследованиями в поведенческой экономике, решают задачи синтеза данных, которые могли бы заменить реальные. Чтобы синтезировать данные, необходимо понимать их структуру и внутренние взаимосвязи, которые в полной мере отражает данное исследование. Результаты данного исследования можно применить как исходные данные для построения модели синтеза данных, или даже для моделирования поведения пользователя в условиях, когда сбор реальных данных может быть затруднен.

Применение сведений о психологических характеристиках пользователей в следующих этапах исследования необходимо, так как это позволит повысить точность работы модели и уменьшить количество ложных срабатываний, являющихся проблемой большинства антифрод систем.

СПИСОК ЛИТЕРАТУРЫ

1. *Конявский В. А., Конявская-Счастная С. В., GigaChat.* Искусственный интеллект и защита информации // Защита информации. Инсайд. М., 2024. № 6. С. 4–11.

2. *Мински М.* Фреймы для представления знаний. М.: Энергия., 1979. – 151 с.

3. *Гренандер У.* Лекции по теории образов. Т. I. М.: МИР, 1979.

4. Конявский В. А., Гадасин В. А. Основы понимания феномена электронного обмена информацией. Минск: Серия «Библиотека журнала "УЗИ"», 2004. – 327 с.

5. Кузнецов Н. А. Информационное взаимодействие в технических и живых системах [Электронный ресурс] // Информационные процессы. 2001. Том 1. № 1. С. 1–9. Режим доступа: http://www.jip.ru/2001/1-1-2201.htm (дата обращения: 30.12.2021).

6. Стрельцов А. А. Правовое обеспечения информационной безопасности России: теоретические и методологические основы. Минск, 2005.

7. *Конявский В. А.* Управление защитой информации на базе СЗИ НСД «Аккорд». Москва: «Радио и связь», 1999. – 325 с.

8. *Конявский В. А., Хованов В. Н.* Страхование информационных рисков и обеспечение информационной безопасности // Управление защитой информации. Мн., 2000. Том 4. № 1.

9. *Акаткин Ю. М., Ясиновская Е. Д.* Цифровая трансформация государственного управления. Датацентричность и семантическая интероперабельность. М., 2022. – 912 с.

10. *Конявский В. А., Конявская С. В.* Доверенные информационные технологии: от архитектуры к системам и средствам. М.: URSS. 2021. – 264 с. 2-е изд.

11. *Конявский В. А., Медведев В. В., Росс Г. В.* Защищенные информационные технологии в цифровой экономике // Вопросы защиты информации. 2022. № 2(137). С. 34—44.

12. *Лихтенштейн В. Е., Конявский В. А., Росс Г. В., Лось В. П.* Мультиагентные системы: самоорганизация и развитие. Москва: Финансы и статистика, 2022. – 264 с.

13. *Конявский В. А.* Идентификация и применение ЭЦП в компьютерных системах информационного общества // Безопасность информационных технологий. М., 2010. № 3. С. 6–13.

14. *Конявская-Счастная С. В.* Начала технической защиты информации. — М.: Типография «Вишнёвый пирог», 2025.

15. *Комаров А.* Современные методы аутентификации: токен и это все о нем…! // T-Comm. 2008. № 6. С. 13–16.

16. *Конявский В. А.* Новая биометрия. Можно ли в новой экономике применять старые методы? // Information Security\Информационная безопасность. 2018. № 4. С. 34–36.

17. *Конявский В. А.* Интерактивный способ биометрической аутентификации пользователя. Патент на изобретение № 267024.10.2018, бюл. № 30.

18. *Shannon C. E.* A mathematical theory of communication // Bell System Technical Journal. 1948. Vol. 27, № 3. P. 379–423; № 4. P. 623–656.

19. *Lehmann E. L., Casella G.* Theory of Point Estimation. 2nd ed. New York: Springer, 1998. – 589 p.

20. *Lehmann E. L., Romano J. P.* Testing Statistical Hypotheses. 3rd ed. New York: Springer, 2005. – 786 p.

