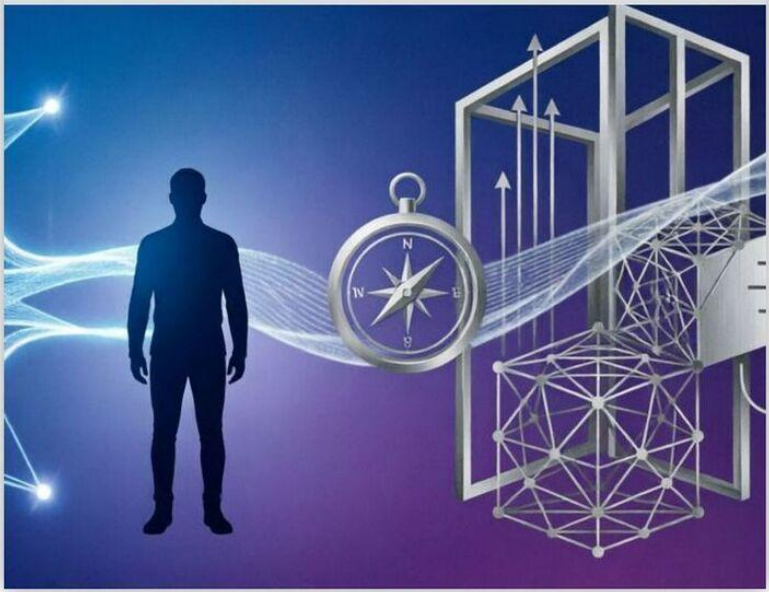
18+

Сергей Сергеевич Дегтев

Когнитивный архитектор

Как управлять ИИ-проектами, когда плана нет



Сергей Дегтев

Когнитивный архитектор. Как управлять ИИ-проектами, когда плана нет

https://litres.ru/74274843
ISBN 9785007067492

Аннотация

ИИ-проекты — это разведка, а не стройка. План бесполезен, сроки гадальны, а модель непредсказуема. Ваша новая роль — когнитивный архитектор. Не диспетчер задач, а проектировщик мышления команды. Семь компетенций, живые артефакты, практики управления диалогом с ИИ и доверием.

Содержание

Когнитивный архитектор Как управлять ИИ-проектами, когда плана нет

Сергей Сергеевич Дегтев

ISBN 978-5-0070-6749-2
Создано в интеллектуальной издательской системе Ridero

ЮРИДИЧЕСКИЙ ДИСКЛЕЙМЕР

Материалы данной книги носят исключительно информационный и образовательный характер. Текст предоставлен в состоянии «как есть» и не является юридической, финансовой, технической, управленческой или профессиональной консультацией. Автор не гарантирует абсолютную точность, полноту или неизменную актуальность сведений, не несёт ответственности за решения, принятые читателем на основе данного текста, и не отвечает за прямые или косвенные последствия их применения.

Упоминание конкретных технологий, платформ, фреймворков, методологий или практик не означает их одобрения, сертификации или рекомендации к обязательному использованию. Рынок искусственного интеллекта, регуляторная среда, стандарты информационной безопасности и корпоративные политики меняются стремительно: подходы, описанные в книге, могут требовать адаптации под ваш контекст, юрисдикцию, отраслевые стандарты и уровень зрелости процессов.

Автор не гарантирует достижения каких-либо результатов от применения описанных методов и не несёт ответственности за любые прямые или косвенные убытки, возникшие в

результате их использования.

Результаты внедрения любых практик зависят от качества данных, архитектуры решений, компетенций команды, внутренней культуры организации и внешних факторов. Перед применением описанных методов, инструментов или изменений в рабочих процессах рекомендуется проконсультироваться с профильными специалистами (юристами, специалистами по информационной безопасности, архитекторами, руководителями направлений, HR-партнёрами).

Читатель самостоятельно оценивает риски, проверяет соответствие действий применимому законодательству, отраслевым нормам и внутренним регламентам своей организации. Все примеры, кейсы и сценарии приведены в образовательных целях и могут отличаться от реальной практики в зависимости от специфики проектов.

