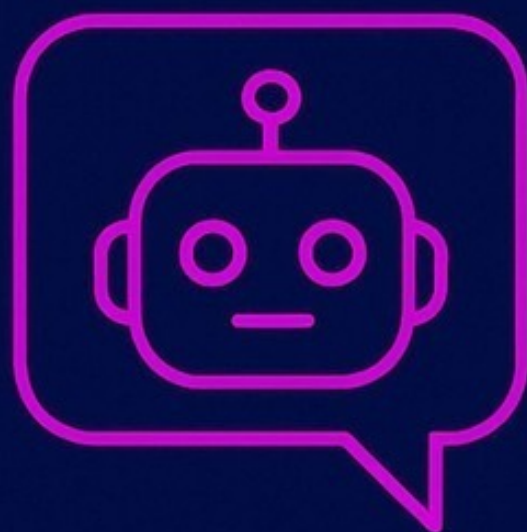


Владимир Васьков

# КАК ГОВОРИТЬ С ИИ



ТЕХНИКИ, СТИЛИ,  
ШАБЛОНЫ И ЛАЙФХАКИ  
ДЛЯ ОБЩЕНИЯ С  
LLM НЕЙРОСЕТЯМИ

Владимир Васьков

**Как говорить с ИИ. Техники,  
стили, шаблоны и лайфхаки для  
общения с LLM нейросетями**

«Издательские решения»

## **Васьков В.**

Как говорить с ИИ. Техники, стили, шаблоны и лайфхаки для общения с LLM нейросетями / В. Васьков — «Издательские решения»,

ISBN 978-5-00-681055-6

Как управлять искусственным интеллектом с помощью текста? Эта книга — ваш быстрый старт в мире промптинга. Освойте ключевые техники взаимодействия с ChatGPT и другими ИИ: Zero-shot, Few-shot, Chain-of-Thought и ReAct. Узнайте, как составлять промпты, которые решают задачи в бизнесе, обучении, маркетинге и коде. Избегайте типичных ошибок и превратите ИИ в мощный инструмент решений рутинных задач. Промпт — ваш новый навык XXI века.

ISBN 978-5-00-681055-6

© Васьков В.  
© Издательские решения

# Содержание

Страница об авторе	6
введение	8
Введение в промптинг и языковые модели	9
Обучение в контексте (In-Context Learning, ICL)	11
Типичные ошибки при использовании In-Context Learning (ICL)	13
Вывод	14
Написание промптов с малым количеством примеров (Few-Shot Prompting)	15
K-Nearest Neighbor (KNN)	16
Vote-K Prompting	18
Self-Generated In-Context Learning (SG-ICL)	20
Prompt Mining	22
Вывод	24
Сложные техники Few-Shot Prompting	25
LENS (Итеративная фильтрация)	26
UDR (Умное извлечение данных)	28
Active Example Selection	30
Вывод	32
Написание промптов с нулевым контекстом (Zero-Shot Prompting)	33
Role Prompting	34
Style Prompting	36
Emotion Prompting	38
System 2 Attention (S2A)	40
Вывод	42
Дополнительные техники Zero-Shot Prompting	43
SimToM	44
Rephrase and Respond (RaR)	46
Re-reading (RE2)	48
Self-Ask	50
Вывод	52
Генерация мыслей (Chain of Thought, CoT) или логическая цепочка	53
Zero-Shot-CoT – логическая цепочка без примеров	54
Конец ознакомительного фрагмента.	55

# **Как говорить с ИИ**

## **Техники, стили, шаблоны и лайфхаки**

### **для общения с LLM нейросетями**

**Владимир Васьков**

© Владимир Васьков, 2025

ISBN 978-5-0068-1055-6

Создано в интеллектуальной издательской системе Ridero

## Страница об авторе

Привет, друзья! Меня зовут Владимир Васьков.

Я – предприниматель, эксперт по продажам, привлечению инвестиций и внедрению нейросетей в бизнес. Уже более 13 лет я помогаю компаниям системно расти – от выстраивания отделов продаж до получения многомиллионного финансирования. За это время мои клиенты привлекли более 43 миллиардов рублей, а я лично обучил и проконсультировал более 35 000 предпринимателей и специалистов.

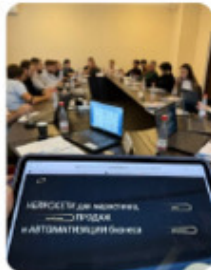
Я основал консалтинговую компанию VASKOV CONSULTING, веду интенсивы, стратегические сессии и корпоративные тренинги, выступаю как наставник в проектах «ОПОРА РОССИИ», «Мой бизнес», «Росмолодёжь», «Малый Бизнес Москвы» и ряде региональных программ развития. В моём портфолио – проекты с компаниями Сбер, Лукойл, Яндекс Go, Минэкономразвития, ЕвроХим, Роквул и другими.

В основе моих подходов – простота, применимость и результат. Именно поэтому я написал эту мини-книгу: чтобы поделиться с вами рабочими техниками взаимодействия с языковыми моделями, которые реально помогают в жизни и бизнесе. Без «воды», с практикой, примерами и готовыми шаблонами.

Если вы – специалист, предприниматель или просто увлечённый человек, которому интересно, как эффективно работать с ИИ, – эта книга для вас.

## Результаты в цифрах

<b>13+</b> Лет опыта в построении системы продаж	<b>43+</b> Млрд. руб. помог привлечь инвестиций своим клиентам	<b>35000+</b> Участников и предпринимателей обучил на тренингах и семинарах
<b>14+</b> Лет опыта публичных выступлений	<b>20+</b> Открыл розничных салонов-магазинов в сети	<b>3000+</b> Собеседований, отобрал и утвердил 20+ директоров и 150+ менеджеров
<b>15+</b> Лет управленческого опыта	<b>1000+</b> выступлений на бизнес-мероприятиях всех форматов	<b>200+</b> Сертификатов, дипломов и благодарностей
<b>100+</b> Проведено управленческих аудитов и выстроил систему обучения и управления в компаниях	<b>50+</b> Компаниях внедрил «Систему продаж» что позволило увеличить оборот компаниям от 25 до 54% в первые 6 месяцев внедрения	<b>5</b> Компаний упаковал во франшизу



## ВВЕДЕНИЕ

Мы живём в эпоху, когда тексты, коды, стратегии и даже бизнес-идеи может создавать искусственный интеллект. Точнее – языковая модель.

Но чтобы ИИ был по-настоящему полезен, недостаточно просто ввести запрос.

Именно здесь на сцену выходит prompt engineering – искусство и наука общения с нейросетями.

### **Для кого эта книга?**

Эта мини-книга создана не для учёных и не для тех, кто пишет код с 8 лет. Она для людей, которые хотят использовать ИИ как реального помощника: в работе, учёбе, творчестве, бизнесе или даже в быту.

Здесь вы не найдёте заумной терминологии или лишней теории.

Только понятные подходы, проверенные шаблоны и примеры, которые работают.

### **Что вы получите?**

На этих страницах – всё, что нужно для старта и роста в мире нейросетей:

- Узнаете как устроены языковые модели и как они «понимают» текст
- Разберётесь в техниках Zero-shot, Few-shot и Chain of Thought
- Изучите как «настроить» ИИ под себя с помощью ролей и инструкций
- Избежите типичных ошибок и не тратьте время впустую
- Научитесь использовать готовые промпты для маркетинга, аналитики, креатива, обучения и др.
- Начнете применять лучшие сервисы, расширения и инструменты для создания идеальных промптов

### **Как читать?**

Можно подряд. А можно – по главам, по задачам, по вдохновению. Эта книга не требует технических знаний, но поможет почувствовать себя уверенно в общении с ChatGPT и другими ИИ-инструментами.

Всё, что вам нужно – желание научиться задавать правильные вопросы.

Помните: не все беседы с ИИ приводят к идеальному результату с первого раза – экспериментируйте с техниками и все получится.

Остальное – внутри. Готовы? Тогда начнём!

## Введение в промптинг и языковые модели

Большие языковые модели (LLM) – это не только технология будущего, но и полноценный инструмент настоящего, который постепенно проникает в повседневную жизнь. Мы встречаем их в чат-ботах, голосовых помощниках, генераторах текста, переводчиках, редакторах и даже в инструментах анализа данных.

### Но как они работают?

Всё начинается с простого: с текста, который вы вводите в модель. Это и есть промпт: ваш запрос, инструкция или вопрос, на который модель должна ответить. Промпт – это диалог с умным помощником: чем яснее вы формулируете мысль, тем точнее и полезнее будет ответ.

### Типы промптов

Промпт – это то, как вы «разговариваете» с моделью.

Например:

- «Составь план запуска онлайн-курса по дизайну интерьеров»
- «Придумай название для кофейни у метро»
- «Сравни Apple и Samsung по четырем критериям: цена, камера, батарея, экосистема»

Даже такая простая фраза, как «Напиши шутку про бухгалтеров», – это уже промпт. И от его качества зависит, насколько хорошим будет результат.

### Зачем нужны хорошие промпты?

Хороший промпт – это не просто запрос. Это инструмент настройки модели. Как правильно заданный вопрос может развернуть глубокий разговор, так и правильно сформулированный промпт «включает» в модели нужные знания, стиль и формат.

Плохой промпт → туманный ответ

Чёткий промпт → понятный, полезный и структурированный результат

### Текстовые и мультимодальные промпты

Существует два основных типа промптов:

1. Текстовые – самый распространённый формат. Вы вводите текст – модель отвечает текстом.

