1 ВВЕДЕНИЕ И ОБЩИЕ СВЕДЕНИЯ

Мы написали эту книгу, чтобы помочь инженерам по машинному обучению и дата-сайентистам успешно пройти собеседование по проектированию систем МО. Книга также может пригодиться всем, кто хочет получить общее представление о том, как МО применяется в реальном мире.

Многие технические специалисты полагают, что системы МО исчерпываются такими алгоритмами МО, как логистическая регрессия или нейронные сети. Тем не менее реальные системы МО далеко не ограничиваются разработкой моделей. Эти системы обычно весьма сложны; они состоят из множества компонентов, включая стеки данных, служебную инфраструктуру (благодаря которой система становится доступной миллионам пользователей), пайплайн для оценки ее эффективности, а также средства мониторинга, которые следят за тем, чтобы качество модели не ухудшалось со временем.

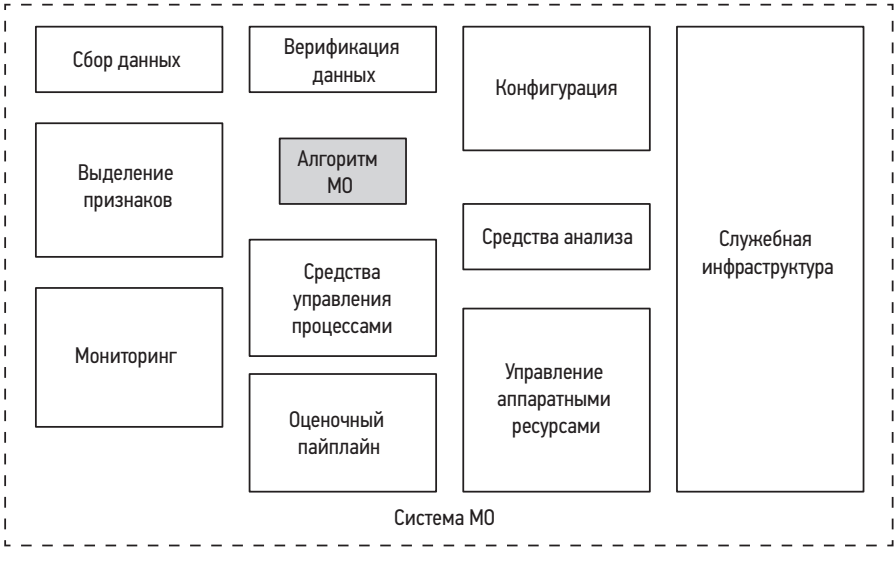

Рис. 1.1. Компоненты системы МО, готовой к эксплуатации

Скорее всего, на собеседовании по проектированию систем МО вам предстоит отвечать на вопросы открытого типа. Например, вам могут предложить спроектировать систему для рекомендаций фильмов или службу поиска видео. У таких задач нет единственно правильного решения. Эксперт, проводящий собеседование,

хочет посмотреть, как вы размышляете, глубоко ли понимаете различные темы из области МО, умеете ли проектировать комплексные системы и находить компромиссы между конфликтующими факторами, которые влияют на проектирование.

Чтобы успешно проектировать сложные системы МО, очень важно придерживаться определенной логики. Без структуры сложно разобраться в проектировочных решениях. Здесь мы предлагаем схему, на которую в этой книге будут опираться примеры проектирования систем МО. Схема состоит из семи основных шагов.

1. Прояснение требований
2. Формулировка проблемы как задачи МО
3. Подготовка данных
4. Разработка модели
5. Оценка
6. Развертывание и эксплуатация
7. Мониторинг и инфраструктура

Рис. 1.2. Основные шаги проектирования систем МО

Каждое собеседование по проектированию систем МО отличается от других, потому что вопросы носят открытый характер и не существует универсальных удачных решений. Эта схема помогает упорядочить мысли, но строго следовать ей необязательно. Сохраняйте гибкость. Если эксперта, проводящего собеседование, в первую очередь интересует разработка модели, почти всегда стоит подстраиваться под его запросы.

Давайте подробнее рассмотрим каждый шаг этой схемы.