О СОАВТОРСТВЕ С ИИ

Эта книга написана в диалоге между человеком и искусственным интеллектом. Я выступал в роли, о которой идёт речь на этих страницах — когнитивного архитектора. Моя задача заключалась не в механическом наборе текста, а в проектировании смыслов: формулировании гипотез, выстраивании структуры, отборе метафор, валидации практик, редактировании и синтезе опыта. ИИ выступал как инструмент расширения мышления — генерировал варианты, систематизировал знания, проверял логику и помогал находить формулировки.

Каждый раздел проходил через человеческий фильтр: я принимал решения, что оставить, что изменить, а что отбросить. Текст не сгенерирован автоматически — он стал результатом управляемой итеративной работы, где ИИ усиливал, а не заменял автора. Это сознательный эксперимент, который подтверждает главную мысль книги: в эпоху неопределённости ценность создаёт не тот, кто «знает всё», а тот, кто умеет выстраивать диалог между человеческой интуицией и машинными возможностями.

Я указываю это открыто, потому что прозрачность — часть профессиональной этики новой роли. Если вы читаете эти строки, значит, подход работает. И я благодарен ИИ за то, что он стал тем самым «инструментом», из взаимодей-

ствия с которым вместе с опытом, практикой и вниманием к деталям родилась эта книга.

ВВЕДЕНИЕ. Почему PM больше не диспетчер

Вы когда-нибудь строили дом по чертежу? Там всё известно заранее: фундамент, стены, крыша, сроки, бюджет. Вы можете нанять бригаду, раздать задачи и следить за графиком. Это классический проектный менеджмент. PMBOK, Waterfall, даже Agile — все они, по сути, учат управлять *исполнением* при условии, что *что* делать более-менее понятно.

Теперь представьте, что вас отправляют в джунгли без карты. Сказано: «Найди клад». Но клад — это ИИ-модель, которая должна решать бизнес-задачу. Вы не знаете, какой алгоритм сработает, сколько данных понадобится, дадут ли они нужное качество, не начнёт ли модель галлюцинировать в проде. Это не стройка. Это разведка.

И вот тут классический PM ломается. Потому что он привык к определённости. А в ИИ-проекте определённости нет. Вы не можете «оценить» время дообучения — оно зависит от данных, которые вы увидите только после первой итерации. Вы не можете «зафиксировать» требования — заказчик сам поймёт, что ему нужно, только когда увидит, что выдает модель. Вы не можете «спланировать» качество — метрики могут прыгать, как горный козел.

Если вы до сих пор думаете, что ваша главная задача — следить за Jira, вы уже проиграли. Ваша новая задача — проектировать мышление команды. Создавать среду, в которой люди и ИИ дополняют друг друга, а не мешают. Решать не «как успеть к дедлайну», а «как принять правильное решение, когда у нас 80% неизвестности».

Эта книга — не про управление задачами. Она про управление неопределённостью. Про то, как перестроить свою роль, какие компетенции развивать, какие инструменты использовать и как не выгореть в новой реальности. Она основана на реальных кейсах, ошибках (да, я их совершал) и победах. Если вы готовы стать когнитивным архитектором — читайте дальше. Если хотите остаться диспетчером — закройте книгу. Искренне желаю вам сделать правильный выбор.

ГЛАВА 1. Кризис роли: почему классический PM умирает

(или «Капитан, который всё ещё смотрит на часы, пока корабль вошёл в шторм»)

Метафора. Стройка vs. Разведка

Представьте двух людей.

Первый — прораб на стройке. У него есть чертёж, смета, график поставки бетона. Он знает, что через три недели нужно залить фундамент, через два месяца возвести стены. Если что-то идёт не так, он смотрит в план, корректирует ресурсы и догоняет график. Его главный инструмент — секундомер и табличка с датами.

Второй — капитан разведывательного корабля, который отправляется в неизведанное море. На карте — лишь примерные очертания берегов, погода меняется каждые полчаса, а подводные течения не нанесены ни на один навигационный справочник. Он не знает, где будет через месяц. Он знает только направление — на восток, где по слухам есть земля. Его главный инструмент — не график, а бинокль, лот для

измерения глубины и умение спрашивать у команды: «Что вы видите? Что мы упускаем?»

Классический PM — это прораб. ИИ-проект — это разведка. И проблема в том, что большинство PM пытаются управлять разведкой методами стройки. Они рисуют диаграммы Ганта, дробят работу на спринты, требуют от дата-сайентистов «точной оценки» дообучения модели. А в ответ слышат: «Мы не знаем, это зависит от данных, которые мы получим после первого прогона». И тогда PM теряется. Он начинает давить, требовать, назначать встречи — и тем самым только усугубляет хаос.