Пример: «Сделай краткое резюме этой статьи» или «Напиши email в деловом стиле»

2. Мультимодальные – вы даёте на вход не только текст, но и изображения, аудио, видео.

Пример: загрузить фото интерьера и попросить «оценить стилистику и предложить улучшения», или отправить аудиофайл и запросить «выдели ключевые тезисы».

### Почему эта тема важна?

Промптинг – это новая цифровая грамотность. Умение правильно общаться с ИИ скоро станет таким же важным, как знание Excel или Google Docs. И хотя языковые модели становятся всё умнее, они по-прежнему зависят от того, как вы с ними разговариваете.

### Зачем нужна эта книга?

Сегодня множество специалистов – от маркетологов и учителей до разработчиков и предпринимателей – используют LLM в работе, но при этом не знают, как получить от них максимум. Промпты часто пишутся «интуитивно», без структуры, без понимания техник и возможностей. Результат – непредсказуемость, ошибки, повторная работа.

Цель этой книги – систематизировать знания о промптинге и сделать их доступными без технического жаргона. Мы рассмотрим десятки техник, подходов, примеров и шаблонов – всё, что поможет вам уверенно и эффективно взаимодействовать с языковыми моделями.

Используйте данную книгу как настольную и это приведет вас к максимальному использованию LLM нейросетей в свою пользу.

## Обучение в контексте (In-Context Learning, ICL)

### Адаптация модели без дообучения – прямо в промпте

Одной из самых революционных особенностей современных языковых моделей стало их умение учиться прямо в момент запроса – без традиционного переобучения. Это свойство получило название In-Context Learning (ICL), или обучение в контексте.

Суть метода проста: вы предоставляете модели несколько примеров прямо в теле промпта, и она начинает выполнять аналогичную задачу, подстраиваясь под структуру, стиль и логику этих примеров. Такое обучение «на лету» помогает решать широкий спектр задач: от настройки формата генерации до имитации специализированных процессов.

### Что такое ICL?

In-Context Learning (ICL) – это способность языковой модели выполнять задачу, опираясь только на содержимое промпта, в котором содержатся инструкции, примеры и формат ответа. Модель не изменяет свои параметры, но временно перенимает паттерн поведения, заданный примерами.

### Пример:

Вместо того чтобы обучать модель классифицировать отзывы, можно вставить 3—5 примеров:

Отзыв: «Очень понравилось обслуживание» → Метка: Позитивный

Отзыв: «Было слишком долго и шумно» → Метка: Негативный  
Затем – новый отзыв, без метки.

Модель продолжит по шаблону.

### Принципы эффективного ICL

1. Количество примеров: Эффективность ICL увеличивается по мере роста числа примеров (часто достаточно 3—5 для ChatGPT, до 30—50 для GPT-4).
2. Порядок примеров: Модели склонны обращать больше внимания к последним примерам («эффект новизны»).
3. Балансировка классов: В задачах анализа текста важно равномерное распределение примеров по классам.
4. Качество разметки: Примеры с экспертной аннотацией повышают точность ICL до 15—25%.
5. Единый формат: Используйте чёткие структуры – «Пример 1:», «Вход:», «Выход:».
6. Семантическое сходство: Примеры должны соответствовать целевой задаче по стилю, длине и тематике.

### Где применяется ICL на практике

- Быстрая адаптация модели к бизнес-стилю
- Классификация обращений
- Обработка таблиц и форм
- Проверка гипотез без датасета
- Симуляция экспертов по ролям

### Преимущества

- Не требует дообучения модели

- Гибкий подход
- Подходит для генерации и анализа
- Прототипирование функций за минуты
- Совместим с любой LLM, поддерживающей промпты

## Типичные ошибки при использовании In-Context Learning (ICL)

Чтобы ICL работал эффективно, важно избегать распространённых ошибок, которые снижают точность, сбивают модель с толку или делают промпт неоднозначным:

### 1. Смешение разных форматов в примерах

**Ошибка:** часть примеров в формате «Вопрос – Ответ», часть – в стиле таблицы или тезисов.

**Почему плохо:** модель теряет единообразие шаблона.

**Что делать:** придерживайтесь одного чёткого формата во всех примерах.

### 2. Слишком обобщённые или абстрактные формулировки

**Ошибка:** инструкции вроде «сделай хорошо» или «напиши красиво».

**Почему плохо:** модель не понимает критерии качества.

**Что делать:** формулируйте конкретно: стиль, тональность, структура, ограничения.

### 3. Слишком длинные или перегруженные примеры

**Ошибка:** вставка примеров на 500+ слов каждый.

**Почему плохо:** вы тратите контекстное окно и «утопаете» в деталях.

**Что делать:** сокращайте примеры до сути (100—150 слов), убирая лишнее.

### 4. Примеры не соответствуют целевой задаче

**Ошибка:** хотите, чтобы модель писала статьи, а вставили примеры постов.

**Почему плохо:** модель перенимает стиль и формат, который вам не нужен.

**Что делать:** подбирайте примеры максимально похожие на желаемый результат.

### 5. Нарушение баланса классов в задачах классификации

**Ошибка:** 5 позитивных отзывов и 1 негативный.

**Почему плохо:** модель начинает «переобучаться» на доминирующий класс.

**Что делать:** соблюдайте равенство: 2—3 примера на каждый класс.

### 6. Отсутствие разграничителей или заголовков в примерах

**Ошибка:** текст сливается в одну «массу», модель не видит структуру.

**Что делать:** используйте чёткие маркеры:

Пример :

Вход:...

Выход:...

### 7. Добавление нерелевантной информации в промпт

**Ошибка:** вставка вводного текста, который не относится к задаче.

**Почему плохо:** модель может принять это за часть шаблона.

**Что делать:** очищайте промпт от всего лишнего – только то, что влияет на поведение модели.

## **Вывод**

In-Context Learning – ключевая техника адаптации модели к задаче с помощью примеров прямо в запросе. Она позволяет быстро и гибко управлять поведением ИИ без технических барьеров.

## Написание промптов с малым количеством примеров (Few-Shot Prompting)

### Как научить модель на лету – без переобучения и кода?

Few-Shot Prompting – это одна из ключевых техник эффективной работы с языковыми моделями. Эта техника позволяет быстро адаптировать поведение модели к нужному стилю, логике и формату ответа. Этот метод особенно ценен, когда необходимо быстро перейти от «общего интеллекта» модели к решению специализированных задач, не прибегая к тонкой настройке или дообучению.

### В чём суть Few-Shot подхода?

Модель получает 2—4 примера выполнения аналогичной задачи, оформленных в едином стиле, например:

- вопрос → ответ
- задание → решение
- вход → желаемый выход

На основе этих примеров модель перенимает структуру, стиль, тональность и формат ответа.

### Где особенно эффективен метод:

- Генерация шаблонных текстов (описания товаров, вакансий, писем)
- Форматирование данных и работа с таблицами
- Написание кода в заданном стиле
- Классификация и категоризация данных
- Поддержание заданного стиля коммуникации (официальный, дружелюбный и т.д.)

### Что входит в продвинутый Few-Shot Prompting:

Помимо простого указания пары примеров, сегодня применяются и более сложные методы, которые усиливают результат:

- KNN (K-Nearest Neighbors) – выбор наиболее похожих примеров по смыслу
- Vote-K – множественная генерация и отбор лучших примеров
- SG-ICL (Self-Generated In-Context Learning) – когда модель сама создаёт себе примеры
- LENS, UDR, Active Selection – автоматизированные стратегии улучшения выборки

Few-Shot Prompting – гибкий и масштабируемый инструмент, который можно адаптировать под любую задачу. В этой главе мы научимся не только формировать примеры, но и выбирать лучшие, тестировать разные форматы, и строить на их основе надёжную и понятную систему взаимодействия с ИИ.

## K-Nearest Neighbor (KNN)

K-Nearest Neighbor (KNN) Prompting – это техника, при которой для нового запроса автоматически подбираются наиболее похожие примеры из базы заданий (промпов) и вставляются в запрос к языковой модели.

Смысл в том, что модель лучше справляется с задачей, если ей показать не случайные примеры, а те, что максимально близки по содержанию или структуре к текущему запросу.

### Как работает:

- Вы формируете базу промптов с готовыми примерами задач и ответов.
- Когда поступает новая задача, система ищет в базе N наиболее похожих примеров.
- Эти примеры подаются в промпт перед новой задачей – как контекст.

### Пример:

Если пользователь спрашивает:

«Как улучшить вовлечённость в рассылке для интернет-магазина одежды?»»

Система может найти в базе похожие вопросы:

«Как повысить CTR email-рассылки для e-commerce?»»

«Как сделать письма более кликабельными?»»

И подставить их как Few-Shot-примеры перед ответом.

### Преимущества:

- Даёт более точные ответы за счёт релевантного контекста
- Масштабируется при наличии базы готовых кейсов
- Хорошо работает в условиях ограниченного количества примеров

**Пример шаблона промпта с использованием метода K-Nearest Neighbor (KNN) в контексте классификации отзывов по тональности:**

#### Задача:

Классифицировать отзыв клиента как положительный, нейтральный или отрицательный.

#### Промпт (шаблон KNN-подхода):

Ты – система анализа тональности. Используй приведённые примеры отзывов, чтобы определить тональность нового текста. Оцени схожесть и выбери класс на основе наиболее похожих.

Примеры:

Отзыв: «Приложение работает быстро, всё удобно и понятно.»