Прояснение требований

Вопросы на собеседовании по проектированию систем МО обычно намеренно ставятся нечетко, с минимумом информации. Например, вопрос может звучать так: «Спроектируйте систему для рекомендации событий». Прежде всего стоит задать уточняющие вопросы. Но какие именно? Нужны такие вопросы, которые помогут понять конкретные требования. Этот систематизированный список вопросов можно взять за основу.

- **Бизнес-цель.** Если система должна рекомендовать отпускное жилье для бронирования, то цели могут заключаться в том, чтобы увеличить количество бронирований и выручку.

- **Функции, которые должна поддерживать система.** Какие из требуемых функциональных возможностей могут повлиять на проектирование системы МО? Допустим, вам предложено спроектировать систему для рекомендации видео. Вероятно, стоит уточнить, могут ли пользователи ставить рекомендуемому контенту лайки или дизлайки, потому что этими оценками можно размечать обучающие данные.
- **Данные.** Откуда поступают данные? Каков размер датасета? Размечены ли данные?
- **Ограничения.** Какая вычислительная мощность доступна? Будет ли система работать в облаке или на локальном устройстве? Планируется ли, что со временем модель будет автоматически совершенствоваться?
- **Масштаб системы.** Сколько пользователей будет у системы? С каким количеством объектов (например, видеороликов) придется иметь дело? С какой скоростью растут эти показатели?
- **Производительность.** Насколько быстрыми должны быть предсказания? Должна ли система работать в реальном времени? Что важнее — точность или низкая задержка?

Это не исчерпывающий список, но его можно принять за отправную точку. Не забывайте, что могут быть и другие важные аспекты — например, конфиденциальность и этика.

Предполагается, что к концу этого этапа вы согласуете с экспертом рамки системы и требования к ней. Обычно имеет смысл составить список требований и ограничений: это поможет убедиться, что все одинаково представляют себе задачу.

Формулировка проблемы в виде задачи МО

При решении задач МО крайне важно правильно сформулировать проблему. Допустим, эксперт предлагает вам увеличить степень вовлеченности пользователей на платформе видеостриминга. Безусловно, недостаточная вовлеченность — это проблема, но это не задача МО. Таким образом, чтобы решить проблему, ее следует переформулировать в виде задачи МО.

В реальности сначала надо выяснить, действительно ли для решения проблемы нужно МО. Однако на собеседовании по проектированию систем МО логично допустить, что от МО все-таки будет польза. Таким образом, чтобы сформулировать проблему как задачу МО, можно поступить так:

- Определить цель МО.
- Определить входные и выходные данные системы.
- Выбрать подходящую категорию МО.

Определение цели МО

Бизнес-цель может состоять в том, чтобы повысить продажи на 20 % или увеличить степень удержания пользователей. Однако цели не всегда определяются четко, а модель невозможно обучить, просто приказав ей увеличить продажи на 20 %. Чтобы система МО решила задачу, нужно преобразовать бизнес-цель в точно определенную цель МО. Хорошей целью МО будет такая, которую можно решить с помощью моделей МО. Некоторые примеры перечислены в табл. 1.1, а в дальнейших главах встретятся и другие примеры.

Таблица 1.1. Преобразование бизнес-целей в цели МО

Приложение	Бизнес-цель	Цель МО
Приложение для продажи билетов на мероприятия	Повысить продажи билетов	Максимизировать количество регистраций на мероприятия
Видеостриминговое приложение	Повысить степень вовлеченности пользователей	Максимизировать время, которое пользователи проводят за просмотром видео
Система прогнозирования кликов по рекламе	Увеличить количество кликов	Максимизировать CTR (кликабельность)
Выявление вредоносного контента в социальной сети	Сделать платформу безопаснее	Точно предсказывать, является ли контент вредоносным
Система рекомендации друзей	Увеличить темп расширения сети пользователя	Максимизировать количество установленных связей

Определение входных и выходных данных системы

Когда цель МО ясна, нужно определить входные и выходные данные системы. Например, для системы выявления вредоносного контента в социальной сети входными данными является пост, а выходными — решение о том, считать ли его вредоносным.