Что на самом деле происходит в ИИ-проекте (и почему PMBOK пасует)

Давайте снимем розовые очки. Вот пять реальностей, с которыми сталкивается каждый PM в ИИ-разработке:

1. Требования — это не список, а гипотеза

В классике вы пишете ТЗ и утверждаете его. В ИИ заказчик говорит: «Мне нужна модель, которая будет предсказывать отток клиентов». Но он сам не знает, какая точность ему достаточна, какие признаки важны, допустима ли ложная тревога в 5% или только в 1%. Выясняется это только после того, как модель начинает выдавать первые результаты. Требования *рождаются* в процессе, а не фиксируются до него.

2. Сроки — это игра в рулетку

Вы оцениваете дообучение модели в две недели. Но через неделю выясняется, что данных слишком мало, нужно собрать новый датасет, а это ещё неделя. Или что модель переобучается, и нужно менять архитектуру. Любая оценка — это число с колоссальным разбросом, но бизнес требует «одну цифру». И вы даёте её, зная, что врёте.

3. Бюджет сжигается незаметно

Вы арендуете GPU-кластер, и он работает 24/7. Стоимость каждого эксперимента — сотни долларов. Команда запускает десятки экспериментов в неделю, многие из которых ни к чему не приводят. Вы не можете это контролировать классическим методом «согласования затрат», потому что в ИИ-разведке каждый неудачный эксперимент — это тоже ценная информация. Но это сложно объяснить финансовому директору.

4. Риски выходят за рамки «задержка поставки»

В ИИ есть риски, которых нет ни в одной книге PMBOK:

— Модель начинает галлюцинировать и выдавать опасный контент.

— Модель дискриминирует определённые группы пользователей.

— Модель «забывает» важные паттерны, потому что дан-

ные устарели (дрейф данных).

— Модель переобучается на шум, и метрики на тесте хорошие, а на реальных данных — провал.

5. Команда говорит на разных языках

Инженеры думают в терминах пайплайнов и latency. Дата-сайентисты — в терминах loss-функций и эпох. Бизнес-заказчик — в терминах ROI и конверсии. PM оказывается переводчиком, но он не знает, как перевести «градиентный спуск» в «увеличение прибыли». И вместо моста он становится глухой стеной.

Признаки того, что ваш классический PM уже мёртв (диагностический тест)

Проверьте себя. Если хотя бы три пункта из этого списка — про вас, вы уже на грани.

— Вы начинаете утро с открытия Jira и проверки статусов.

— Вы требуете от команды «точной оценки» задачи, зная, что она невозможна.

— Вы тратите больше времени на согласование изменения сроков, чем на обсуждение данных.

— Вы не понимаете, почему модель вдруг стала давать странные ответы, и вините во всём инженеров.

— Вы боитесь сказать заказчику: «Мы не знаем, получится ли, но мы попробуем».

— Вы не участвуете в обсуждении архитектуры промптов, потому что «это не моё».

— Вы считаете, что главное — уложиться в бюджет, даже если модель работает плохо.

Если вы узнали себя — не паникуйте. Это нормальная реакция на ненормальную ситуацию. Но признание — первый шаг к исцелению.

Что делать прямо сейчас? Практическая плоскость

Вместо того чтобы ждать, пока кризис вас съест, выполните три действия **в ближайшие 48 часов**.

Действие 1. Проведите «аудит неопределённости»

Составьте список из 10 самых больших неизвестных в вашем проекте. Не задач, а *неизвестных*. Например:

— Насколько репрезентативны наши обучающие данные?

— Как часто мы будем переобучать модель?

— Какие метрики качества действительно важны для бизнеса?

— Как мы будем реагировать на галлюцинации в продакшне?

— Каков запас прочности нашей инфраструктуры?

Затем оцените для каждой: **как мы узнаем, что этот вопрос решён?** У вас должны быть конкретные критерии. Если критериев нет — вы не управляете этим, вы надеетесь.

Действие 2. Замените одно совещание на «когнитивный разбор»

Вместо статус-митинга проведите встречу, на которой вы задаёте только три вопроса:

— Что мы узнали за последнюю неделю о данных, модели или процессах, чего не знали раньше?

