

Содержание

1	Математические основы машинного обучения	5
1.1	Методы оптимизации	5
1.1.1	Градиентный спуск	7
1.1.2	Распределенное обучение	9
1.1.3	Стохастическая аппроксимация	12
1.2	Оптимальный транспорт	18
2	Методы машинного обучения	22
2.1	Нейросетевые подходы	22
2.2	Генеративные подходы	26
2.2.1	Автокодировщики и модели потоков	29
2.2.2	Энергетические модели	32
2.3	Обработка естественного языка	33
2.3.1	Формальные языки	34
2.3.2	Представление языка в виде вложений	35
2.3.3	Большие языковые модели	37
3	Психометрия в образовании	40
3.1	Значение образования	40
3.1.1	Механизм	41
3.1.2	Систем наград	44
3.2	Структура методического материала	46
3.2.1	Психологические подходы к описанию интеллектуального развития	47
3.3	Методическая задача	51
3.3.1	Тестовые системы	51
4	Описание работы	55
4.1	Подготовка данных для обучения	55
4.1.1	Распознавание текста	56
4.1.2	Дополнительные подходы к сбору данных	57
4.2	Разработка интерфейса взаимодействия	59
4.2.1	Проект интеллектуального ассистента	60

4.2.2	Подготовка веб-приложения	63
4.2.3	Описание разработанных игр	64
4.3	Рейтинговая система	65
4.3.1	Адаптация рейтинговой системы	66
4.3.2	Численные эксперименты	70
5	Заключение	75
5.1	Итоги работы	75
5.2	Дальнейшая работа	76
5.2.1	Благодарности	77
	Список литературы	78
5.3	Использованная литература	88
5.4	Численное моделирование	90

Генеративное моделирование в сфере образования активно развивается и является одним из актуальных направлений научной работы. С 2019 года UNESCO регулярно публикует аналитические доклады [1][2], содержащие обзоры применения искусственного интеллекта в образовании. Одним из наиболее востребованных направлений является адаптация интеллектуальных ассистентов для персонального обучения. Доступные технологии распределенных вычислений и обработки больших данных обусловили создание больших языковых моделей, возможности которых в области работы с естественным языком приближаются к человеческим. Такие модели уже широко используются в практических задачах, таких как программирование, анализ предметных корпусов и преобразования экспериментальных данных [3]. Коммерческий сектор также активно использует возможности языковых моделей и принимает участие в их развитии, предоставляя корпусы данных и обученные модели. Модели успешно справляются с задачами перевода, выделения ключевых слов и пересказа. Эти возможности используются для динамического формирования описаний продуктов с учетом интересов потребителей, в чат-ботах служб поддержки и оптимизации информационной выдачи.

Целью данной работы является исследование возможностей больших языковых моделей в контексте задач образования. Область исследования включает анализ преимуществ подхода с использованием языковых моделей и методов компенсации его недостатков. Исходя из цели были поставлены задачи:

- подготовка корпуса образовательных данных на русском языке;
- организация среды эффективной эксплуатации большой языковой модели;
- разработка адаптивного алгоритма подбора сложности заданий для обучаемых.

Для их решения автор проводит предметный обзор актуальных подходов к развитию образовательных технологий и наиболее значимых достижений в области современного машинного обучения. Работа содержит 4 главы.

В первой главе работы представлено теоретическое описание аппарата оптимизации и статистического моделирования. Рассмотрены методы градиентного стохастического спуска, обучения графовых вероятностных моделей и решения транспортных задач. Выполнен анализ ключевых теорем градиентного спуска и стохастической аппроксимации.

Во второй главе рассмотрены современные подходы генеративного моделирования. Описаны техники генерации, включающие вариационный автокодировщик, нормализационные потоки и диффузионные модели. Приведены методы численно эффективной адаптации моделей с ключевыми теоретическими выкладками.

Третья глава посвящена применению психометрии в системе образования. Описаны методы оценки знаний учащихся с помощью байесовых систем тестирования и подходы теории игр к построению системы рейтинга с использованием экономических механизмов. Включены описания этапов детского развития, разработанные

психологами Жаном Пиаже [4] и Львом Выготским [5]. Описана качественная модель выбора сложности заданий согласно Михею Чиксентмихай [6].

В четвертой главе описан порядок исследований и вывод ключевых теорем работы. Приведено описание подхода к сбору данных из открытых источников и количественное описание полученных корпусов. Рассмотрен вариант усовершенствования системы оптического распознавания символов на русском языке. Выполнен синтез оптимального алгоритма спуска к заданному параметру бернуллевого распределения с откликом в виде логистической функции.

В заключении работы выполнен анализ итогов и намечены перспективы дальнейших исследований, включающие подход с использованием стохастической аппроксимации.

В качестве ключевых результатов работы можно выделить:

- разработанный алгоритм адаптации сложности заданий d , обеспечивающий оптимальную сходимость вероятности решения задачи s к методически рекомендованной s^* ;
- интеграция большой языковой модели в систему адаптивного подбора сложности заданий.

Апробация работы была выполнена на конференции МФТИ в форме двух докладов:

- "Оценка влияния кредитных условий на конкурентные предложения малых поставщиков в сфере образования";
- "Разработка пакетного модуля ShuemacherOCR на языке Python для работы с методической литературой".