21. *Колмогоров А. Н.* Теория информации и теория алгоритмов. М.: Наука, 1987. – 304 с.

22. *Cover T. M., Thomas J. A.* Elements of Information Theory. 2nd ed. Hoboken: John Wiley & Sons, 2006. – 776 p.

23. *Batini C., Scannapieco M.* Data Quality: Concepts, Methodologies and Techniques. Berlin; Heidelberg: Springer, 2006. – 262 p.

24. *Moreau L., Missier P.* PROV-DM: The PROV Data Model. W3C Recommendation, 30 April 2013. URL: https://www.w3.org/TR/prov-dm/

25. *Jøsang A.* An algebra for assessing trust in certification chains // Proceedings of the Network and Distributed Systems Security Symposium (NDSS '99). 1999.

26. *Jøsang A., Ismail R., Boyd C.* A survey of trust and reputation systems for online service provision // Decision Support Systems. 2007. Vol. 43. № 2. P. 618–644.

27. *Pipino L. L., Lee Y. W., Wang R. Y.* Data quality assessment // Communications of the ACM. 2002. Vol. 45, № 4. P. 211–218.

28. *Sambasivan N., Akhtar A., Anand A., Aroyo L., Dell N., Vaden C., Paritosh P., Aroyo L. M.* "Everyone wants to do the model work, not the data

work": Data cascades in high-stakes AI // Proceedings of the CHI Conference on Human Factors in Computing Systems. 2021. P. 1–15.

29. *Ribeiro M. T., Singh S., Guestrin C.* "Why should I trust you?": Explaining the predictions of any classifier // Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016. P. 1135–1144.

30. *Madry A., Makelov A., Schmidt L., Tsipras D., Vladu A.* Towards deep learning models resistant to adversarial attacks // Proceedings of the International Conference on Learning Representations (ICLR 2018). 2018.

31. *Ситников А. Ю.* Изменение доверенности наборов данных при их комплексировании // Актуальные вопросы защиты информации. 2025. Вып. 4. С. 51.

32. *Mitchell M., Wu S., Zaldivar A., Barnes P., Vasserman L., Hutchinson B., Spitzer E., Raji I. D., Gebru T.* Model cards for model reporting // Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT* '19). 2019. P. 220–229.

33. *Gebru T., Morgenstern J., Vecchione B., Vaughan J. W., Wallach H., Daumé H., Crawford K.* Datasheets for datasets // Communications of the ACM. 2021. Vol. 64. № 12. P. 86–92.

34. *Guo C., Pleiss G., Sun Y., Weinberger K. Q.* On calibration of modern neural networks // Proceedings of the 34th International Conference on Machine Learning (ICML 2017). 2017. P. 1321–1330.

35. *Hendrycks D., Dietterich T.* Benchmarking neural network robustness to common corruptions and perturbations // Proceedings of the International Conference on Learning Representations (ICLR 2019). 2019.

36. *Lakshminarayanan B., Pritzel A., Blundell C.* Simple and scalable predictive uncertainty estimation using deep ensembles // Advances in Neural Information Processing Systems. 2017. Vol. 30. P. 6402–6413.

37. *Hendrycks D., Gimpel K.* A baseline for detecting misclassified and out-of-distribution examples in neural networks // Proceedings of the International Conference on Learning Representations (ICLR 2017). 2017.

38. *Wong A., Wang X. Y., Hryniowski A.* How much can we really trust you? Towards simple, interpretable trust quantification metrics for deep neural networks // arXiv preprint arXiv:2009.05835. 2020.

39. *Dai C., Lin D., Bertino E., Kantarcioglu M.* An approach to evaluate data trustworthiness based on data provenance // Secure Data Management. Lecture Notes in Computer Science. Berlin; Heidelberg: Springer, 2008. Vol. 5159. P. 82–98.