Тональность: Положительная

Отзыв: «Интерфейс хороший, но часто зависает.»

Тональность: Нейтральная

Отзыв: «Постоянно вылетает, невозможно нормально пользоваться.»

Тональность: Отрицательная

Теперь проанализируй следующий отзыв и укажи его тональность с объяснением выбора:

Отзыв: «Функции вроде есть, но половина не работает, техподдержка игнорирует.»

**Ожидаемый ответ:**

**Тональность:** Отрицательная

**Обоснование:** Отзыв содержит негативные оценки («не работает», «техподдержка игнорирует»), что схоже с примером «Постоянно вылетает...» – следовательно, классификация – отрицательная.

**Такой шаблон можно адаптировать под любую задачу:**

- Классификация эмоций
- Определение категории текста
- Маркировка спама / не-спама
- Подбор релевантных решений по аналогии

## Vote-K Prompting

Vote-K – это техника, при которой языковая модель сначала генерирует несколько разных вариантов ответа (кандидатов) на одну и ту же задачу, без указания правильного ответа.

Затем каждый из этих вариантов оценочно сравнивается – либо самой моделью, либо внешним критерием (например, другим промптом или экспертом). Побеждают те ответы, которые получают наибольшее количество "голосов" – то есть считаются наилучшими по качеству или соответствию цели.

После этого отобранные (аннотированные) варианты могут использоваться:

- как шаблоны для будущих промптов,
- как контекст в последующих Few-shot задачах,
- для обучения более точной версии модели.

### Как работает:

- Модель генерирует K вариантов ответа.
- Оценивает или сравнивает их (через вспомогательный промпт или критерия).
- Выбирает наиболее релевантные (по «голосам»).
- Использует выбранные как примеры или лучшие ответы.

### Применение:

- Улучшение качества ответов (особенно в генеративных задачах)
- Повышение надёжности в задачах с высоким риском ошибок
- Сбор обучающих данных на основе лучших сгенерированных примеров

### Пример:

Задача: «Сформулируй заголовок к email-рассылке по акции»

Модель генерирует:

- «Скидки до 50% – только сегодня!»
- «Распродажа мечты – непусти шанс»
- «Время обновить гардероб – акции внутри»

Вспомогательный промпт или оценщик выбирает 2 и 3 как лучшие. Они попадают в финальный ответ или используются как примеры в будущем.

### Преимущества:

- Меньше случайных или слабых ответов
- Возможность повысить точность без дополнительного обучения
- Сбор лучших практик и наглядных кейсов в автоматическом режиме

**Пример шаблона промпта с использованием метода Vote-K, где модель сначала генерирует несколько «кандидатов-решений», а затем выбирает наилучшие и обучается на них:**

#### Задача:

Написать заголовок к статье на тему «Как повысить продуктивность при удалённой работе»

#### Промпт-шаблон (Vote-K):

Ты – редактор онлайн-медиа. Сначала предложи 5 вариантов заголовков к статье на тему «Как повысить продуктивность при удалённой работе». Затем выбери 2 лучших по критериям: цепкость, ясность, соответствие теме. Объясни свой выбор.

ШАГ 1: Сгенерируй 5 заголовков.

ШАГ 2: Выбери 2 лучших и объясни, почему именно они.

**Ожидаемый ответ:**

**ШАГ 1: Кандидаты (5 заголовков):**

1. Как быть продуктивным, работая из дома
2. Твоя квартира – твой офис: секреты эффективности
3. Продуктивность на удалёнке: 7 проверенных методов
4. Работай из дома, как профи: гайд для современного специалиста
5. Почему ты устаёшь на удалёнке и как это исправить

**ШАГ 2: Отбор лучших:**

*Продуктивность на удалёнке: 7 проверенных методов* – коротко, по делу, с числом (это работает в заголовках)

*Работай из дома, как профи: гайд для современного специалиста* – конкретика и ценность («гайд»), обращение к целевой аудитории

Эти лучшие примеры затем можно использовать в следующих промптах как эталонные (аннотированные) для дообучения или генерации в другом стиле.

## Self-Generated In-Context Learning (SG-ICL)

SG-ICL – это техника, при которой сама языковая модель генерирует примеры (задание → ответ), если у пользователя нет готовых обучающих данных или шаблонов.

Модель сначала получает задачу, а затем сама предлагает 1—3 примера, которые можно использовать в качестве контекста для дальнейшей генерации. Метод позволяет обойтись без ручной подготовки примеров, что особенно полезно в новых или нестандартных ситуациях.

### Как работает:

- Пользователь формулирует задачу (например, «Напиши описание продукта по названию и характеристикам»).
- Модель сама придумывает 2—3 примера выполнения этой задачи.
- Эти примеры вставляются в промпт как контекст.
- Далее модель продолжает работу по аналогии с созданными ею же примерами.

### Пример:

#### Промпт:

«Сгенерируй несколько примеров описания товаров, а затем создай новое по образцу. Категория: Умная колонка

Характеристики: голосовое управление, Bluetooth, дизайн в стиле минимализм»

Модель может сама добавить:

#### Пример 1:

Категория: Фитнес-браслет

Особенности: Пульсометр, трекер сна, OLED-экран

Описание: Умный помощник для вашего здоровья – контролируйте активность и сон в реальном времени.

#### Пример 2:

Категория: Беспроводные наушники

Особенности: Шумоподавление, автономность до 20 ч, зарядка через USB-C

Описание: Чистый звук без проводов – комфорт и качество в любом ритме жизни.

Категория: Умная колонка

Особенности: голосовое управление, Bluetooth, дизайн в стиле минимализм

Описание:

### Применение

- Когда нет готовых данных, но хочется применить Few-shot подход
- Для ускоренной генерации шаблонов «на лету»
- При создании обучающих материалов, типовых форм, инструкций

### Преимущества:

- Экономит время: не нужно вручную собирать примеры
- Повышает согласованность вывода

– Помогает работать с новыми темами без предварительной подготовки

**Шаблон промпта с использованием метода Self-Generated In-Context Learning (SG-ICL) – когда модель сама генерирует обучающие примеры, чтобы затем использовать их для выполнения задачи.**

**Задача:**

Определить тональность пользовательских отзывов.

**Промпт-шаблон (SG-ICL):**

Ты – система анализа тональности. У тебя пока нет готовых обучающих данных, поэтому сначала сгенерируй 3 примера отзывов с метками тональности: положительная, нейтральная, отрицательная.

Затем, используя эти примеры, проанализируй новый отзыв и укажи его тональность с обоснованием.

ШАГ 1: Сгенерируй обучающие примеры.

ШАГ 2: Проанализируй целевой отзыв.

**Ожидаемый ответ:**

**ШАГ 1 – Сгенерированные примеры:**

1. *«Очень доволен, всё работает быстро и стабильно.»*

**Тональность:** Положительная

2. *«Приложение нормальное, но ничего особенного.»*

**Тональность:** Нейтральная

3. *«После обновления не запускается, жалею о покупке.»*

**Тональность:** Отрицательная

**ШАГ 2 – Новый отзыв:** *«Интерфейс удобный, но часто возникают ошибки при оплате.»*

**Тональность:** Нейтральная

**Обоснование:** Как и во втором примере, есть положительные и негативные моменты, что даёт смешанную, но сбалансированную оценку.

**SG-ICL особенно полезен, когда:**

- нет доступа к реальным примерам
- нужно быстро адаптировать модель к новому типу задач
- необходимо протестировать гипотезу «на лету»

## Prompt Mining

Prompt Mining – это метод систематического поиска наиболее эффективных формулировок промптов для конкретной задачи. Вместо того чтобы полагаться на шаблон «вопрос → ответ», этот подход предполагает анализ и тестирование различных вариантов инструкций, чтобы найти те, которые дают наилучший результат.

По сути, это исследование различных формулировок промптов, направленное на выявление наиболее результативных.

### Как работает

- Берётся конкретная задача (например, генерация текстов, классификация, объяснение понятий).
- Генерируется или собирается большое количество промптов с разными формулировками.
- Все варианты тестируются на модель, и сравниваются по результатам (качество, полнота, стиль, точность).
- Выявляются наиболее результативные шаблоны, которые затем можно стандартизировать и использовать повторно.

### Пример:

Вместо простого:

«Объясни, что такое экономическая рецессия»

Могут быть протестированы такие варианты:

«Ты – профессор экономики. Объясни студенту, что такое рецессия, простыми словами.»

«Приведи краткое определение экономической рецессии с примерами.»

«Объясни термин „экономическая рецессия“ как для старшеклассника, используя аналогии.»

Анализ покажет, какой формат даёт самый понятный и точный результат.

### Когда применять:

- При разработке промптов для массового использования (чат-боты, обучающие ИИ-сценарии, генерация контента)
- В UX-исследованиях взаимодействия с LLM
- Для повышения качества и надёжности ответов в критичных задачах (финансы, медицина, право)

### Преимущества:

- Позволяет находить нетривиальные, но эффективные способы взаимодействия с моделью
- Повышает качество вывода без изменений в самой модели
- Помогает формировать библиотеку рабочих промптов под задачи бизнеса, продукта или пользователя

**Пример-шаблон, демонстрирующий Prompt Mining – метод, при котором мы ищем и тестируем разные формулировки промптов, чтобы найти наиболее эффективные.**

**Задача:**

Получить от модели краткий и точный пересказ новостной статьи.

**Промпт-шаблон (Prompt Mining):**

Твоя задача – найти наиболее эффективную формулировку промпта для задачи: «Сделай краткий пересказ новости».