Глава 1

Математические основы машинного обучения

Математический аппарат искусственного интеллекта был предложен в 60-е годы прошлого века Б. Т. Поляком в теории оптимизации [7], В. В. Наумовичем и А. Я. Червоненкисом [8] в теории минимизации эмпирического риска и Л. В. Кантаровичем в теории оптимального планирования [7]. Современные достижения во многом являются результатом адаптации разработанного теоретического аппарата к применению мощных вычислительных систем, ставших доступными в последнее десятилетие.

В главе приведено описание математического аппарата оптимизации, включающие техники градиентного спуска, стохастической аппроксимации, распределенного обучения и оптимального транспорта.

Также необходимо отметить важность этапа, предшествующего поиску и исполнению оптимального алгоритма оптимизации: определение аналитической функции оптимизации, отвечающей потребностям предметной постановки. Такая функция должна быть скалярной величиной, но может зависеть от многих факторов. Функция должна быть "удобной" для оптимизации и оценки. Обычно используется монотонная функция с гладкими производными, при убывании которой система наблюдаемо изменяет свои свойства. Также важны ограничения на допустимость сочетания параметров, что позволяет снизить вероятность перехода оптимизируемой системы в нежелательный режим.

1.1 Методы оптимизации

Математическая оптимизация это современное и активно развивающееся направление исследований, определяющее выбор оптимальных параметров для максимизации результата. Теория оптимизации широко применяется в экономике, теории игр и физике для разрешения важных практических постановок управления системами и их проектирования. Задачи оптимизации классифицируются по наличию ограничений на аргументы функции, виду отклика и постановке задачи. Для каждой спе-

цифичной постановки, как правило, существует специальная теория, адаптирующая базовый аппарат оптимизации под конкретную задачу. Общими же являются методы, использующие производные оптимизируемой функции для поиска численной схемы, обеспечивающей эффективный поиск экстремума.

Современная задача оптимизации с учетом ошибки наблюдения записывается как [9]:

$$f(x) = \mathbb{E}f(x, \xi) \rightarrow \min_x, \quad (1.1)$$

где пара (x, ξ) является результатом наблюдения с детерминированным параметром x и случайной величиной ξ . В секции для упрощения будут рассмотрены только методы первого порядка.

Определение: Функция f называется L -липшицевой для метрики ρ , если $\forall x, y \rightarrow \rho(f(x), f(y)) \leq L \cdot \rho(x, y)$. Для оптимизации в качестве метрики, как правило, выбирают евклидово расстояние $\rho(x, y) = \|x - y\|^2$, задающее сходимость по вероятности нормального распределения:

$$|f(x) - f(y)| \leq L \cdot (x - y)^2 \quad (1.2)$$

Определение: μ -гладкой называется $f(X)$ функция, для которой выполнено:

$$\forall x_1, x_2 \in S \rightarrow f(\lambda x_1 + (1 - \lambda)x_2) \leq \lambda f(x_1) + (1 - \lambda)f(x_2) - \frac{\mu}{2}\lambda(1 - \lambda)\|x_1 - x_2\|^2, \quad (1.3)$$

где S — выпуклое множество и $\mu > 0$.

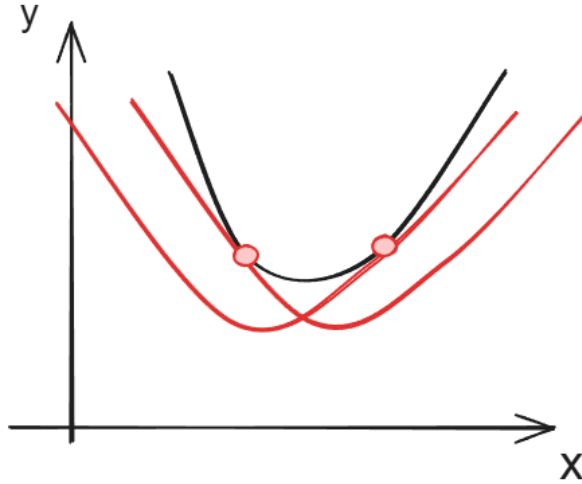


Рис. 1.1: Сильная выпуклость позволяет выполнить квадратичную оценку снизу

Для выпуклых функций справедливо неравенство Йенсена, широко применяющееся для задания оценки снизу на скорость сходимости численной схемы:

$$f\left(\sum_{i=1}^k \alpha_i x_i\right) \leq \sum_{i=1}^k \alpha_i f(x_i), \quad (1.4)$$

причем равенство выполняется только при $x_1 = \dots = x_k$.

Практически важным для выпуклой функции является величина ее матожидания:

$$f(\mathbb{E}X) \leq \mathbb{E}f(X) \quad (1.5)$$

Для современных постановок задач с большими данными существенно исключение стохастического шума. Базовым методом оптимизации в условиях случайного отклика является усреднение по подвыборке (*англ.* — batch). 1 **Определение:** **Усредненным по подвыборке** мощности B стохастическим градиентом называется $\nabla^B f(x, \xi) = \sum_{j=i}^B \nabla f(x, \xi_i)$.