— Какие наши предположения оказались ошибочными?

— Как это меняет наш план (не сроки, а направление)?

Записывайте ответы. Через месяц вы увидите, как картина проясняется.

Действие 3. Перестаньте врать про сроки

На следующей встрече с заказчиком скажите правду: «Мы можем дать диапазон — от двух до пяти недель, с вероятностью 80%. Мы будем обновлять прогноз каждую неделю. Можем ли мы работать в таком режиме?» Это шокирует заказчика, но если вы не начнёте сейчас, то через месяц он шокируется ещё сильнее, когда вы сорвёте обещанную дату.

Самая важная мысль главы

Классический PM — это диспетчер. Диспетчер управляет движением по расписанию. В ИИ-проекте расписания нет. Есть только направление. И если вы не перестанете цепляться за старые методы, вы станете не просто бесполезны — вы будете вредны, потому что создаёте иллюзию

контроля, которая мешает команде адаптироваться.

Кризис роли — это не повод для паники, а приглашение к трансформации. Первый шаг — осознать, что старые инструменты не работают. Второй шаг — начать использовать новые. Именно этому посвящены следующие главы.

ГЛАВА 2. Кто такой когнитивный архитектор

(или «Почему дирижёр важнее, чем первый скрипач»)

Метафора. Дирижёр vs. Диспетчер

Представьте себе два оркестра.

Первый оркестр играет по нотам. У каждого музыканта своя партия, есть метроном, есть дирижёр, который отсчитывает такты и следит, чтобы все вступали вовремя. Если кто-то сбивается, дирижёр стучит палочкой и указывает на такт. Это идеальный классический PM. Его работа — чтобы все играли по расписанию. Но что, если нот нет? Что, если оркестру нужно импровизировать джаз на основе одной лишь мелодии, услышанной вчера во сне?

Второй оркестр — джазовый ансамбль. У них есть тема, но нет партитуры. Саксофонист начинает, пианист подхватывает, барабанщик чувствует пульс. Каждый слушает других и реагирует в реальном времени. А есть человек, который стоит в стороне. Он не играет. Он не дирижирует в классическом смысле. Но он:

— выбирает, какую тему они будут развивать;
— решает, когда нужно сменить тональность;
— следит, чтобы никто не доминировал и не выпадал;
— создаёт безопасное пространство для экспериментов.

Этот человек — не дирижёр, он **когнитивный архитектор**. Он не управляет нотами, он управляет атмосферой, диалогом и направлением. Он не говорит: «Ты ошибся в такте», он спрашивает: «Что мы сейчас услышали? Как мы можем это развить?»

В ИИ-проекте вы — этот архитектор. Команда — джазовый ансамбль. А ИИ — новый инструмент, который умеет издавать звуки, которых никто раньше не слышал. Ваша задача — не указывать, когда нажимать клавиши, а создавать условия, чтобы из этого хаоса родилась музыка.

Определение. Когнитивный архитектор — это не PM

Давайте раз и навсегда зафиксируем определение.

Когнитивный архитектор — это проектный лидер, который управляет не задачами и сроками, а *мыслительными процессами* команды и качеством её диалога с ИИ. Он проектирует среду, в которой человеческая интуиция и машинная точность взаимодополняются, а не конфликтуют. Он отвечает на три вопроса:

— Как мы задаём вопросы ИИ?

— Как мы интерпретируем его ответы?

— Как мы учимся на каждой итерации, чтобы следующая была лучше?

Когнитивный архитектор не знает, как работает градиентный спуск. Но он знает, когда команда застревает в ложных предположениях. Он не пишет код. Но он видит, что инженеры тратят время на ручные промпты, которые можно шаблонизировать. Он не принимает бизнес-решения в одиночку. Но он выстраивает диалог между заказчиком и моделью так, чтобы ни у кого не возникало иллюзий. Его инструмент — не знание деталей, а способность видеть, где детали мешают видеть целое.

Сравнительная таблица: Диспетчер против Архитектора

Это не косметические отличия. Это разные профессии. Давайте посмотрим в лоб.