**ШАГ 1:** Придумай 3 разных формулировки промпта, каждая в разном стиле.

**ШАГ 2:** Применишь их к одному и тому же тексту.

**ШАГ 3:** Сравни результаты и выбери лучший по критериям: точность, краткость, логичность.

Текст новости:

«Компания X анонсировала запуск новой платформы для анализа данных. Решение предназначено для малого и среднего бизнеса. Ожидается, что запуск состоится в третьем квартале 2025 года.»

Начинай.

**Ожидаемый ответ:**

**ШАГ 1 – Варианты промптов:**

1. **Формулировка 1 (инструктивный):** «Сделай краткий пересказ следующей новости.»

2. **Формулировка 2 (ролевой):** «Ты – редактор новостного сайта. Напиши короткую аннотацию для читателей.»

3. **Формулировка 3 (журналистский стиль):** «Представь, что ты пишешь заголовок и лид к заметке. Перескажи суть.»

**ШАГ 2 – Результаты:**

1: «Компания X запускает платформу для аналитики данных. Релиз – в Q3 2025.»

2: «Компания X объявила о запуске новой аналитической платформы для малого и среднего бизнеса. Выход – в третьем квартале 2025.»

3: «X запускает новый продукт: аналитическая платформа для бизнеса. Релиз – Q3 2025.»

**ШАГ 3 – Вывод:** Промпт №2 дал наиболее ясный и сбалансированный ответ – подходит как для пересказа, так и для читательской аннотации. Рекомендуется использовать его как шаблон.

## **Вывод**

Few-Shot Prompting не только позволяет эффективно решать задачи с минимальным числом примеров, но и открывает возможности для масштабирования и оптимизации. Благодаря тестированию различных формулировок и сбору метрик, можно систематически находить наиболее результативные шаблоны, что делает этот подход мощным инструментом для тонкой настройки генерации, классификации, перевода и других LLM-задач.

## Сложные техники Few-Shot Prompting

### Как обучать модель на лету без переобучения?

Few-Shot Prompting – это мощный инструмент, который позволяет обучать языковую модель на лету, предоставляя ей всего несколько примеров для ориентира. Но за рамками базового подхода («вопрос → ответ», повторенный 2—3 раза) скрывается целый пласт продвинутых стратегий, которые превращают Few-Shot в настоящую систему настройки модели под задачу – без дополнительного обучения.

В этой главе мы сосредоточимся на сложных техниках Few-Shot Prompting, таких как:

- LENS – итеративная фильтрация, которая отбирает только релевантные примеры и постепенно улучшает качество результата.
- UDR – умное извлечение данных на основе векторной близости и семантики.
- Active Example Selection – активный выбор примеров с использованием обратной связи и обучения с подкреплением.

### В чём сила этих методов?

- Они позволяют адаптировать модель к конкретной задаче за секунды – без переобучения и без доступа к большим датасетам.
- Они повышают точность и устойчивость модели в нестандартных или сложных сценариях.
- И что особенно важно: они открывают путь к масштабируемому и автоматизированному созданию примеров, что делает Few-Shot не разовой техникой, а системным инструментом.

### Эти техники особенно актуальны в условиях, когда:

- требуется высокая точность без дополнительного обучения,
- есть ограниченный доступ к обучающим данным,
- или задача постоянно меняется в зависимости от контекста и цели.

В следующих разделах вы найдёте подробное описание каждого метода, примеры их использования и шаблоны промптов, которые можно адаптировать под собственные задачи.

## LENS (Итеративная фильтрация)

LENS – это метод, при котором качество используемых примеров для языковой модели постепенно улучшается за счёт последовательной фильтрации. Система или пользователь многократно отбирают наиболее релевантные и полезные примеры, отбрасывая слабые или шумные.

Такой подход позволяет добиться более точных, логичных и соответствующих задачам ответов – особенно в сложных или чувствительных контекстах.

### Как это работает:

**Шаг 1** – собрать широкий набор примеров (например, 10—20 ответов на похожие задачи);

**Шаг 2** – фильтрация, отсеиваются нерелевантные, дублирующие, поверхностные или некорректные.

**Шаг 3** – оценка оставшихся: какие примеры действительно помогли бы модели понять задачу?

**Шаг 4** – повторная фильтрация. Сохраняются только лучшие (например, 3—5), которые используются как контекст (в Few-shot Prompting).

По мере применения и обратной связи – примеры могут дорабатываться, уточняться или заменяться.

### Пример:

**Задача:** Помочь модели давать точные советы по улучшению презентаций.

Исходный набор примеров:

- Общие советы из интернета – Сильные тезисы из TED-презентаций – Слабые рекомендации без аргументов
- Кейсы с реальными примерами и результатами

После фильтрации остаются 3 примера с конкретными рекомендациями и форматированием по шаблону «до → после», которые модель использует в дальнейшей генерации.

### Когда использовать:

- При построении обучающих наборов для моделей
- В задачах, где требуется высокая точность или соответствие тону
- При создании базы «эталонных» ответов для подражания

### Преимущества:

- Позволяет улучшать результаты без изменения модели – только за счёт качества контекста
- Поддерживает качественное масштабирование промптов (особенно в автоматизированных системах)
- Повышает понятность и применимость генерируемых ответов

**Пример шаблона промпта с использованием метода LENS (итеративная фильтрация)** – подхода, при котором модель поэтапно улучшает выборку обучающих примеров, отбрасывая менее релевантные и усиливая контекст.

**Задача:**

Создать подборку наиболее полезных примеров заголовков для статей о здоровье.

**Промпт-шаблон (LENS – итеративная фильтрация):**

Ты – редактор медиа о здоровье. У тебя есть 7 заголовков. Твоя задача – выбрать сначала топ-4, затем из них – финальные 2 лучших по критериям: ясность, конкретность, кликабельность.

ШАГ 1: Отбери 4 лучших заголовка из 7.

ШАГ 2: Из оставшихся выбери 2 финалиста и объясни свой выбор.

Вот заголовки:

1. «Утренние привычки, которые меняют жизнь»
2. «5 причин пить воду натощак»
3. «Как восстановить сон без таблеток»
4. «Здоровье – это просто»
5. «Что скрывают производители БАДов»
6. «Лучшие упражнения для спины»
7. «Как не заболеть осенью: советы врачей»

**Ожидаемый ответ:**

**ШАГ 1 – Отбор 4 заголовков:**

2. «5 причин пить воду натощак»
3. «Как восстановить сон без таблеток»
6. «Лучшие упражнения для спины»
7. «Как не заболеть осенью: советы врачей»

*Комментарий:* Эти заголовки конкретны, несут пользу, содержат цифры и ключевые слова (сон, упражнения, осень, вода).

**ШАГ 2 – Финальный выбор 2 заголовков:**

3. «Как восстановить сон без таблеток»
7. «Как не заболеть осенью: советы врачей»

*Обоснование:* Обе темы актуальны, легко масштабируются в полезные статьи, содержат элемент практической пользы и высокий поисковый потенциал.

В этом примере LENS используется как **двухшаговая воронка отбора**: сначала широкая фильтрация, затем уточнение. Такой шаблон подходит для:

- отбора обучающих примеров
- построения датасетов
- выбора тональности, цитат, шаблонов
- генерации контента «по качеству»

## UDR (Умное извлечение данных)

UDR – это метод интеллектуального подбора информации и примеров, основанный на векторных встраиваниях (embeddings), которые позволяют находить семантически близкие данные – не по ключевым словам, а по смыслу.

В отличие от простого поиска по ключевым словам, UDR использует семантическое сходство (многомерные числовые представления), а затем находит те, которые наиболее близки по значению, даже если формулировки разные.

### Как работает:

- Все элементы базы (примеры, документы, фрагменты текста) преобразуются в векторное пространство.
- Пользовательский запрос тоже переводится в вектор.
- Система рассчитывает косинусное сходство между запросом и базой.
- Возвращаются те элементы, которые максимально близки по смыслу.

### Пример:

#### Запрос:

«Как сократить тексты, сохранив суть?»

Модель не ищет буквально слово «сократить», а находит семантически схожие примеры:

- «Сжать статью без потери смысла»
- «Как сделать резюме текста»
- «Упрощение длинных описаний для презентации»

Все эти примеры могут быть возвращены как релевантные, несмотря на разную формулировку.

### Где применяется:

- В системах RAG (Retrieval-Augmented Generation)
- В чат-ботах, которые опираются на базу знаний
- При генерации ответов с учётом контекста из документов
- Для подбора обучающих примеров в prompting-сценариях (например, Few-Shot + Embeddings)

### Преимущества:

- Позволяет работать не по точному совпадению, а по смысловой близости
- Идеально для объёмных баз знаний и сложных запросов
- Повышает точность и уместность выдачи, даже при расплывчатых формулировках

**Пример шаблона промпта с использованием метода UDR (умное извлечение данных) – подхода, при котором модель находит примеры, максимально близкие по смыслу к текущей задаче, используя семантическое сходство.**

#### Задача:

Проанализировать отзыв клиента и классифицировать его как положительный, нейтральный или отрицательный, используя **наиболее релевантные по смыслу примеры** из базы.

### **Промпт-шаблон (UDR – умное извлечение):**

Ты – система анализа тональности. Найди в базе 3 примера, семантически похожих на целевой отзыв. Затем на их основе классифицируй отзыв и объясни своё решение.

Целевой отзыв:

«Интерфейс симпатичный, но приложение всё время зависает и тормозит. Уже жалею, что скачал.»