Техника усреднения на шаге приобрела популярность с развитием технологий распределенных вычислений, позволяющих выполнять расчет параллельно на нескольких вычислителях. Такая методика позволяет ускорить обучение за счет проведения большего объема вычислений на каждом шаге расчета.

1.1.1 Градиентный спуск

Базовым методом оптимизации является спуск по координатам функции в направлении антиградиента. В условиях достаточной гладкости и выпуклости оптимизируемой функции такой метод позволяет находить глобальный максимум. В практических постановках функции не всегда удовлетворяют заданным условиями, что приводит к затуханию градиента в локальных минимумах. Исследователи разрешают проблему, модифицируя методы за счет использования моментов и усреднения траектории.

Определение: **Метод градиентного спуска** — численный метод нахождения экстремума функции, использующий в схеме градиент оптимизируемой функции:

$$x_{t+1} = x_t - f(\nabla L(x_t)). \quad (1.6)$$

Б. Т. Поляк показал [10], что для функции с L -липшицевым градиентом оптимальной будет разностная схема:

$$x^{k+1} = x^k - \frac{1}{L} \nabla f(x^K) \quad (1.7)$$

В этом случае ошибка будет асимптотически линейно убывать с числом шагов $\frac{1}{N}$:

$$f(x^N) - f(x_*) \leq \frac{LR^2}{N}, \quad (1.8)$$

где $R = \|x^0 - x_*\|_2$.

Докажем этот факт, используя μ -выпуклость и L -гладкость функции.

Теорема: *О простейшей схеме градиентного спуска* Пусть необходимо задать $x_* = \arg \min_x f(x)$, где f — L -гладкая и μ -гладкая, тогда градиентный метод 1.7

имеет линейную скорость сходимости.

Доказательство: Ввиду L -гладкости функции f

$$f(x^{k+1}) \leq f(x^k) + \langle \nabla f(x^k), x^{k+1} - x^k \rangle + \frac{L}{2} \|x^{k+1} - x^k\|_2^2 \quad (1.9)$$

Тогда:

$$f(x^{k+1}) \leq f(x^k) - \frac{1}{2L} \|\nabla f(x^k)\|_2^2 \quad (1.10)$$

С другой стороны из μ -выпуклости получаем:

$$f(x^{k+1}) \leq f(x^k) - \frac{1}{2\mu} \|\nabla f(x^k)\|_2^2. \quad (1.11)$$

Объединяя выражения получаем:

$$f(x^{k+1}) - f(x_*) \leq \left(1 - \frac{\mu}{L}\right) (f(x^k) - f(x_*)). \quad (1.12)$$

Рекурсивное применение неравенства задает:

$$f(x^k) - f(x_*) \leq \left(1 - \frac{\mu}{L}\right)^k (f(x^0) - f(x_*)) \quad (1.13)$$

■

На практике линейная сходимость может быть недостаточна, потому разрабатываются продвинутое методы, выполняющий спуск с асимптотически квадратичным убыванием ошибки. К таким методам относятся методы сопряженных градиентов и метод тяжелого шарика Поляка.

Схема выполнения нового шага по методу сопряженных градиентов выглядит как:

$$x^{k+1} = x^k - \alpha_k \nabla f(x^k) + \beta_k (x^k - x^{k-1}), \quad (1.14)$$

где $(\alpha_k, \beta_k) \in \arg \min_{\alpha, \beta} f(x^k - \alpha \nabla f(x^k) + \beta(x^k - x^{k-1}))$ дает оценку скорости сходимости как:

$$f(x^N) - f(x_*) \leq \frac{LR^2}{N^2} \quad (1.15)$$

Метод тяжелого шарика Поляка учитывает значение с предыдущего шага x^{k-1} . Оптимальная численная схема для μ -выпуклой функции с L -гладким градиентом запишется как:

$$x^{k+1} = x^k - \frac{4}{(\sqrt{L} + \sqrt{\mu})^2} \nabla f(x^k) + \frac{(\sqrt{L} - \sqrt{\mu})^2}{(\sqrt{L} + \sqrt{\mu})^2} (x^k - x^{k-1}) \quad (1.16)$$

В приложении к машинному обучению известны также модификации метода тяжелого шарика:

1. AdaGrad [11] задает адаптивный шаг спуска для каждого параметра i с учетом

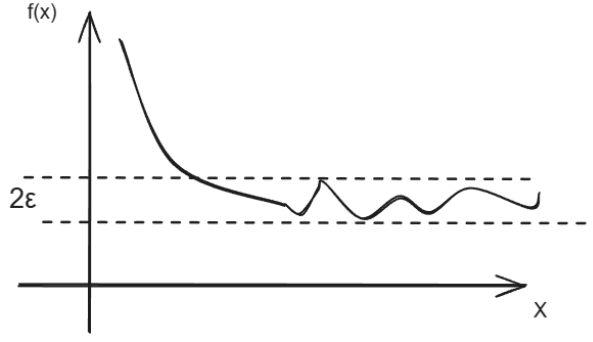


Рис. 1.2: Устойчивость метода при выходе на плато

индивидуального градиента $\nabla_{\theta_i} \mathcal{L}(\theta_i)$:

$$\begin{aligned} x_{t+1} &= x_t - \frac{\eta}{\sqrt{g_t + \epsilon}} \cdot \nabla L(x_t) \\ g_{t+1} &= g_t + \nabla_{\theta} \mathcal{L}(\theta)^2 \end{aligned} \quad (1.17)$$

Итерация начинается с $g_t = 0$. Исследователи отмечают повышение эффективности метода в постановках с функциями распределения параметров с тяжелыми хвостами.