40. *Pina D., Kohwalter T., Murta L., Mattoso M.* DLProv: a suite of provenance services for deep learning workflow analyses // Future Generation

Computer Systems. 2024; Wang W., Lu W., Li B. Model provenance via model DNA // arXiv preprint arXiv:2308.02121. 2023.

41. *Mirzadeh S. I., Farajtabar M., Li A., Levine N., Matsukawa A., Ghasemzadeh H.* Improved knowledge distillation via teacher assistant // Proceedings of the AAAI Conference on Artificial Intelligence. 2020. Vol. 34. № 04. P. 5191–5198.

42. *Конявский В. А, Конявская-Счастная С. В., Росс Г. В. Райгородский А. М., Тренин С. А., Леонидов А. В., Васильева Е. Е., Васильев С. Б., Коновалихин М. Ю.* Технология «слепой» обработки привлекаемых данных в системах машинного обучения// Вопросы защиты информации. М., 2024. № 2. С. 17–32.

43. *Dinur I., Nissim K.* Revealing Information While Preserving Privacy // Proceedings of the ACM Symposium on Principles of Database Systems (PODS), 2003.

44. *Dwork C., Roth A.* The Algorithmic Foundations of Differential Privacy // Foundations and Trends in Theoretical Computer Science, 2014.

45. *Fredrikson M., Jha S., Ristenpart T.* Model inversion attacks that exploit confidence information and basic countermeasures // Proceedings of the 22nd ACM Conference on Computer and Communications Security (CCS), 2015. С. 1322–1333.

46. *Galmanov P.* Inverse-problem perspective on indirect information leakage in AI systems // Robot Autom Eng J. Vol. 7. Iss. 1. 2025. Article 555701.

47. *Галманов П. А.* Подход регуляризованных обратных задач к оценке риска возникновения косвенной утечки в анклавах данных // Вопросы защиты информации. 2025. Вып. 4. С. 25–30.

48. *Galmanov P., Konyavskiy V.* A Regularized Inverse-Problem Framework for Assessing Indirect Leakage Risk in Privacy-Preserving Data Enclaves // Frontiers in Artificial Intelligence and Applications. 2026. P. 116–122.

49. *Галманов П. А.* Вероятность ранней косвенной утечки в анклавах данных: от порога обнаруживаемости к распределению времени до компрометации // Вопросы защиты информации. 2026. Вып. 1. С. 40–48.

50. Hadamard J. Lectures on Cauchy's Problem in Linear Partial Differential Equations. Yale University Press, 1923.

51. Kirsch A. An Introduction to the Mathematical Theory of Inverse Problems, 3-я изд. Springer, 2021.

52. Hansen P. C. Discrete Inverse Problems: Insight and Algorithms. Philadelphia: SIAM, 2010.

53. *Тихонов А. Н., Арсенин В. Я.* Методы решения некорректных задач. М.: Наука, 1979.

54. Морозов В. А. Регулярные методы решения некорректно поставленных задач. М.: Наука, 1987.

55. *Wahba G.* Spline Models for Observational Data. Philadelphia: SIAM, 1990.

56. *Schölkopf B., Smola A. J.* Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. Cambridge, MA: MIT Press, 2002.

57. *Landweber L.* An iteration formula for Fredholm integral equations of the first kind // American Journal of Mathematics. 1951. Vol. 73. Iss. 3. P. 615–624.

58. *Yao Y., Rosasco L., Caponnetto A.* On early stopping in gradient descent learning // *Constructive Approximation*. 2007. Vol. 26. Iss. 2. P. 289–315.

59. *Hastie T., Tibshirani R., Friedman J.* The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2-е изд-е. New York: Springer, 2009.

60. *Good P.* Permutation, Parametric and Bootstrap Tests of Hypotheses, 3-е изд-е. New York: Springer, 2005.

61. *Конявский В. А.* Доверенные системы как средство противодействия киберугрозам. Базовые понятия // Информационная безопасность. 2016. № 3. С. 40–41.