База примеров:

1. «Очень доволен, работает стабильно, ничего не тормозит.»  
→ Положительный
2. «Пользоваться можно, но есть мелкие баги и подтормаживает.»  
→ Нейтральный
3. «Неплохо, но интерфейс мог быть получше.» → Нейтральный
4. «Сильно глючит, теряет данные, удаляю.» → Отрицательный
5. «Красиво, но тормозит и часто зависает.» → Отрицательный
6. «Приложение отличное, но иконка не нравится.» → Положительный

**Ожидаемый ответ:**

**Найденные релевантные примеры (по смыслу):**

2. «Пользоваться можно, но есть мелкие баги и подтормаживает.»  
→ Нейтральный
4. «Сильно глючит, теряет данные, удаляю.» → Отрицательный
5. «Красиво, но тормозит и часто зависает.» → Отрицательный

**Классификация целевого отзыва: Отрицательный**

**Обоснование:** Семантически ближе к примерам 4 и 5 – есть негатив по стабильности и сожаление о скачивании. Нейтральный пример менее релевантен из-за мягкой формулировки.

**UDR-подход** особенно полезен в:

- классификации отзывов, заявок, писем
- генерации ответов по аналогии
- подборе инструкций или рекомендаций из базы знаний
- анализе ошибок, логов, пользовательских сценариев

## Active Example Selection

Active Example Selection – это метод, при котором система динамически подбирает примеры для языковой модели, опираясь на результаты предыдущих взаимодействий. Метод основан на принципах обучения с подкреплением (Reinforcement Learning): примеры оцениваются по их влиянию на результат, стратегия выбора постепенно уточняется.

### Как это работает

- Система изначально выбирает случайные или предполагаемо релевантные примеры.
- После каждого ответа модели фиксируется обратная связь (оценка человеком или метрикой).
- Алгоритм обновляет стратегию – например, начинает чаще использовать те примеры, после которых были высокие оценки.
- Со временем выбор становится всё более точным и эффективным.

### Пример:

Представьте, что модель помогает составлять деловые письма. Сначала она получает примеры разного стиля – короткие, формальные, с юмором. Пользователь чаще выбирает деловые и лаконичные. Система это «запоминает» и в следующих сессиях чаще предлагает примеры в этом стиле, отсекая лишнее.

### Где применяется:

- В интеллектуальных системах подбора примеров (например, адаптивные ассистенты)
- В автообучающихся интерфейсах генерации промптов
- При создании систем с долгосрочной памятью или контекстом

### Преимущества

- Учитывает реальную обратную связь, а не предположения
- Повышает персонализацию – подстраивается под стиль пользователя
- Оптимизирует выборку примеров по результату, а не только по сходству

**Пример шаблона для метода Active Example Selection – генерация описаний товаров для интернет-магазина.**

### Задача:

Сгенерировать привлекательное описание товара (рюкзак) для карточки интернет-магазина. Модель **выбирает лучшие примеры описаний**, пробует применить, получает обратную связь и улучшает выбор.

### Промпт-шаблон (Active Example Selection):

Ты – копирайтер для e-commerce. Используй подход Active Example Selection:

**Шаг 1.** Изучи 5 примеров описаний. Выбери 2 наиболее удачных, которые, по твоему мнению, помогут создать цепляющее описание для нового товара.

**Шаг 2.** На основе выбранных примеров напиши описание рюкзака «Urban Pro X 25L».

**Шаг 3.** Получи обратную связь: если описание неудачное – объясни ошибку и выбери другую пару примеров для следующей итерации.

**Примеры описаний:**

1. «Этот минималистичный рюкзак подойдёт тем, кто ценит стиль и практичность. Подходит для города и путешествий.»
2. «Компактный и вместительный. Отделение для ноутбука 15» и водоотталкивающая ткань делают его идеальным для офиса и учёбы.»
3. «Просто рюкзак. Без лишнего. Делает своё дело.»
4. «Яркий дизайн, много карманов, отличная посадка – ваш идеальный спутник каждый день.»
5. «Классический городской рюкзак с эргономичной спинкой и мягкими лямками. Продуман до мелочей.»

**Новый товар:**

**Urban Pro X 25L**

Чёрный рюкзак для города. Объём – 25 литров. Водоотталкивающая ткань, USB-порт, скрытый карман, отделение для ноутбука 16», стильный, мужской дизайн.

**Ожидаемый ответ:**

**Шаг 1 – Выбор примеров:**

- 2 – упор на практичность, офисный стиль
- 5 – акцент на комфорт и детали

**Шаг 2 – Описание:** «Urban Pro X 25L – рюкзак для тех, кто ценит практичность и комфорт. Отделение для 16» ноутбука, скрытые карманы, USB-порт и эргономичная спинка делают его отличным выбором для города и поездок.»

**Шаг 3 – Обратная связь: Недостаточно ярко. Не цепляет визуально.** Описание полезное, но скучное. Недостаёт энергии и эмоционального окраса.

**Коррекция:** Выбраны новые примеры:

- 1 – стиль + универсальность
- 4 – эмоциональный и образный стиль

**Этот подход идеально подходит, если ты работаешь с генерацией:**

- описаний товаров
- рекламных слоганов
- заголовков, постов, писем

## **Вывод**

Продвинутые техники Few-Shot Prompting делают модель более адаптивной и точной, особенно при ограниченных данных и времени. Это особенно важно в условиях ограниченных данных и ресурсов, когда требуется быстро настроить модель на конкретную задачу. Использование тщательно подобранных примеров повышает точность, стабильность и предсказуемость работы модели.

## Написание промптов с нулевым контекстом (Zero-Shot Prompting)

### Как давать точные инструкции без примеров?

Один из наиболее доступных и универсальных способов взаимодействия – это Zero-Shot Prompting. Он не требует подготовки обучающих примеров и опирается исключительно на инструкцию, которую получает модель. В этом подходе важны не данные, а формулировка: насколько ясно, точно и логично вы поставили задачу.

Zero-Shot работает особенно хорошо, когда:

- задача достаточно понятна сама по себе,
- требуется оперативный ответ,
- или нет возможности подбирать Few-Shot примеры.

Несмотря на свою простоту, Zero-Shot Prompting способен решать широкий спектр задач – от генерации текстов и перевода до классификации и анализа – при условии, что инструкция написана грамотно.

### Основные принципы эффективного Zero-Shot Prompting:

**Чёткость.** Формулировка задачи должна быть максимально конкретной. Модель не читает между строк – она следует указаниям буквально.

**Контекст.** Чем больше релевантных деталей вы укажете, тем выше точность ответа. Даже в Zero-Shot важно задать нужный фон: кто, что, для кого, в каком формате.

**Простота.** Избегайте сложных, двусмысленных или перегруженных конструкций. Чем прямей инструкция – тем лучше модель её выполнит.

В этой главе мы рассмотрим практику написания Zero-Shot промптов, разберём типичные ошибки, дадим готовые шаблоны и покажем, как добиться максимальной эффективности даже без примеров. Этот подход станет вашей базой – отправной точкой для освоения более сложных техник, таких как Few-Shot, Chain of Thought или Role Prompting.

## Role Prompting

Role Prompting – это приём, при котором моделью заранее задаётся определённая роль (например, «бизнес-эксперт», «маркетолог», «психолог», «учитель» и т.д.). Это помогает сформировать нужный стиль, тональность и глубину ответов, сделать их более контекстуальными и полезными.

### Как это работает:

В начало промпта добавляется установка вроде:

- «Ты – опытный финансовый консультант...»
- «Представь, что ты – HR-директор с 10-летним стажем...»
- «Ты – преподаватель, объясняющий тему студенту первого курса...»

После этого – формулируется сам запрос.

Модель будет ориентироваться на заданную роль при генерации ответа: выбирать соответствующую лексику, стиль общения, структуру объяснения и даже приоритет информации.

### Пример:

#### Промпт без роли:

Объясни, как работает блокчейн.

#### Промпт с ролью:

Ты – преподаватель колледжа. Объясни, как работает блокчейн, понятным языком для студентов первого курса.

**Результат:** второй ответ оказывается проще, с аналогиями и без лишнего жаргона.

### Где применяется:

- В обучающих и наставнических сценариях
- В маркетинге, продажах и консультациях (роли: эксперт, клиент, оппонент и т.д.)
- При создании чат-ботов, ИИ-ассистентов, диалогов
- Для генерации текстов в нужной тональности

### Преимущества:

- Повышает точность и уместность ответа
- Управляет стилем, глубиной и подходом модели
- Делает ответы более персонализированными и понятными для целевой аудитории

**Пример-шаблона для Role Prompting, где модель «входит в роль», чтобы дать более контекстный и осмысленный ответ.**

#### Задача:

Сгенерировать рекомендации по продвижению онлайн-курса.

#### Промпт-шаблон (Role Prompting):

Ты – опытный digital-маркетолог с 10-летним стажем в сфере онлайн-образования. Твоя задача – дать рекомендации по продвижению нового онлайн-курса на тему «Психология уверенности».

Опиши:

1. Целевую аудиторию

2. Каналы продвижения
3. Типы контента
4. Потенциальные офферы
5. Основные ошибки, которых стоит избежать

**Ожидаемый ответ (в роли эксперта):**

**1. Целевая аудитория:** Женщины 25—45 лет, интересующиеся саморазвитием, психологией, коучингом. Особенно – те, кто сталкивается с тревожностью или синдромом самозванца.