2. RMSProp [12] модификация метода скользящим средним по параметру накопления градиента:

$$\begin{aligned} x_{t+1} &= x_t - \frac{\eta}{\sqrt{g_t + \epsilon}} \cdot \nabla L(x_t) \\ g_t &= \mu g_{t+1} + (1 - \mu) \nabla L(x_t) \cdot \nabla \mathcal{L}(\theta)^2 \end{aligned} \quad (1.18)$$

3. Adam [13] — метод совмещающий скользящее среднее по градиенту и параметру накопления градиента:

$$\begin{aligned} x_{t+1} &= x_t - \frac{\eta}{\frac{\sqrt{g_t + \epsilon}}{1 - \mu^t}} \cdot \frac{v_{t+1}}{1 - \beta^t} \\ v_{t+1} &= \beta v_t + (1 - \beta) \nabla L(x_t) \\ g_t &= \mu g_{t-1} + (1 - \mu) \nabla L(x_t) \cdot \nabla L(x_t) \end{aligned} \quad (1.19)$$

Отметим, что метод не обязательно сходится даже в выпуклой постановке [14]. Тем не менее на практике полученный алгоритм, как правило, обладает наилучшими показателями сходимости.

1.1.2 Распределенное обучение

Существенным препятствием к обучению больших моделей является распределение структурированных данных по множеству локальных устройств, в условиях ограничения доступа к ним по соображениям приватности. Подобные постановки часто встречаются при обучении в сфере медицины. Для ее разрешения вводятся алго-

ритмы распределенного обучения, эффективно использующие локальные ресурсы и обменивающиеся между устройствами защищенными пакетами данных. Принципиально постановка задачи заключается в определении порядка локальных и глобальных пересчетов с оптимальным уровнем компрессии при передаче данных.

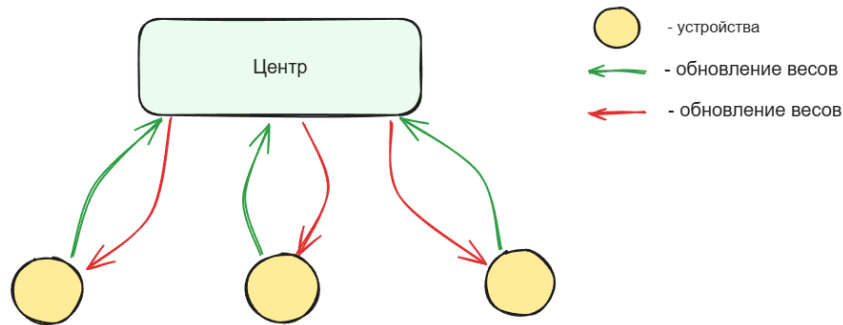


Рис. 1.3: Распределенное обучение включает локальные и глобальные ресурсы, разделяемые раундами коммуникации

Определим виды распределенного обучения:

- кластерное — единая организация с доверенными узлами;
- коллаборативное — добровольное присоединение к процессу обучения с возможностью покинуть вычисление в любой момент времени;
- федеративное — на устройствах пользователей с применением локальных вычислительных мощностей.

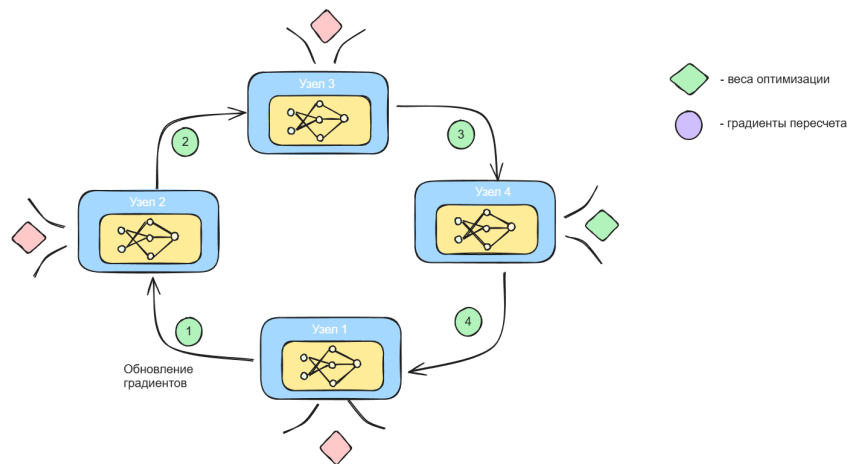


Рис. 1.4: Одной из распространенных архитектур взаимодействия между вычислительными узлами является кольцевая.