Иногда система может состоять более чем из одной модели МО. В таком случае требуется определить входные и выходные данные для каждой модели. Например, в примере с выявлением вредоносного контента одна модель может выявлять призыв к насилию, а другая — непристойные изображения. Система опирается на обе эти модели, чтобы решить, считать ли пост вредоносным.

Еще одно важное соображение состоит в том, что может существовать несколько способов определить входные и выходные данные модели — см. пример на рис. 1.4.

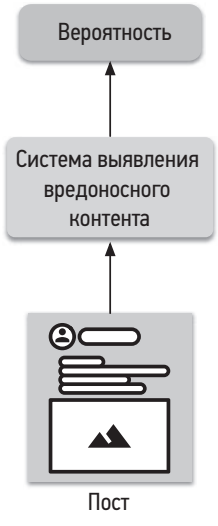

Рис. 1.3. Входные и выходные данные системы выявления вредоносного контента

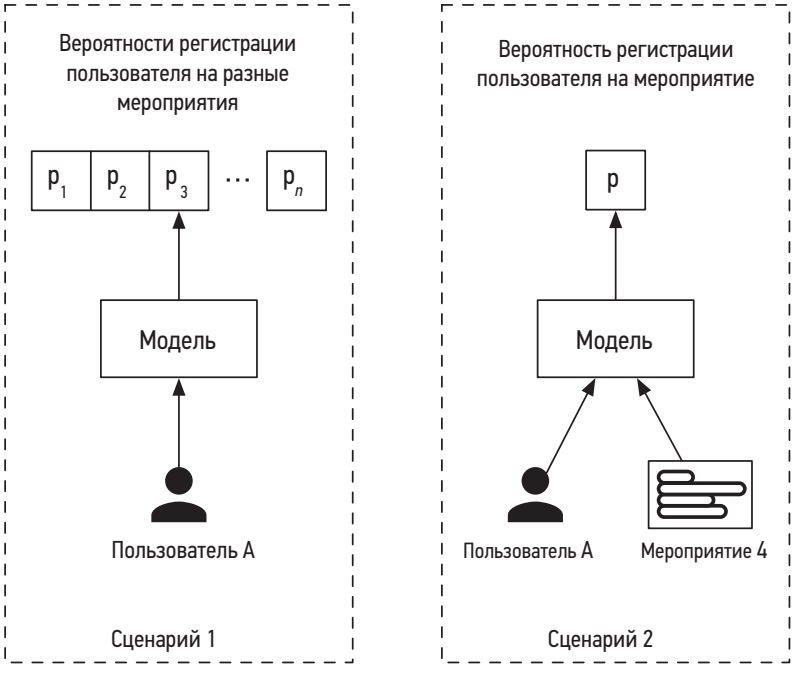

Рис. 1.4. Разные способы определения входных и выходных данных модели

Выбор подходящей категории МО

Существует много способов переформулировать проблему в виде задачи МО. Большинство проблем можно представить так, чтобы они относились к одной из категорий МО, изображенных на рис. 1.5. Поскольку эти категории, вероятно, уже знакомы большинству читателей, мы ограничимся краткой сводкой.

Рис. 1.5. Распространенные категории МО

Обучение с учителем. В этих моделях используется обучающий набор данных. На практике многие проблемы относятся к этой категории, потому что обучение на размеченном наборе данных обычно приводит к лучшим результатам.

Обучение без учителя. Чтобы делать предсказания, такие модели обрабатывают данные, которые не содержат правильных ответов. Цель обучения — выявить осмысленные закономерности в данных. Популярные алгоритмы обучения без учителя — кластеризация, ассоциирование и снижение размерности.

Обучение с подкреплением. Система учится решать задачу, многократно взаимодействуя со средой методом проб и ошибок. Например, таким способом можно научить робота ходить по комнате или натренировать такую программу, как AlphaGo, чтобы она успешно соревновалась с человеком в игре го.