**2. Каналы:** ВК, Дзен, YouTube Shorts, email-рассылки, вебинары, коллаборации с блогерами в сфере self-help.

**3. Контент:**

- Микроистории преобразования
- Советы из курса
- Тесты типа «Узнай свой уровень уверенности»
- Цитаты + экспертные видео-ответы

**4. Офферы:**

- Бесплатный мини-курс
- Диагностическая сессия
- Скидка для подписчиков Telegram-канала

**5. Ошибки:**

- Слишком абстрактный месседж
- Отсутствие чёткой пользы в контенте
- Сухие рекламные тексты без эмоций

**Этот шаблон Role Prompting можно адаптировать под любые роли:**

- UX-дизайнер
- Юрист
- Детский психолог
- Финансовый консультант
- Персональный тренер
- HR-менеджер и др.

## Style Prompting

Style Prompting – это техника, при которой модель явно инструктируют писать в определённом стиле, тоне или настроении. Этот метод помогает задать нужный тон: формальный отчёт, дружеское письмо, юмористический текст или вдохновляющая речь.

### Как это работает:

В запрос добавляется формулировка, задающая стиль ответа. Например:

«Напиши текст в деловом стиле...»

«Сделай это описание простым и дружелюбным...» «Сформулируй с лёгким юмором и ноткой самоиронии...»

«Оформи как официальное письмо от компании клиенту...»

Модель учитывает эту установку и адаптирует лексику, структуру и тональность результата.

### Пример:

Промпт:

Опиши, как работает облачное хранилище.

Вариант 1 (формальный):

Облачное хранилище – это удалённый сервер, обеспечивающий хранение и доступ к данным через интернет с высокой степенью надёжности и масштабируемости.

Вариант 2 (дружелюбный):

Представьте себе огромный виртуальный чемодан, куда вы можете сложить файлы и получить к ним доступ с любого устройства, где бы вы ни были.

Вариант 3 (юмористический):

Облачное хранилище – это как запасной жёсткий диск, только вы его не теряете, не роняете и не проливаете на него кофе.

### Где применяется:

- Копирайтинг и генерация контента под целевую аудиторию
- Маркетинг, PR, блогинг, сценарии видео и соцсетей
- Персонализация общения в чат-ботах
- Образовательные тексты под разный уровень восприятия

### Преимущества

- Делает тексты более точными по тону и адаптированными к ситуации
- Повышает вовлечённость и лояльность аудитории
- Ускоряет создание разностильного контента для одного и того же запроса

**Пример шаблона для Style Prompting, где модель должна писать в заданном стиле, адаптируя тональность и лексику к контексту.**

**Задача:**

Сообщить о задержке доставки клиенту интернет-магазина.

**Промпт-шаблон (Style Prompting):**

Напиши сообщение клиенту об извинении за задержку доставки, используя один из следующих стилей (варианты ниже). Сохраняй суть, но меняй тон:

1. Официальный стиль
2. Дружеский стиль
3. Юмористический стиль

Текстовое ядро:

«Заказ №24871 задерживается из-за сбоя на складе. Мы приносим извинения и сделаем всё возможное, чтобы ускорить доставку.»

### **Ожидаемые стилизованные ответы:**

**1. Официальный стиль:** Уважаемый клиент, Сообщаем, что доставка заказа №24871 задерживается в связи с техническими сложностями на складе. Приносим извинения за доставленные неудобства. Мы предпринимаем все меры для скорейшего разрешения ситуации.

С уважением, Служба поддержки

**2. Дружеский стиль:** Привет! Хотим сразу сказать – заказ №24871 чуть задерживается из-за сбоя на складе Мы уже всё чиним и постараемся доставить как можно быстрее. Спасибо за понимание – вы с нами, и это круто

**3. Юмористический стиль:** Ну вот, наш склад решил поиграть в прятки с вашим заказом №24871 Но не волнуйтесь – мы его уже нашли и возвращаем в реальность! Скоро приедет. Спасибо, что относитесь с юмором – мы вас не подведём

**Такой шаблон идеально подходит для:**

- Писем
- Уведомлений
- Рекламных текстов
- Постов в соцсетях
- Сценариев общения в чат-ботах

## Emotion Prompting

Emotion Prompting – это техника, при которой модель получает инструкцию не только о содержании, но и об эмоциональной окраске ответа. Метод позволяет управлять эмоциональным тоном текста: сделать его вдохновляющим, сочувственным, тревожным, радостным и т. д.

### Как работает:

В промпт добавляется указание на желаемую эмоцию или эмоциональный тон. Например:

- «Опиши это с волнением и восторгом...»
- «Сформулируй с сочувствием и теплотой...»
- «Расскажи с иронией и лёгким раздражением...»
- «Объясни с воодушевлением, как будто мотивируешь аудиторию»

Модель адаптирует лексику, ритм, эмоциональные слова и построение фраз под заданный тон.

### Пример:

Запрос:

Объясни, зачем людям ставить цели в жизни.

Вариант 1 (нейтрально):

Цели помогают определить направление действий, расставить приоритеты и оценивать прогресс.

Вариант 2 (эмоционально, вдохновляюще):

Цели – маяки в океане повседневности. Без них мы дрейфуем. А с ними – движемся, растём, свершаем невозможное. Сначала переписывает промпт,

Вариант 3 (с ноткой тревоги):

Когда у человека нет цели, каждый день становится копией предыдущего.

Появляется ощущение пустоты и потери смысла.

### Где применяется:

- В мотивационном и образовательном контенте
- В рекламных сообщениях и сторителлинге
- При написании писем, обращений, выступлений
- В empathetic AI – чат-ботах с эмоциональной чуткостью

### Преимущества:

- Позволяет точно передавать нужное настроение
- Повышает вовлечённость и эмпатию со стороны аудитории
- Делает тексты более живыми, убедительными и запоминающимися

**Шаблон для Emotion Prompting – когда модель должна включить конкретную эмоцию (радость, вдохновение, тревогу, решимость и т.д.) в ответ.**

### Задача:

Сформулировать сообщение от руководителя команды перед запуском нового проекта, но в **заданной эмоциональной окраске.**

### Промпт-шаблон (Emotion Prompting):

Ты – руководитель команды. Напиши короткое сообщение своей команде перед стартом важного проекта. Сформулируй одно и то же сообщение в разных эмоциональных стилях:

1. Воодушевляющее
2. Спокойное и ободряющее
3. Срочное и тревожное
4. Торжественное и гордое
5. Ироничное, но поддерживающее

Сохрани смысл: «Проект стартует завтра. У нас мало времени, но много потенциала. Верю в вас.»

**Ожидаемые ответы:**

**1. Воодушевляющее:** Завтра стартуем! Впереди вызов – и мы точно справимся. Вы – лучшая команда, и этот проект мы сделаем круто

**2. Спокойное и ободряющее:** Проект начинается завтра. Всё под контролем. У нас есть силы, опыт и поддержка друг друга. Просто делаем шаг за шагом.

**3. Срочное и тревожное:** Завтра старт, времени почти нет. Придётся включаться на максимум с первого дня. Держим фокус, не расслабляемся.

**4. Торжественное и гордое:** Завтра начинается новый этап. Этот проект – шанс показать, на что мы способны. Мы идём не просто работать – мы создаём результат.

**5. Ироничное:** Завтра начинаем – да-да, опять. Мало сна, много дедлайнов, но зато куча адреналина. Заряжаемся и вперёд

**Этот шаблон легко адаптируется для:**

- Писем
- Мотивационных речей
- Постов в соцсетях
- Рекламы
- Сценариев чат-ботов

## System 2 Attention (S2A)

System 2 Attention (S2A) – это подход, при котором модель сначала переписывает промпт, убирая лишнюю или вводящую в заблуждение информацию, а уже затем – формирует на него ответ. Это позволяет повысить точность и сфокусировать модель на сути запроса.

Название отсылает к концепции «Системы 2» из когнитивной психологии (по Даниэлю Канеману) – более медленного и осознанного типа мышления.

### Как работает:

Пользователь вводит запрос, содержащий как полезную, так и лишнюю информацию. Например:

«Слушай, мне нужно срочно объяснение, но я вообще не шарю в этом – что такое GPT, можешь как-то просто рассказать?»

Модель сначала переписывает этот запрос, оставляя только суть:

«Объясни, что такое GPT простыми словами.»

После этого даётся чёткий и релевантный ответ.

### Пример использования в промпте:

Ты – продвинутый помощник.

**Шаг 1:** перепиши мой вопрос, убрав лишнее.

**Шаг 2:** дай точный и ясный ответ. Это очень раздражает!

### Когда применять:

- Когда пользователи пишут длинные и неструктурированные запросы
- В интерфейсах ИИ-ассистентов и поддержки, где важна ясность и фокус
- В системах, где промпты собираются автоматически или из речи

### Преимущества:

- Повышает понятность и точность ответа
- Убирает «шум» из запроса, фокусируясь на сути
- Полезен при взаимодействии с начинающими пользователями

**Пример-шаблон для техники System 2 Attention (S2A) – когда модель сначала переписывает промпт, очищая его от лишнего, а затем отвечает уже на уточнённый и сфокусированный запрос.**

### Задача:

Ответить на перегруженный, путанный запрос пользователя – сначала упростив его.

### Промпт-шаблон (System 2 Attention):

Ты – ассистент, который сначала упрощает запрос, убирая из него всё лишнее, эмоциональное или не относящееся к сути. Затем отвечаешь только на суть вопроса.