Параметр	Классический PM (Диспетчер)	Когнитивный архитектор
Что он контролирует	Сроки, бюджет, объём работ	Качество решений, уровень неопределённости, доверие
Главный вопрос	«Всё ли идёт по плану?»	«Правильно ли мы думаем?»
Отношение к ИИ	Инструмент для отчётов и автоматизации	Партнёр, поведение которого нужно проектировать
Как он управляет командой	Даёт задания, назначает ответственных	Создаёт контекст, задаёт вопросы, фасилитирует обсуждение
Инструменты	Jira, MS Project, диаграммы Ганта	Реестр промптов, дашборд вероятностей, журнал доверия
Как он реагирует на ошибку	Ищет виноватого, корректирует план	Ищет, что мы недоучли, меняет подход
Как он общается с заказчиком	Даёт обещания по срокам	Даёт сценарии и вероятности
Как он измеряет успех	Уложились в срок и бюджет	Модель приносит бизнес-ценность, команда не выгорела

Что когнитивный архитектор делает вместо «следить за Jira»

Разберём конкретный день.

Понедельник, 9:00. Вместо открытия Jira — открытие дашборда вероятностей

Вы не смотрите, сколько задач перешло в «Done». Вы смотрите на прогноз: с какой вероятностью текущая модель достигнет целевой метрики к концу следующего спринта? Этот прогноз строится на основе истории предыдущих экспериментов, а не на экспертных оценках. Вы видите распределение — и понимаете, где риск наибольший.

10:00. Вместо статус-митинга — «когнитивный аудит»

Вы собираете команду не для того, чтобы каждый отчитался о прогрессе. Вы задаёте три вопроса:

— Какие гипотезы мы проверили на прошлой неделе?

— Что из этого оказалось ложным, а что — подтвердилось?

— Как это меняет наш следующий шаг?

Вы записываете ответы в специальный документ — **Журнал гипотез**. Через месяц у вас будет карта того, как менялось ваше понимание проекта.

12:00. Вместо согласования изменения сроков — пересмотр стратегии HITL

Заказчик звонит и говорит: «Мы хотим запустить через месяц». Вместо того чтобы паниковать и давить на инженеров, вы отвечаете: «Давайте посмотрим, где мы можем снизить порог автоматического доверия к модели, а где оставить человека в контуре. Если мы разрешим модели принимать решения при уверенности 70% вместо 90%, мы сможем сократить срок на две недели. Но это повысит риск ошибок. Давайте вместе решим, какой уровень риска для нас приемлем».

Вы не врёте. Вы предлагаете осознанный выбор.

15:00. Вместо контроля переработок — дизайн сре-

ды

Вы замечаете, что дата-сайентист уже вторую неделю сидит допоздна. Вы не говорите ему: «Соберись». Вы идёте к нему и спрашиваете: «Какая часть твоей работы самая рутинная?» Он отвечает: «Я вручную проверяю каждый ответ модели, чтобы отфильтровать галлюцинации». Вы даёте задание инженеру написать скрипт автоматической проверки по ключевым паттернам. Через два дня дата-сайентист уходит в 18:00.

17:00. Вместо отчёта о прогрессе — запись в журнал доверия

Сегодня модель приняла важное решение — рекомендовала клиенту инвестиционный продукт. Вы фиксируете в журнале:

— Запрос, который получила модель.

— Ответ модели (с цепочкой рассуждений CoT).

— Какие данные использовались.

— Кто из команды проверил ответ.

— Какое решение принято.

— Ваш комментарий: «Доверие — высокое, потому что запрос был простым и данные свежие. В следующий раз для сложных кейсов добавим дополнительную проверку».

Практическая плоскость: три действия для перехода к новой роли

Вы прочитали это и думаете: «Звучит красиво, но я работаю в корпорации с Jira, отчётами и годовыми бюджетами. Как мне внедрить это без революции?»

Вот три конкретных шага, которые вы можете сделать *уже на этой неделе*, не спрашивая разрешения у начальства.

Действие 1. Нарисуйте «карту контекста»

Возьмите лист бумаги или доску Miro. В центре напишите «Текущая цель проекта» (например, «Увеличить точность предсказания оттока до 85%»). Вокруг нарисуйте кружки:

— **Данные** — что мы подаём в модель?

— **Модель** — архитектура, гиперпараметры.

— **Оценка** — как мы измеряем качество?

— **Интерпретация** — как мы объясняем результаты бизнесу?

— **Доверие** — где мы проверяем модель, а где доверяем?