62. *Schroeder de Witt C.* Open Challenges in Multi-Agent Security: Towards Secure Systems of Interacting AI Agents. — arXiv:2505.02077, 2025.

63. Щербаков А. Ю. Хрестоматия специалиста по современной информационной безопасности. Palmarium Academic Publishing, 2016. Т. 1. — 265 с.

64. *Кузнецов А. В.* Краткий обзор многоагентных моделей // Управление большими системами. 2010. № 30. С. 327–350.

65. Wang L. et al. A Survey on Large Language Model based Autonomous Agents. — arXiv:2308.11432, 2023.

66. *Li X. et al.* A Survey on LLM-based Multi-Agent Systems: Workflow, Infrastructure, and Challenges // Vicinagearth. 2024.

67. Wang Y. et al. Trustworthy Edge Intelligence: A Survey // IEEE Communications Surveys & Tutorials. 2024.

68. *McMahan B. et al.* Communication-Efficient Learning of Deep Networks from Decentralized Data // AISTATS 2017. P. 1273–1282.

69. *Kairouz P. et al.* Advances and Open Problems in Federated Learning // Foundations and Trends in Machine Learning. 2021. Vol. 14. № 1–2. P. 1–210.

70. *Zhang H. et al.* A Survey of Trustworthy Federated Learning: Issues, Solutions, and Challenges // ACM Computing Surveys. 2024.

71. *Tanenbaum A. S., Van Steen M.* Distributed Systems: Principles and Paradigms. 2nd ed. Upper Saddle River: Pearson, 2007.

72. Peer to Peer: Harnessing the Power of Disruptive Technologies / ed. by A. Oram. Sebastopol: O'Reilly, 2001.

73. *Stoica I., Morris R., Karger D., Kaashoek M. F., Balakrishnan H.* Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications // SIGCOMM 2001. New York: ACM, 2001. P. 149–160.

74. *Maymounkov P., Mazieres D.* Kademlia: A Peer-to-Peer Information System Based on the XOR Metric // IPTPS 2002. Berlin: Springer, 2002. P. 53–65.

75. Nakamoto S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008.

76. Readings in Distributed Artificial Intelligence / ed. by A. Bond, L. Gasser. San Mateo: Morgan Kaufmann, 1988.

77. Rao A. S., Georgeff M. P. BDI Agents: From Theory to Practice // Proceedings of the First International Conference on Multi-Agent Systems. AAAI Press, 1995. P. 312–319.

78. *Olfati-Saber R., Fax J. A., Murray R. M.* Consensus and Cooperation in Networked Multi-Agent Systems // Proceedings of the IEEE. 2007. Vol. 95, № 1. P. 215–233.

79. *Satyanarayanan M.* The Emergence of Edge Computing // Computer. 2017. Vol. 50, № 1. P. 30–39.

80. *Bonomi F., Milito R., Zhu J., Addepalli S.* Fog Computing and Its Role in the Internet of Things // MCC'12. 2012. P. 13–16.

81. *Kamvar S. D., Schlosser M. T., Garcia-Molina H.* The EigenTrust Algorithm for Reputation Management in P2P Networks // WWW 2003. New York: ACM, 2003. P. 640–651.

82. Xiong L., Liu L. PeerTrust: Supporting Reputation-Based Trust for Peer-to-Peer Electronic Communities // IEEE Transactions on Knowledge and Data Engineering. 2004. Vol. 16. № 7. P. 843–857.

83. *Sabater J., Sierra C.* REGRET: A Reputation Model for Gregarious Societies // Proceedings of AGENTS 2001. New York: ACM, 2001. P. 194–195.

84. Huynh T. D., Jennings N. R., Shadbolt N. R. An Integrated Trust and Reputation Model for Open Multi-Agent Systems // Autonomous Agents and Multi-Agent Systems. 2006. Vol. 13. P. 119–154.