**ШАГ 1:** Перепиши запрос в краткой и чёткой форме.

**ШАГ 2:** Ответь на него максимально ясно.

Исходный запрос:

«Я уже полдня бьюсь над этим Excel, у меня всё плывёт в таблице, формулы какие-то странные, и я просто хочу понять – как вообще сделать так, чтобы ячейки тянулись по порядку, без сбоев? Это просто сводит с ума!»

**Ожидаемый ответ:**

**ШАГ 1 – Упрощённый запрос:** Как сделать так, чтобы в Excel ячейки автоматически заполнялись по порядку без сбоев?

**ШАГ 2 – Ответ:** Выделите начальную ячейку, введите первое значение (например, 1), затем в следующей ячейке – следующее (2). Выделите обе и протяните вниз за угол – Excel автоматически продолжит последовательность.

**Такой шаблон отлично работает для:**

- Хаотичных, эмоциональных, неполных запросов
- Пользовательских обращений в поддержку
- Инструкций и ответов на форумы
- Упрощения «шумных» входов в ИИ-системы

## **Вывод**

Zero-Shot Prompting полезен, когда задача достаточно ясна сама по себе, а модель уже имеет необходимые знания из предобучения. Он позволяет получать быстрые и качественные ответы без подготовки примеров, что делает этот подход удобным для универсальных, стандартных или справочных запросов. Однако для сложных задач, где важны структура и контекст, эффективнее применять Few-Shot или другие техники.

## Дополнительные техники Zero-Shot Prompting

### Как повысить качество генерации без обучающих примеров?

Большинство пользователей ассоциируют Zero-Shot Prompting с простыми командами в духе «Напиши...», «Объясни...» или «Сделай...». Однако этот подход значительно богаче, чем кажется на первый взгляд. При правильной подаче запроса модель может решать даже сложные задачи без каких-либо обучающих примеров – если промпт сформулирован точно, логично и структурировано.

В этой главе мы рассмотрим дополнительные техники Zero-Shot Prompting, которые позволяют существенно повысить точность, релевантность и стабильность ответов, опираясь исключительно на внутренние знания модели. Такие методы особенно полезны, когда:

- задача новая или нестандартная,
- нет возможности подготовить примеры,
- требуется мгновенная реакция от модели,
- или важно получить универсальное решение без привязки к предыдущим данным.

В фокусе – управление стилем, ролями, логикой мышления и структурой ответа. Вместо просто «попросить» модель – мы научимся направлять её поведение, используя роли, эмоции, уточняющие шаблоны, переосмысление запроса и другие продвинутые практики.

## SimToM

Моделирование мышления нескольких субъектов на основе заданных фактов

SimToM – это техника промптинга, которая используется для обработки сложных вопросов, где задействовано несколько участников, точек зрения или переменных. Модель при этом опирается только на явно заданные факты, не додумывает и не выносит предположений вне рамок входных данных.

### Как работает:

- В промпте заранее формулируется набор известных фактов или утверждений.
- Модель получает задачу анализировать ситуацию строго на их основе, не привлекая внешние знания.
- Ответ строится логично, поэтапно, с учётом всех заявленных позиций.

### Пример:

#### Факты:

- Алиса думает, что коробка с мячом стоит в комнате.
- Боб знает, что мяч переместили на кухню.
- Алиса не знает, что мяч перенесли.

**Вопрос:** Где, по мнению Алисы, находится мяч?

Инструкция модели: Ответь, основываясь только на перечисленных фактах.

Ответ: «По мнению Алисы, мяч находится в комнате.»

### Где применяется:

- В задачах, связанных с моделированием убеждений, намерений, знаний разных субъектов
- В анализе сценариев и конфликтов (переговоры, психология, философия)
- В обучении логике, критическому мышлению и этике
- В разработке AI-агентов, которым нужно учитывать состояние знаний других агентов

### Преимущества:

- Повышает строгость логики в рассуждениях модели
- Исключает галлюцинации и «домыслы» за пределами заданного контекста
- Учит работе с ограниченной и асимметричной информацией

**Пример-шаблон для техники SimToM (Simulation-to-Model) – подхода, при котором модель отвечает на вопрос, строго опираясь на заранее заданный набор фактов, не дополняя их вымыслом или предположениями.**

#### Задача:

Проанализировать взаимодействие персонажей на основе конкретных известных фактов, без домысливаний.

#### Промпт-шаблон (SimToM):

Используй только перечисленные факты. Не добавляй домыслов или внешних знаний. На основе этих фактов – ответь на вопрос о взаимодействии персонажей.

**Факты:**

1. Анна не доверяет Максиму после инцидента с отчётом.
2. Максим старается наладить отношения, но избегает прямых разговоров.
3. Анна консультируется с Олегом по ключевым вопросам.
4. Олег дружит с Максимом, но нейтрален в рабочих конфликтах.
5. Анна и Олег вместе предложили новую структуру отчётности.
6. Максим не участвовал в этом обсуждении.

**Вопрос:**

Почему Анна не включает Максима в обсуждение новой структуры и как это влияет на динамику в команде?

**Ожидаемый ответ:**

Анна не включает Максима в обсуждение, поскольку не доверяет ему после инцидента с отчётом (факт 1). Она предпочитает работать с Олегом, с которым у неё сложились рабочие отношения (факт 3 и 5). Максим, избегая прямого диалога (факт 2), не пытается восстановить доверие, что усиливает дистанцию между ним и Анной. Это снижает уровень открытого взаимодействия в команде и формирует устойчивую коалицию между Анной и Олегом, где Максим оказывается в стороне (факт 6).

**Шаблон SimToM отлично подходит для:**

- Анализа кейсов (HR, юриспруденция, консалтинг)
- Этических или деловых ситуаций
- Психологических или литературных разборов
- Сюжетного анализа в сценариях/книгах/играх

## Rephrase and Respond (RaR)

Переформулируй – затем отвечай

Rephrase and Respond (RaR) – это техника, при которой модель сначала переписывает исходный запрос своими словами, а затем даёт ответ на уже уточнённую и структурированную формулировку. Такой подход помогает улучшить понимание задачи, особенно если начальный промпт был неясным, разговорным или перегруженным лишними деталями.

### Как работает:

- Пользователь формулирует вопрос в произвольной или неструктурированной форме.
- Модель перефразирует его, чтобы выделить суть и устранить неоднозначности.
- Только после этого формируется финальный, обоснованный ответ.

### Пример:

Промпт пользователя:

Я немного запутался, можешь объяснить, как вообще работают нейросети, ну типа вот эти как GPT?

RaR-подход:

Перефразировано:

«Объясни, как работают языковые нейросети на примере GPT, простым и понятным языком.»

Ответ:

«Языковые нейросети, такие как GPT, обучаются на огромных объёмах текста и затем используют вероятностную модель для генерации логичных продолжений заданных фраз...»

### Где применяется:

- В обучающих продуктах и курсах, где важно прояснение запроса
- В интерфейсах поддержки пользователей (чат-боты, ассистенты)
- В системах, работающих с голосовыми запросами или неструктурированными сообщениями

### Преимущества:

- Повышает ясность и фокусировку запроса
- Снижает риск неправильной интерпретации
- Делает ответ более точным и релевантным
- Позволяет использовать модель в роли «мышления вслух», как в обучении

**Пример-шаблон для техники Rephrase and Respond (RaR) – модель сначала переформулирует вопрос, чтобы чётко понять его суть, а затем даёт структурированный ответ на уточнённую формулировку.**

### Задача:

Улучшить понимание неконкретного или перегруженного вопроса пользователя перед тем, как на него отвечать.

### Промпт-шаблон (Rephrase and Respond):

ШАГ 1: Переформулируй вопрос, сделав его более чётким и фокусным.

**ШАГ 2:** Ответ на него подробно и логично.

**Исходный вопрос:**

«Я хочу, чтобы вы помогли мне разобраться, как вообще строится стратегия контент-маркетинга, потому что вроде все делают посты, статьи, но как из этого получается рост продаж – вообще непонятно.»

**Ожидаемый ответ:**

**ШАГ 1 – Переформулированный вопрос:** Как связана стратегия контент-маркетинга с ростом продаж и какие шаги включает её построение?

**ШАГ 2 – Ответ:** Контент-маркетинг работает как воронка:

1. **Привлечение внимания** – через полезные статьи, посты, видео.
2. **Удержание интереса** – за счёт регулярности, пользы и релевантности контента.
3. **Формирование доверия** – через экспертность, кейсы и отзывы.
4. **Конверсия в продажу** – когда контент подводит к конкретному предложению или решению проблемы. Стратегия строится от целевой аудитории: её боли, интересы, этапы воронки. Контент должен закрывать каждую фазу – от «узнал о вас» до «купил».

**Этот подход отлично помогает:**

- Структурировать запутанные вопросы
- Сделать запросы более «машиночитаемыми»
- Давать чёткие, целевые ответы без лишнего контекста

## Re-reading (RE2)

Повтори прочтение перед ответом – для лучшего понимания

Re-reading (RE2) – это техника, при которой языковой модели явно указывается: прежде чем отвечать, перечитай запрос ещё раз. Такой подход особенно полезен для длинных, многослойных или двусмысленных вопросов, когда требуется более глубокое внимание к деталям.

### Как это работает:

В инструкцию модели добавляется указание:

«Прежде чем ответить – перечитай запрос ещё раз и сформируй чёткое понимание.»

Модель повторно «внимательно» обрабатывает промпт, фиксируя ключевые элементы. После этого она генерирует более точный, логически связный ответ.