Теперь для каждого кружка напишите **одну неизвестную**, которая вас беспокоит. Например, для «Данные» — «Мы не знаем, репрезентативна ли выборка». Для «Интерпретация» — «Мы не умеем объяснять решения модели заказчику».

Это ваша карта неопределённости. Каждую неделю пересматривайте её и отмечайте, что прояснилось.

Действие 2. Замените одно совещание на «вопросный час»

Вместо стандартного статус-митинга проведите встречу, на которой вы **запрещаете доклады «что сделано»**. Вы разрешаете только вопросы:

— «Что нас сейчас сбивает с толку?»

— «Какое решение мы откладываем, потому что боимся?»

— «Если бы мы могли спросить что угодно у ИИ, что бы мы спросили?»

Записывайте эти вопросы. Через месяц у вас будет архив главных точек роста.

Действие 3. Возьмите одну ответственность «в контур»

Выберите одну область, где вы сейчас не участвуете, но могли бы помочь. Например:

— Вы не знаете, как пишутся промпты. Попросите инженера показать вам три самых частых промпта и предложите создать для них шаблоны и реестр.

— Вы не понимаете метрики качества. Попросите дата-сайентиста в течение 15 минут объяснить вам, что означает ROC-AUC на языке бизнеса. Запишите это объяснение и используйте в общении с заказчиком.

Один шаг, одна новая зона. Без героизма.

Чемодан когнитивного архитектора: что у вас должно появиться после этой главы

К концу второй главы у вас уже должен быть **практический инструментарий** для перехода. Вот минимальный набор:

Что	Как выглядит	Для чего
Журнал гипотез	Таблица с колонками: гипотеза, данные, проверка, результат, следующее действие	Чтобы видеть, как меняется понимание проекта
Карта контекста	Визуальная схема с кружками «Данные», «Модель», «Оценка», «Интерпретация», «Доверие»	Чтобы не упускать из виду взаимосвязи
Список вопросов для команды	«Что мы не знаем?», «Какое предположение самое рискованное?», «Что мы упускаем?»	Чтобы сместить фокус с отчётности на размышление
Шаблон для диалога с заказчиком	«У нас есть три сценария: оптимистичный (вероятность 30 %), базовый (50 %), пессимистичный (20 %). Вот что нужно сделать для каждого.»	Чтобы перестать врать про сроки

Самая важная мысль главы

Когнитивный архитектор не управляет задачами. Он управляет вопросами, которые команда задаёт себе и ИИ. Он не распределяет работу, он проектирует пространство для диалога. В этом пространстве ошибки становятся данными, неопределённость — топливом, а команда — не исполнителями, а со-исследователями.

Вы не станете когнитивным архитектором за один день. Это сдвиг не в навыках, а в мышлении. Но если вы начнёте с трёх действий из этой главы — вы сделаете первый шаг. А первый шаг — это 90% пути.

ГЛАВА 3. Семь компетенций когнитивного архитектора

(или «МАЯК ПМ: как не сбиться с курса в тумане»)

Метафора. Семь узлов — семь инструментов

Представьте, что вы — капитан корабля в открытом море. У вас нет карты, но у вас есть компас, секстант, лот для измерения глубины, барометр, карта ветров, судовой журнал и телескоп. Каждый инструмент даёт вам часть информации. Вместе они превращают хаос в осмысленную навигацию.

МАЯК ПМ — это семь таких инструментов. По отдельности они полезны. Вместе они создают систему, которая позволяет вам вести команду сквозь туман неопределённости.

Акроним легко запомнить: **МАЯК** (Моделирование, Архитектура диалога, Ясность, Контроль доверия) + **ПМ** (Проектирование среды, Мета-обучение). В сумме — семь компетенций.

Компетенция 1. M — Моделирование неопределённости

(или «Как перестать врать про сроки и начать играть в вероятности»)

Метафора. Метеоролог vs. Гадалка

Гадалка говорит: «Завтра будет дождь». Метеоролог говорит: «Вероятность дождя — 70%, потому что атмосферный фронт движется с запада со скоростью 30 км/ч, но есть 30% шанс, что он сместится к северу и мы окажемся в сухой зоне». Гадалка даёт иллюзию определённости. Метеоролог даёт инструменты для принятия решений: «Если вы планируете пикник, возьмите зонт, но если риск вам не подходит — перенесите на завтра».