Ключевыми вопросами для разработки алгоритма являются определения следующих характеристик:

- числа пересчетов на вычислительных узлах N ;
- объема коммуникаций при вычислениях K ;

- степени компрессии данных при коммуникациях β .

Определение: **Несмещенной компрессией** называется компрессия Π со свойствами

- $\mathbb{E}[\Pi(x)] = x$
- $\mathbb{E}[\|\Pi(x)\|_2^2] \leq \omega \|x\|_2^2$, где $\omega \geq 1$

Рассмотрим современные подходы к компрессии в постановке распределенных систем.

Определение: **Случайной спарсификацией** [15] называется оператор, выбирающий из вектора компоненты по правилу:

$$\text{Rankd}(x) = \frac{d}{k} \sum_{i \in S} x_i e_i, \quad (1.20)$$

где i — случайно выбранные компоненты из базиса.

Определение: **Трехуровневая** \downarrow_2 квантизация задается оператором $Q(x) = \|x\|_2 \text{sign}(x) \xi_i$, $i = 1, \dots, d$ и x_i — бернулевская случайная величина с параметром, определяемым вкладом компоненты в модуль $\frac{|x_i|}{\|x\|_2}$.

Определение: Пусть round_ν^- оператор, округляющий число до ближайшей степени $\nu \in \mathcal{N}$ снизу, round_ν^+ — аналогичный оператор для округления в большую сторону. Тогда **натуральная компрессия** задается оператором:

$$\text{Nat}(x)_i = \begin{cases} \text{round}_2^-(x_i), & \text{с вероятностью } p = \frac{x_i - \text{round}_2^-(x)}{\text{round}_2^+(x) - \text{round}_2^-(x)} \\ \text{round}_2^+(x_i), & \text{by fxt} \end{cases} \quad (1.21)$$

Теорема: [16] Если все функции f_m являются μ -сильно выпуклыми и имеют L -липшицев градиент, тогда при шаге $\eta \leq L^{-1}(\frac{2\omega}{M} + 1)^{-1}$:

$$\mathcal{O} \left((1 - \gamma\mu)^K \|x_0 - x^*\|^2 + \frac{1}{K} \frac{2\omega}{\mu M^2} \sum_{m=1}^M \|\nabla f_m(x^*)\|^2 \right) \quad (1.22)$$

Достижение единого согласованного состояния в распределенных системах обеспечивается с помощью специализированных алгоритмов Raft [17] и Paxos [18]. В основе моделей лежит акторное (actor) разделение, позволяющее распределять данные между вычислительными узлами для обеспечения их сохранности и согласованности.

Определение: **Консенсус** является результатом достижения согласованного состояния между несколькими независимыми процессами или узлами в системе, которые могут взаимодействовать друг с другом.

Для достижения консенсуса необходимо выполнить условия:

1. корректности: $\forall i \in \{1, \dots, n\}$, если $\text{input}(N_i) = v$, то $\forall j \in \{1, \dots, n\}$, $\text{output}(N_j) = v$. Если все узлы начинают с одним и тем же начальным значением v , то лю-

бое значение, принятое в результате выполнения протокола консенсуса, должно быть равно v ;

2. единогласие: $\forall i, j \in \{1, \dots, n\}$, если $\text{output}(N_i) = v$, то $\text{output}(N_j) = v$. Если один узел завершает протокол с некоторым значением v , то все другие узлы, которые также завершили протокол, должны иметь идентичное значение v ;
3. завершение: $\forall i \in \{1, \dots, n\}$, узел N_i завершает выполнение протокола в конечное время.

1.1.3 Стохастическая аппроксимация

Теория стохастической аппроксимации описывает численные схемы для поиска корня в общем случае неизвестной выпуклой монотонной функции с несмещенным откликом. Наиболее известным результатом теории стохастической аппроксимации является алгоритм Роббинса-Монро, задающий численную схему, обеспечивающую сходимость к корню заданного уравнения. В работе [19] показано, что в случае если функции отклика s строго выпукла и дважды дифференцируема, то асимптотическая скорость сходимости линейна $\mathcal{O}(\frac{1}{n})$

Определение Стохастическая аппроксимация — метод решения задач статистического оценивания, строящихся в виде последовательного приближения на основании наблюдений, представленных случайной величиной.

Теорема. Алгоритм Роббинса-Монро. Алгоритм, выполняющий пересчет по правилу $d_{n+1} = d_n + a_d(s^* - s)$ сходится в L^2 норме при выполнении:

1. значения функции отклика монотонны и s ограничены: $\exists N \forall x : |s(x)| \leq N$;
2. $\sum_{t=0}^{\infty} a_t = \infty$;
3. $\exists M : |\sum_{t=0}^{\infty} a_t^2| < M$.