По сравнению с обучением с учителем, обучение без учителя и обучение с подкреплением менее популярны в реальных системах, потому что модели МО обычно лучше обучаются, если есть обучающие данные. Поэтому для большинства проблем, которые рассматриваются в этой книге, применяется обучение с учителем. Давайте поближе познакомимся с разными его видами.

Регрессионная модель. Регрессия предсказывает непрерывное числовое значение — например, ожидаемую стоимость дома.

Классификационная модель. Классификация предсказывает дискретную метку класса — например, следует ли отнести входное изображение к классу «собака», «кошка» или «кролик». Классификационные модели можно разделить на две группы.

- **Бинарная классификация** предсказывает бинарный результат — например, есть на изображении собака или нет.
- **Многоклассовая классификация** разбивает входные данные на несколько классов: например, можно классифицировать объект на изображении как собаку, кошку или кролика.

Предполагается, что на этом шаге вы выберете правильную категорию МО. В следующих главах приводятся примеры того, как выбрать подходящую категорию во время собеседования.

Темы для обсуждения

Вот некоторые из тем, которые могут обсуждаться во время собеседования.

- Что такое хорошая цель МО? Как сравнить между собой разные цели МО? Какие у них плюсы и минусы?
- Какие входные и выходные данные будут у системы для конкретной цели МО?
- Если в системе МО задействованы несколько моделей, каковы входные и выходные данные у каждой из них?
- Как должно проводиться обучение — с учителем или без?
- Какая модель лучше поможет решить проблему — регрессия или классификация? Если используется классификация, то должна ли она быть бинарной или многоклассовой? А если регрессия, то каким должен быть диапазон выходных значений?

Подготовка данных

Модели МО обучаются непосредственно на данных, а это значит, что для обучения чрезвычайно важны данные с высокой предсказательной способностью. Этот раздел посвящен тому, как готовить качественные входные данные для моделей МО с помощью двух основных процессов — инженерии данных (data engineering) и конструирования признаков (feature engineering). Мы рассмотрим важные аспекты того и другого.

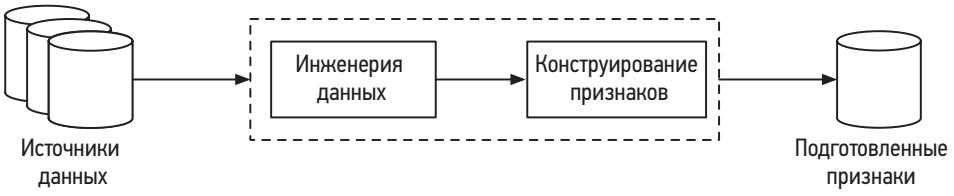

Рис. 1.6. Процесс подготовки данных

Инженерия данных

Инженерия данных заключается в том, чтобы проектировать и строить пайплайны для сбора, хранения, извлечения и обработки данных. Кратко рассмотрим основные принципы инженерии данных, чтобы понять, какие основные компоненты для нее могут понадобиться.

Источники данных

Система МО может работать с данными из многих источников. Полезно разбираться в источниках данных, чтобы отвечать на различные контекстные вопросы, например: кто собирал данные? Насколько они чисты? Можно ли доверять источнику? Данные созданы пользователями или сгенерированы машиной?

Хранилище данных

Хранилище данных (или база данных, БД) — это репозиторий, который позволяет долгосрочно хранить коллекции данных и управлять ими. Для разных сценариев использования применяются разные БД, поэтому важно понимать на высоком уровне, как работают те или иные базы данных. Для собеседования по проектированию систем МО обычно не требуется разбираться во внутреннем устройстве баз данных.