85. *Tajeddine A., Kayssi A., Chehab A., Artail H.* PATROL-F: A Comprehensive Reputation-Based Trust Model with Fuzzy Subsystems // Autonomic and Trusted Computing. Berlin: Springer, 2006. P. 205–216.

86. *Wang Y., Vassileva J.* Bayesian Network-Based Trust Model // Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems. 2003.

87. Rose S., Borchert O., Mitchell S., Connelly S. Zero Trust Architecture (NIST SP 800-207). Gaithersburg: NIST, 2020.

88. *Manna A., Sengupta A., Mazumdar C.* A Survey of Trust Models for Enterprise Information Systems // International Conference on Computational Modeling and Security. 2016. P. 527–534.

89. Cost of insider risks: глобальный отчет, 2023 / Independently conducted by Ponemon Institute and DTEX Systems Inc. 1-е изд-е. Traverse City: Ponemon Institute, 2023.

90. *Белоглазов Д. А.* Особенности нейросетевых решений, достоинства и недостатки, перспективы применения // Известия ЮФУ. Технические науки. 2008. № 7. С. 105–110.

91. *Резников Р.* Искусственный интеллект в киберзащите [Электронный ресурс]. Режим доступа: https://ptsecurity.com/ru-ru/research/analytics/iskusstvennyi-intellekt-v-kiberzaschite/#id12 (дата обращения: 31.05.2025).

92. *Воробьев И. А.* Методы машинного обучения в задаче оценки риска мошенничества в автостраховании // Известия Саратовского университета. Новая серия. Серия: Математика. Механика. Информатика. 2024. Т. 24, № 4. Режим доступа: https://cyberleninka.ru/article/n/metody-mashinnogo-obucheniya-v-zadache-otsenki-riska-moshennichestva-v-avtostrahovanii (дата обращения: 17.04.2026).

93. *Федюнина А. П.* Выявление характерологических признаков и составление психологического портрета возможного нарушителя и лояльного сотрудника в сфере информационной безопасности // Вестник Астраханского государственного технического университета. 2007. № 4(39). С. 231–236.

94. *Магомедов Ш. Г.* Модели и методы адаптивного риск-ориентированного управления доступом в распределенных информационных системах [Электронный ресурс]: дис. … д-ра техн. наук: 2.3.6. Екатеринбург, 2025. Режим доступа: https://www.dissercat.com/content/modeli-i-metody-adaptivnogo-risk-orientirovannogo-upravleniya-dostupom-v-raspredelennykh-inf (дата обращения: 11.05.2026).

95. *Корниенко С. В., Пантюхина А. В.* Методика выявления потенциальных внутренних нарушителей информационной безопасности // Интеллектуальные технологии на транспорте. 2023. № 2(34). С. 50–57.

96. *Батюкова Л. Е., Карлова Т. В.* Методика обеспечения безопасности транзакций на основе использования антифрод-системы // Автоматизация и моделирование в проектировании и управлении. 2024. № 1 (23). [Электронный ресурс]. Режим доступа:

https://cyberleninka.ru/article/n/metodika-obespecheniya-bezopasnosti-tranzaktsiy-na-osnove-ispolzovaniya-antifrod-sistemy (дата обращения: 18.04.2026).

97. *Islam M. M., Zerine I., Rahman M. A., Islam M. S., Ahmed M. Y.* AI-Driven Fraud Detection in Financial Transactions – Using Machine Learning and Deep Learning to Detect Anomalies and Fraudulent Activities in Banking and E-Commerce Transactions // International Journal of Communication Networks and Information Security (IJCNIS). 2024. Vol. 1. No. 5.

98. *Воробьев И. А.* Исследования по разработке методов противодействия мошенничеству в финансовых организациях с применением машинного обучения [Электронный ресурс]: дис. … канд. техн. наук: 2.3.6. Москва, 2024. Режим доступа: https://www.dissercat.com/content/issledovaniya-po-razrabotke-metodov-protivodeistviya-moshennichestvu-v-finansovykh-organizat (дата обращения: 11.05.2026).