Доказательство: Следуя доказательству [20], используем рекуррентную схему связи между ошибками на каждом шаге алгоритма $\gamma_t = d_t - d^*$. В этом случае правило обновления Роббинса-Монро запишется как:

$$\gamma_n^2 = \mathbb{E}_{x_t \sim p(x|s(d))} (d_n + a_t(s^* - x_t) - d^*)^2. \quad (1.23)$$

Раскрываем квадрат разности:

$$\gamma_n^2 = (d_n - d^*)^2 + a_t^2 \mathbb{E}_{x_t \sim p(x|s(d))} (s^* - x_t)^2 - 2a_t \mathbb{E}_{x_t \sim p(x|s(d))} [(s^* - x_t)(d_n - d^*)]. \quad (1.24)$$

Используем несмещенность оценки $\mathbb{E}_{x \sim p(x|s)} x = s$:

$$d_n = 2a_t(s^* - s(d))(d_n - d^*). \quad (1.25)$$

Получаем связь между ошибками на каждом шаге $gamma_n$:

$$\lim_{n \rightarrow \infty} \gamma_n = \gamma_1 + \sum_{j=1}^{\infty} a_n^2 c_n - 2 \sum_{j=1}^{\infty} a_n d_n < \infty. \quad (1.26)$$

Сходимость обоснуем через два последовательных шага:

- $\sum_{j=1}^{\infty} a_n^2 c_n < \infty$, поскольку при $\sum_{j=1}^{\infty} a_n^2 < \infty$ и c_n -ограничены;
- $\sum_{j=1}^{\infty} a_n d_n < \sum_{j=1}^{\infty} a_n^2 c_n$, т.к. $b_n \geq 0$.

Покажем, что $\lim_{n \rightarrow \infty} \gamma_n = 0$ следует из $\sum_{j=1}^{\infty} a_j = \infty$. Действительно, зададим ряд $m_n = \frac{\gamma_n}{d_n}$, т.к. он неотрицателен, то $\sum_{i=1}^{\infty} a_n k_n = \infty$. Тогда $\sum_{i=1}^{\infty} a_n k_n \gamma_N < \infty$. Из сходимости ряда из неотрицательных элементов следует, что $\lim_{n \rightarrow \infty} \gamma_n = 0$. ■

Выбор коэффициента a_n в численной схеме существенно влияет на число шагов, необходимых для того, чтобы последовательность сошлась. Авторы оригинального алгоритма предлагают в качестве $a_n = \frac{\lambda}{n}$, где коэффициент λ определяется экспериментально для каждой постановки. Это обстоятельство затрудняет адаптацию метода в практических случаях, поэтому существуют модификации схемы, не требующие подбора гиперпараметра. Например, в работе [21] был разработан метод выбора коэффициентов исходя из требований несмещенности оценки $Ed_n = d$ и минимизации дисперсии $Dd_n = 0$ на каждом шаге. Такой подход позволил получить оптимальные численные коэффициенты для случая цепи гауссовых распределений [22].

Для ускорения сходимости современные исследователи дополняют оригинальную схему заменой члена s^* на параметр шага b_n : $x_{n+1} = x_n - a_n(y_n - b_n)$ [23]. Доказательство сходимости метода описано в работе [24].

Теорема. Алгоритм Роббинса-Монро для случая $b_n \neq s^*$ [24] Модифицированная схема $x_{n+1} = x_n - a_n(y_n - b_n)$ сходится в условиях сходимости алгоритма Роббинса-Монро при условии:

$$\sum_{n=2}^{\infty} a_n |b_n - s^*| \sum_{j=1}^{n-1} a_j < \infty \quad (1.27)$$

На практике также представляет интерес работы с многомерными данными. В работе [25] описан подход с вектором бинарных величин.

Теорема. Алгоритм Роббинса-Монро для многомерного бинарного случая [25]: Пусть \mathbf{x} — вектор бернулевских случайных величин с параметрами $\mathbf{s} = f(\mathbf{d})$, где $f(\mathbf{x})$ — выпуклая. Тогда схема пересчета $d_t = \mathbf{x}_t + A^{(t)}(s - \mathbf{x}_t)$ с шагами $A^{(t)}$, удовлетворяющими условиям: $\forall t, j \rightarrow a_{jj}^{(t)} > 0$, $\sum_{t=1}^{\infty} a_{jj}^{(t)} = \infty$, $\sum_{n=1}^{\infty} (a_{jj}^{(t)})^2 < \infty$ сходится по вероятности к целевому значению \mathbf{s}^*

Доказательство:

$$\begin{aligned}
& \mathbb{E} \left\{ (\mathbf{x}_{n+1} - \boldsymbol{\theta})^\top (\mathbf{x}_{n+1} - \boldsymbol{\theta}) \right\} \\
&= \mathbb{E} \left(\mathbb{E} \left[\{ \mathbf{x}_n - \boldsymbol{\theta} - A_n (\mathbf{y}_n - \boldsymbol{\alpha}) \}^\top \right. \right. \\
&\quad \left. \left. \times \{ \mathbf{x}_n - \boldsymbol{\theta} - A_n (\mathbf{y}_n - \boldsymbol{\alpha}) \} \mid \mathbf{x}_n \right] \right) \\
&= \mathbb{E} (\mathbf{x}_n - \boldsymbol{\theta})^\top (\mathbf{x}_n - \boldsymbol{\theta}) - 2 \mathbb{E} (\mathbf{x}_n - \boldsymbol{\theta})^\top A_n (\mathbf{m}_n^x - \boldsymbol{\alpha}) \\
&\quad + \mathbb{E} \left\{ (\mathbf{y}_n - \boldsymbol{\alpha})^\top A_n^\top A_n (\mathbf{y}_n - \boldsymbol{\alpha}) \right\} \\
&= \mathbb{E} (\mathbf{x}_1 - \boldsymbol{\theta})^\top (\mathbf{x}_1 - \boldsymbol{\theta}) - 2 \sum_{i=1}^n \mathbb{E} (\mathbf{x}_i - \boldsymbol{\theta})^\top A_i (\mathbf{m}_i^x - \boldsymbol{\alpha}) \\
&\quad + \sum_{i=1}^n \mathbb{E} \left\{ (\mathbf{y}_i - \boldsymbol{\alpha})^\top A_i^\top A_i (\mathbf{y}_i - \boldsymbol{\alpha}) \right\} \\
&\geq 0,
\end{aligned} \tag{1.28}$$