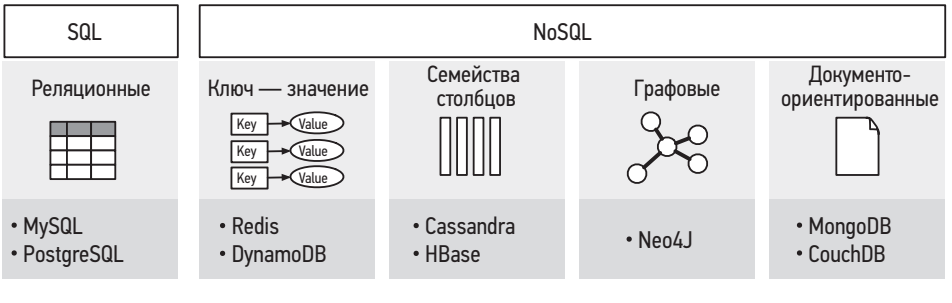

Рис. 1.7. Разные виды баз данных

Извлечение, преобразование и загрузка (ETL)

Процедура ETL (Extract, Transform, Load — «извлечение, преобразование, загрузка») состоит из трех фаз:

- **извлечение:** данные извлекаются из разных источников;
- **преобразование:** в этой фазе данные обычно очищаются, приводятся в порядок и преобразуются в тот формат, который нужен для выполняемых задач;
- **загрузка:** преобразованные данные загружаются в приемник — файл, базу данных или хранилище данных [1].

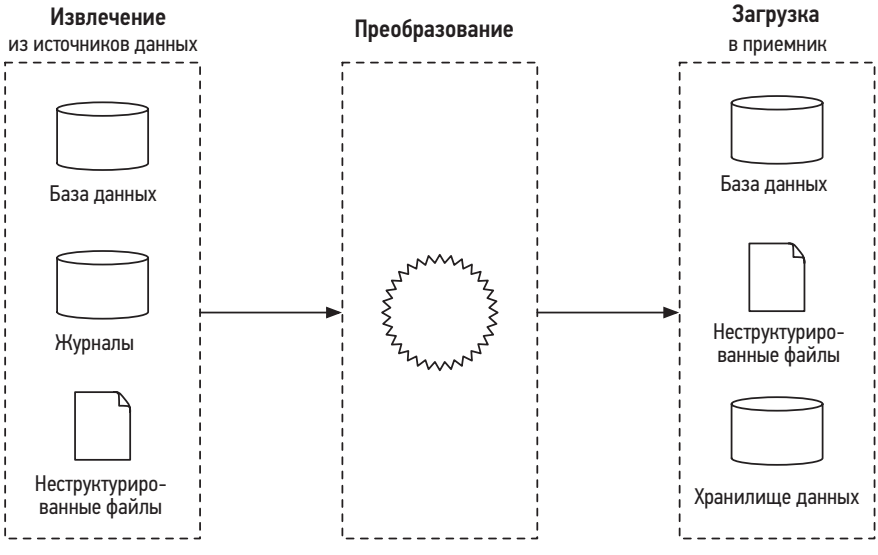

Рис. 1.8. Обзор процесса ETL

Типы данных

Типы данных в машинном обучении отличаются от типов в языках программирования (`int`, `float`, `string` и т. д.). На высоком уровне типы данных можно разделить на две категории: структурированные и неструктурированные (рис. 1.9).

Структурированные данные подчиняются заранее определенной схеме. Например, структурированными данными можно считать даты, имена, адреса, номера кредитных карт и вообще все, что можно представить в табличном формате со строками и столбцами. Неструктурированные данные не соответствуют какой-то конкретной схеме; к этой категории относятся текст, изображения, аудио- и видеозаписи. В табл. 1.2 перечислены основные различия между структурированными и неструктурированными данными.