В ИИ-проектах мы все становимся метеорологами. Вы не можете сказать: «Модель будет готова через три недели». Вы можете сказать: «С вероятностью 60% мы уложимся в три недели, с 30% — в четыре, с 10% — понадобится пересбор данных, и тогда срок вырастет до шести недель». Это не слабость — это честность.

Что это даёт на практике

— Вы перестаёте давать ложные обещания.

— Вы можете показать заказчику распределение рисков и

предложить варианты: «Если мы увеличим бюджет на GPU, мы сократим разброс. Если мы упростим модель — ускоримся, но потеряем качество».

— Вы начинаете принимать решения на основе *ожидаемой ценности*, а не пессимизма или оптимизма.

Конкретное упражнение (сделайте завтра)

Возьмите любую задачу, которую вы оценили в «X дней». Нарисуйте три сценария:

— **Оптимистичный** (всё идёт идеально) — X * 0.7

— **Базовый** (обычные сложности) — X * 1.0

— **Пессимистичный** (что-то пошло не так) — X * 1.8

Теперь назначьте вероятности этим сценариям. Например: 20% / 50% / 30%. Покажите это заказчику. Спросите: «Какой уровень риска для вас приемлем? Мы можем сместить распределение, инвестируя в данные или вычислительные ресурсы».

Вы только что применили моделирование неопределённости.

Компетенция 2. A — Архитектура диалога с ИИ

(или «Почему плохой промпт хуже плохого кода»)

Метафора. Переводчик vs. Телефонный справочник

Есть два способа общаться с иностранцем. Первый — вы даёте ему список фраз из разговорника. Второй — вы нанимаете профессионального переводчика, который понимает контекст, улавливает нюансы, адаптируется к ситуации.

Большинство команд общаются с ИИ как с разговорником: «Сделай то», «Напиши это». А надо как с переводчиком: объяснять контекст, уточнять, задавать уточняющие вопросы, просить показать ход мыслей.

Архитектура диалога — это искусство формулировать запросы так, чтобы ИИ выдавал полезные, проверяемые и объяснимые ответы.

Что это даёт на практике

— Вы уменьшаете галлюцинации, потому что просите модель показать цепочку рассуждений (CoT).

— Вы повышаете качество ответов, потому что даёте модели достаточно контекста.

— Вы создаёте реестр промптов, который позволяет мас-

штабировать успешные практики на всю команду.

Конкретное упражнение (сделайте сегодня)

Возьмите самый частый запрос к ИИ в вашем проекте. Напишите его в **трёх версиях**:

— Базовый — просто вопрос.

— С контекстом — вопрос + фон + ограничения + примеры.

— С цепочкой рассуждений — вопрос + попросите модель сначала перечислить факты, затем сделать вывод, затем предложить решение.

Протестируйте все три версии на одном и том же наборе кейсов. Сравните качество ответов. Зафиксируйте лучший вариант в реестре. Обучите команду использовать этот шаблон.

Вы только что стали архитектором диалога.

Компетенция 3. Я — Ясность мышления

(или «Как отличить шум от сигнала, когда модель говорит складно»)

Метафора. Шахматист vs. Паникёр

Шахматист смотрит на доску и видит не фигуры, а угрозы и возможности. Он не реагирует на первую атаку — он просчитывает варианты. Паникёр видит, что противник взял пешку, и сразу сдаётся.

ИИ выдаёт ответы, которые *звучат* уверенно. Но уверенность модели — не гарантия истины. Вы должны уметь отделять факты от предположений, корреляцию от причинности, шум от сигнала. Вы учитесь задавать себе вопросы: «Какие данные лежат в основе этого ответа?», «Какие альтернативные объяснения возможны?», «Если бы я не знал, что ответ сгенерирован ИИ, я бы ему поверил? Почему?»

Что это даёт на практике

— Вы перестаёте принимать решения на основе красиво сформулированной лжи.

— Вы учите команду сомневаться конструктивно, а не парализующе.

— Вы снижаете риск «автоматизации ошибки» — когда

модель повторяет системное смещение, а вы его не замечаете.

Конкретное упражнение (сделайте на следующем обсуждении)

Возьмите один ответ модели, который кажется убедительным. Проведите **«тест пяти почему»**

Конец ознакомительного фрагмента.