где ожидания получены по \mathbf{x}_n , $\mathbf{m}_n^x = \mathbb{E}(\mathbf{y}_n \mid \mathbf{x}_n) = (M_1(x_{1n}), \dots, M_p(x_{pn}))^\top$.

Обозначим $\mathbf{e}_i = (e_{i1}, \dots, e_{ip})^\top = \mathbf{y}_i - \boldsymbol{\alpha}$. Из ограниченности e_{ji} сумма $\sum_{i=1}^n \mathbb{E} \left\{ (\mathbf{y}_i - \boldsymbol{\alpha})^\top A_i^\top A_i (\mathbf{y}_i - \boldsymbol{\alpha}) \right\}$ запишется как

$$\sum_{j=1}^p \sum_{i=1}^n \left\{ \mathbb{E} (e_{ji}^2) \sum_{s=1}^p a_{sj,i}^2 \right\} + \sum_{j \neq k} \sum_{i=1}^n \left\{ \mathbb{E} (e_{ji} e_{ki}) \sum_{s=1}^p a_{sj,i} a_{sk,i} \right\}.$$

Докажем в два шага:

1. Ряды $\sum_{i=1}^n \left\{ \mathbb{E} (e_{ji}^2) \sum_{s=1}^p a_{sj,i}^2 \right\}$ и $\sum_{i=1}^n \left\{ \mathbb{E} (e_{ji} e_{ki}) \sum_{s=1}^p a_{sj,i} a_{sk,i} \right\}$ сходятся. Следовательно, сходится и ряд $\sum_{i=1}^n \mathbb{E} \left\{ (\mathbf{y}_i - \boldsymbol{\alpha})^\top A_i^\top A_i (\mathbf{y}_i - \boldsymbol{\alpha}) \right\}$, причем к положительному значению $\sum_{n=1}^\infty \mathbb{E} \left\{ \mathbf{x}_n - \boldsymbol{\theta}^\top A_n (\mathbf{m}_n^x - \boldsymbol{\alpha}) \right\} < \infty$.
2. Также для $j = 1, \dots, p$, поскольку M_j функция распределения и $\dot{M}_j(\theta_j) > 0$, существует положительные постоянные ℓ_j и u_j такие что:

$$0 < \ell_j \leq \frac{M_j(x_j) - \alpha_j}{x_j - \theta_j} \leq u_j < \infty$$

для всех x_j . Пусть $\ell = \min \{ \ell_j : j = 1, \dots, p \}$ и $u = \max \{ u_j : j = 1, \dots, p \}$.

Обозначим $\mathbf{a}_{j,n}$ j столбцом матрицы A_n . Тогда:

$$\begin{aligned}
& (\mathbf{x}_n - \boldsymbol{\theta})^\top A_n (\mathbf{m}_n^x - \boldsymbol{\alpha}) \\
&= \sum_{j=1}^p (\mathbf{x}_n - \boldsymbol{\theta})^\top \mathbf{a}_{j,n} (x_{jn} - \theta_j) \frac{M_j(x_{jn}) - \alpha_j}{x_{jn} - \theta_j} \\
&\geq \sum_{j=1}^p (\mathbf{x}_n - \boldsymbol{\theta})^\top \mathbf{a}_{j,n} (x_{jn} - \theta_j) \delta_{jn} \\
&= (\mathbf{x}_n - \boldsymbol{\theta})^\top A_n \Delta_n (\mathbf{x}_n - \boldsymbol{\theta}),
\end{aligned} \tag{1.29}$$

где $\Delta_n = \text{diag}(\delta_{1n}, \dots, \delta_{pn})$ с

$$\delta_{jn} = \begin{cases} \ell, & \text{если } (\mathbf{x}_n - \boldsymbol{\theta})^\top \mathbf{a}_{j,n} (x_{jn} - \theta_j) \geq 0 \\ u, & \text{иначе} \end{cases} \quad (1.30)$$

Следовательно, имея $\sum_{n=1}^{\infty} \mathbb{E} (\mathbf{x}_n - \boldsymbol{\theta})^\top A_n \Delta_n (\mathbf{x}_n - \boldsymbol{\theta}) < \infty$. Поскольку это выражение можно записать как:

$$\begin{aligned} & \sum_{j=1}^p \sum_{n=1}^{\infty} a_{jj,n} \delta_{jn} \mathbb{E} (x_{jn} - \theta_j)^2 \\ & + \sum_{j \neq k} \sum_{n=1}^{\infty} a_{jk,n} \delta_{kn} \mathbb{E} (x_{jn} - \theta_j) (x_{kn} - \theta_k) \end{aligned} \quad (\text{A2})$$