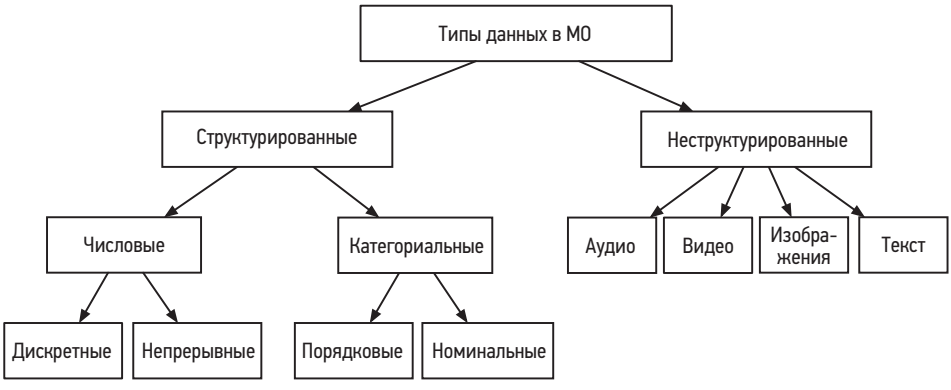

Рис. 1.9. Типы данных в МО

Таблица 1.2. Структурированные и неструктурированные данные

	Структурированные данные	**Неструктурированные данные**
Характеристики	Заранее определенная схема. Просто выполнять поиск	Нет определенной схемы. Трудно выполнять поиск
Место хранения	Реляционные базы данных. Во многих базах данных NoSQL могут храниться структурированные данные. Хранилища данных	Базы данных NoSQL. Озера данных
Примеры	Даты. Телефонные номера. Номера кредитных карт. Адреса. Имена	Текстовые файлы. Аудиофайлы. Изображения. Видео

Как показано на рис. 1.10, для разных типов данных подходят разные модели МО. Важно понимать, структурированы данные или нет, чтобы выбрать подходящую модель МО на шаге разработки модели.

Числовые данные

К числовым данным относятся любые значения, представленные числами. Как показано на рис. 1.9, числовые данные делятся на непрерывные и дискретные. Например, цены на недвижимость можно считать непрерывными, потому что цена может принимать любое значение из соответствующего диапазона. С другой стороны, количество домов, проданных за последний год, можно считать примером дискретных числовых данных, так как оно принимает только целые значения.

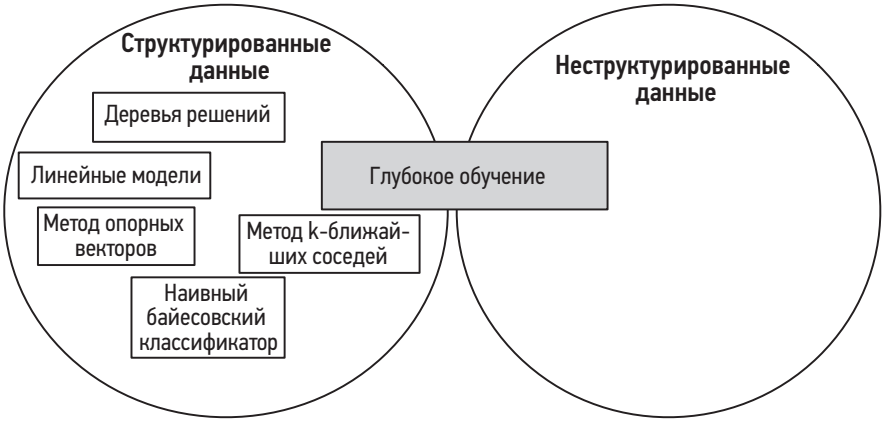

Рис. 1.10. Модели для структурированных и неструктурированных данных (см. [2])

Категориальные данные

Категориальные данные — это данные, которые можно хранить и идентифицировать с помощью присвоенных им имен или меток. Например, пол относится к категориальным данным, потому что его значение берется из ограниченного набора нечисловых значений. Категориальные данные можно разделить на две группы: номинальные и порядковые.

Под номинальными данными понимаются данные, между категориями которых нет числовой связи. Например, сюда относится пол, потому что между значениями «мужской» и «женский» нет числовых отношений. Порядковые данные состоят из значений, между которыми есть заранее определенный или последовательный порядок, — например, сюда относятся оценки с тремя уникальными значениями: «недоволен», «нейтрально» и «доволен».

Конструирование признаков

К конструированию признаков относятся два процесса:

- использование знаний о предметной области, чтобы выбирать и извлекать предсказательные признаки из необработанных данных;
- преобразование предсказательных признаков в формат, пригодный для модели.

Выбрать подходящие признаки — одна из главных задач при разработке и обучении моделей МО. Важно выбрать признаки, которые приносят наибольшую практическую пользу. На этом этапе нужно хорошо знать предметную область, причем процесс также сильно зависит от конкретной задачи. Чтобы помочь вам его освоить, в книге мы приводим многочисленные примеры.