99. *Саенко И. Б., Котенко И. В., Аль-Барри М. Х.* Применение искусственных нейронных сетей для выявления аномального поведения пользователей центров обработки данных // Вопросы кибербезопасности. 2022. № 2(48). С. 87–97.

100. *Есипов Д. А., Асланова Н., Шабала Е. Е., Щетинин Д. С., Попов И. Ю.* Метод обнаружения инцидентов информационной безопасности по аномалиям в биометрических поведенческих чертах пользователя // Научно-технический вестник информационных технологий, механики и оптики. 2022. Т. 22. № 4. С. 760–768; [Электронный ресурс]. Режим доступа: https://cyberleninka.ru/article/n/metod-obnaruzheniya-intsidentov-informatsionnoy-bezopasnosti-po-anomaliyam-v-biometricheskih-povedencheskih-chertah-polzovatelya (дата обращения: 28.04.2026).

101. *Ннамоко Н., Барроуклаф Д., Липтротт М., Корконцелос И.* Набор данных поведенческой биометрии для идентификации и аутентификации пользователей [Текст] = [A behaviour biometrics dataset for user identification and authentication] // Data in Brief. 2022. Т. 45. Ст. 108728.

102. *Vaidya A. T., Mhatre J. D.* Behavioural Biometrics for Continuous Authentication in Cybersecurity Systems // International Journal of Computer Technology and Electronics Communication. 2025. Vol. 8. Iss. 4. P. 11007–11012.

103. *Modic, D.* Willing to be scammed: how self-control impacts Internet scam compliance: дис. … PhD. Exeter: University of Exeter, 2012.

104. *Guerra A., Taylor K.* Scam Susceptibility: Determining the Dominant Factor for an Adolescent's Decision-making // Journal of Student Research. 2021. Vol. 10. No. 4.

105. *Еременко Т. В.* Потенциал «треугольника мошенничества» в предупреждении академической нечестности студентов // Непрерывное образование: XXI век. 2024. Вып. 3 (47). [Электронный ресурс]. Режим доступа: https://cyberleninka.ru/article/n/potentsial-treugolnika-moshennichestva-v-preduprezhdenii-akademicheskoy-nechestnosti-studentov (дата обращения: 11.05.2026).

106.*Стародубов М. И., Боршевников А. Е., Селин Н. А.* Генерация синтетических данных для систем интеллектуального анализа в задаче обнаружения вредоносного программного обеспечения // Вопросы кибербезопасности. 2025. № 2 (66). [Электронный ресурс]. Режим доступа: https://cyberleninka.ru/article/n/generatsiya-sinteticheskih-dannyh-dlya-sistem-intellektualnogo-analiza-v-zadache-obnaruzheniya-vredonosnogo-programmnogo (дата обращения: 27.11.2025).

107.*Фомина Е. Е.* Обзор методов и программного обеспечения для восстановления пропущенных значений в массивах социологических данных // Гуманитарный вестник. 2019. № 4 (78). [Электронный ресурс]. Режим доступа: https://cyberleninka.ru/article/n/obzor-metodov-i-programmnogo-obespecheniya-dlya-vosstanovleniya-propuschennyh-znacheniy-v-massivah-sotsiologicheskih-dannyh (дата обращения: 07.05.2026).

Для ссылок:

Конявская-Счастная С. В., Ситников А. Ю., Галманов П. А., Буянов С. С., Ищанова С. Г. Информационные процессы в системах доверенного ИИ. М.: «Вишневый пирог», 2026. – 302 с.

Подписано в печать: 20.04.2026
Формат 60×90/16. Печать цифровая.
Бумага офсетная №1.
Усл. печатных л.: 19
Усл. авт. л.: 12,3
Тираж 500 экз.
Тип. заказ № 1326288

ISBN 978-5-6056038-2-5

Типография «Вишневый пирог»
115114, Москва, 2-й Кожевнический пер., 12
Тел: +7 495 994 49 94