### Пример использования:

Ты – эксперт по логике и критическому мышлению.

Прежде чем отвечать на каждый вопрос, перечитай его внимательно, выдели важные части, и только после этого сформулируй точный ответ.

### Где применяется:

- В юридических, технических и научных задачах
- При анализе многоступенчатых или длинных промптов
- В задачах, где ошибка в интерпретации деталей критична

### Преимущества:

- Повышает внимательность модели к ключевым элементам запроса
- Снижает риск генерации поверхностного или поспешного ответа
- Повышает логическую связанность и полноту ответа
- Особенно эффективна при работе с многошаговыми или составными вопросами

**Пример-шаблон для техники Re-reading (RE2) – модель вначале перечитывает вопрос, чтобы лучше уловить смысл и все детали, и только после этого даёт точный, контекстуально полный ответ.**

### Задача:

Ответить на многофакторный, перегруженный вопрос, не упустив важных деталей.

### Промпт-шаблон (Re-reading RE2):

Перед тем как отвечать, внимательно перечитай вопрос. Убедись, что понял все его части и нюансы. Затем дай полный, точный и логично структурированный ответ.

### Вопрос:

«Я веду блог про саморазвитие, аудитория в основном женщины 25—35 лет. Планирую запустить курс по уверенности в себе. Подскажи, как выбрать формат (вебинар или записанные уроки), какие темы стоит затронуть и как вообще протестировать идею до запуска?»

**Ожидаемый ответ:**

**Перечитано. Вопрос состоит из 3 частей:**

1. Выбор формата курса
2. Темы для контента
3. Тестирование идеи до запуска

**Ответ:**

**1. Формат:** Для вашей ЦА (женщины 25—35) лучше подойдут записанные уроки + бонус: живой вебинар с ответами на вопросы. Это даёт гибкость и ощущение личного контакта.

**2. Темы:**

- Причины неуверенности
- Практики повышения самооценки
- Коммуникация без страха
- Упражнения на публичные выступления
- Как сохранять уверенность в стрессовых ситуациях

**3. Тестирование:**

- Проведите опрос в блоге/VK – Запустите бесплатный мини-урок или эфир
- Посмотрите вовлечённость: клики, комментарии, сохранения
- Предложите ранний доступ за символическую цену – это даст реальную обратную связь

**Шаблон RE2 идеально работает для:**

- Консультаций
- Анализа сложных бизнес-запросов
- Образовательных кейсов
- Личных писем и заявок

## Self-Ask

Сначала уточни – потом отвечай

Self-Ask – это техника, при которой модель сама формулирует дополнительные вопросы к исходному запросу, если чувствует, что для точного ответа не хватает информации. После этого она отвечает, учитывая эти уточнения. Это приближает поведение модели к поэтапному человеческому мышлению: сначала разобраться, потом действовать.

### Как работает:

Модель получает сложный или неполный запрос. Прежде чем сразу отвечать, она задаёт себе один или несколько уточняющих вопросов:

«Что нужно знать, чтобы правильно ответить на этот вопрос?»

После самоуточнения – формирует финальный ответ с учётом выявленных факторов.

### Пример:

Запрос:

Объясни, как улучшить продажи.

Self-Ask-подход:

Внутренние уточнения:

– Для какого продукта? – Какая целевая аудитория? – Используются ли сейчас онлайн-каналы?

Финальный ответ с допущениями или общими рекомендациями, учитывающими эти переменные.

### Где применяется:

- В бизнес- и аналитических задачах
- В экспертных консультациях, где часто неполный ввод
- В продуктах, где требуется гибкость мышления (боты, ассистенты, образовательные платформы)

### Преимущества:

- Делает ответы более точными и персонализированными
- Повышает контекстную осведомлённость модели
- Позволяет компенсировать неполноту запроса
- Подходит для задач с множеством переменных и сценариев

**Пример-шаблон для техники Self-Ask – когда модель сначала анализирует исходный вопрос, понимает, достаточно ли информации для ответа, и при необходимости задаёт уточняющие вопросы, прежде чем сформулировать ответ.**

### Задача:

Помочь пользователю выбрать платформу для запуска онлайн-курса.

### Промпт-шаблон (Self-Ask):

Прочитай вопрос. Если в нём недостаточно информации для уверенного ответа, сначала задай 2—3 уточняющих вопроса, а затем – ответь на основной вопрос, с учётом предположений или возможных вариантов.

Вопрос:

«Посоветуй, где лучше запускать онлайн-курс – чтобы удобно было и недорого.»

**Ожидаемый ответ:**

**Уточняющие вопросы:**

1. Какой тип контента вы планируете использовать – видео, тексты, тесты?
2. Важно ли вам наличие встроенной платёжной системы?
3. Курс рассчитан на узкий круг (до 100 человек) или массовую аудиторию?

**Ответ:** Если вам нужен бюджетный старт и вы планируете видеоуроки + базовые тесты, подойдёт **GetCourse** (массово) или **Teachable** (англоязычная, удобная). Если важна интеграция с соцсетями и маркетинг – обратите внимание на **Stepik** или **Thinkific**. Для узкой группы и простоты управления – **Notion + Telegram** с платным доступом тоже может быть решением.

**Техника Self-Ask полезна:**

- В консультациях и поддержке
- При проектировании решений
- Для пользовательских сценариев с неполными вводными
- В B2B и B2C переписке, где часто нужно «довыяснить»

## Вывод

Техники Zero-Shot Prompting особенно эффективны в ситуациях, когда необходимо оперативно решать задачи без примеров, опираясь на внутренние знания и логику модели. Они позволяют добиться высоких результатов при правильно сформулированных инструкциях и хорошо подходят для задач, где важна скорость и универсальность.

Хотя Zero-Shot требует более точной и чёткой формулировки, его гибкость и масштабируемость делают его мощным инструментом в арсенале prompt-инженера. Для достижения наилучших результатов важно экспериментировать с различными техниками, адаптируя подход под конкретную задачу и модель.

В отличие от Few-Shot, Zero-Shot требует более точных инструкций, но при правильном применении может обеспечивать сопоставимые результаты, особенно для современных моделей с обширной базой знаний.

## Генерация мыслей (Chain of Thought, CoT) или логическая цепочка

Когда языковая модель сталкивается с задачей, требующей многоэтапного анализа, простая инструкция «Дай ответ» может привести к ошибке или поверхностному результату. Особенно это заметно в математических задачах, логических головоломках, а также в задачах, где требуется пошаговое рассуждение или объяснение решений.

Именно здесь на помощь приходит техника Chain of Thought – метод, побуждающий модель проговаривать ход своих мыслей, как это делает человек при размышлении вслух. Вместо мгновенного ответа, модель сначала строит логическую цепочку рассуждений, двигаясь от этапа к этапу, пока не придёт к обоснованному выводу.

Этот подход не только повышает точность ответов, но и делает процесс генерации прозрачным, понятным и легко проверяемым. Он особенно полезен в сферах, где важна логика и обоснованность: от образования и программирования до права и бизнес-аналитики.

### **В основе CoT лежат три ключевых принципа:**

– Пошаговое мышление – задача разбивается на отдельные логические шаги, каждый из которых формулируется последовательно.

– Ясность и подробность – модель объясняет свои действия понятным языком, обеспечивая прозрачность рассуждений.

– Контроль промежуточных результатов – каждый шаг проверяется на корректность, прежде чем сформируется финальный ответ.

В этой главе мы разберём, как работает CoT, где он особенно эффективен, и как правильно формулировать промпты, чтобы задействовать этот мощный инструмент.

## Zero-Shot-CoT – логическая цепочка без примеров

Zero-Shot-CoT – это техника, которая объединяет преимущества Zero-Shot и Chain of Thought (CoT). В ней пользователю достаточно явно указать модели мыслить пошагово, даже без предоставления примеров.

### Суть метода:

Вы добавляете к запросу фразу вроде:

«Пожалуйста, рассуждай пошагово.» или «Думай шаг за шагом.»

Эта простая инструкция запускает в модели механизм пошагового анализа, благодаря чему она сама строит цепочку размышлений, даже если не видела примеров до этого.

### Пример:

Промпт:

В корзине лежат 3 яблока. Петя взял 1. Сколько осталось? Пожалуйста, думай пошагово.

Ответ:

В корзине было 3 яблока. Петя взял 1 яблоко.  $3 - 1 = 2$ . Ответ: 2 яблока.

### Когда использовать:

- Для логических, математических или аналитических задач
- Когда нужна интерпретируемость рассуждений
- Если нет возможности сформулировать или подобрать примеры (в отличие от Few-Shot)

### Преимущества:

- Улучшает точность ответов на сложные задачи
- Требуется только 1 дополнительная инструкция
- Подходит даже для кратких промптов
- Повышает «прозрачность» мышления модели

**Универсальный шаблон для Zero-Shot Chain-of-Thought (Zero-Shot-CoT) – техника, при которой модель рассуждает пошагово, не опираясь на обучающие примеры.**

## **Конец ознакомительного фрагмента.**

Текст предоставлен ООО «Литрес».

Прочитайте эту книгу целиком, [купив полную легальную версию](#) на Литрес.

Безопасно оплатить книгу можно банковской картой Visa, MasterCard, Maestro, со счета мобильного телефона, с платежного терминала, в салоне МТС или Связной, через PayPal, WebMoney, Яндекс.Деньги, QIWI Кошелек, бонусными картами или другим удобным Вам способом.