Заметим, что для второго члена выполняется:

$$\begin{aligned} & \sum_{j \neq k} \sum_{n=1}^{\infty} \frac{1}{2} |a_{jk,n} \delta_{kn}| \{ \mathbb{E} (x_{jn} - \theta_j)^2 + \mathbb{E} (x_{kn} - \theta_k)^2 \} \\ & = \frac{1}{2} \sum_{j=1}^p \sum_{n=1}^{\infty} \mathbb{E} (x_{jn} - \theta_j)^2 \sum_{k \neq j} \{ |a_{jk,n} \delta_{kn}| + |a_{kj,n} \delta_{jn}| \}. \end{aligned} \quad (1.31)$$

Предположение $|a_{jk,n}/a_{jj,n}| \rightarrow 0$ где $n \rightarrow \infty$ для $j \neq k$ и $a_{jj,n} > 0$ означает, что $\exists N : \forall n > N, |a_{jk,n}| < a_{jj,n}/\{2c(p-1)\}, k \neq j$, где c константа большая $u/(2\ell)$ и u и ℓ . Тогда для заданного j выполняется:

$$\begin{aligned} \sum_{k \neq j} |a_{jk,n} \delta_{kn}| + |a_{kj,n} \delta_{jn}| & < \sum_{k \neq j} \left\{ \frac{a_{jj,n} \delta_{kn}}{2c(p-1)} + \frac{a_{jj,n} \delta_{jn}}{2c(p-1)} \right\} \\ & < \frac{a_{jj,n} u}{c} \end{aligned} \quad (1.32)$$

Тогда сумма запишется как:

$$\begin{aligned} & \sum_{j=1}^p \sum_{n > N}^{\infty} a_{jj,n} \delta_{jn} \mathbb{E} (x_{jn} - \theta_j)^2 \\ & + \sum_{j \neq k} \sum_{n > N}^{\infty} a_{jk,n} \delta_{kn} \mathbb{E} (x_{jn} - \theta_j) (x_{kn} - \theta_k) \\ & \geq \sum_{j=1}^p \sum_{n > N}^{\infty} a_{jj,n} \delta_{jn} \mathbb{E} (x_{jn} - \theta_j)^2 - \frac{1}{2} \sum_{j=1}^p \sum_{n > N}^{\infty} \mathbb{E} (x_{jn} - \theta_j)^2 \\ & \times \sum_{k \neq j} \{ |a_{jk,n} \delta_{kn}| + |a_{kj,n} \delta_{jn}| \} \\ & > \sum_{j=1}^p \sum_{n > N}^{\infty} \left(\delta_{jn} - \frac{u}{2c} \right) a_{jj,n} \mathbb{E} (x_{jn} - \theta_j)^2 \end{aligned} \quad (1.33)$$

Заметим, что оценка положительна исходя из выбора c . Следовательно, исходя из

$E(x_{jn} - \theta_j)^2$ сходится к 0 $\forall j = 1, \dots, p$, что означает сходимость по вероятности последовательности \mathbf{x}_n к $\boldsymbol{\theta}$ ■

Стохастическую аппроксимацию можно представить как непрерывную постановку $\dot{z} = g(z(t))$. Для этого запишем численную схему в более общем виде:

$$x_{n+1} = x_n + a_n [g(x_n) + \varepsilon_n], \quad (1.34)$$

где $g(x_n)$ — преобразование, сужающее ряд к решению x^* . Зададим промежутки наблюдения t_n через шаги a_n :

$$t_n = \sum_{k=0}^n a_k \quad (1.35)$$

Заметим, что в условиях сходимости алгоритма Роббинса-Монро, последовательность времен неограничена, т.к. $\sum_{k=0}^n a_k = \infty$. Но шаг с течением времени сужается как $\frac{1}{n^2}$ в асимптотике $\sum_{k=0}^n a_k^2 < \infty$. Покажем, что в условиях липшицевости $g(x_t)$ существует аппроксимирующее схему обыкновенное дифференциальное уравнение.

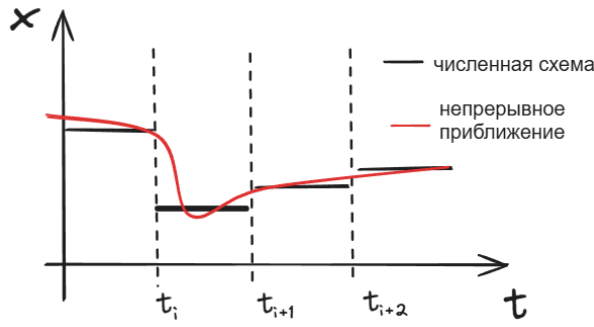


Рис. 1.5: Численная схема стохастической аппроксимации может быть приближена дифференциальным уравнением $\frac{dz}{dt} = g(z(t))$

Теорема: Об переходе к непрерывному случаю Обыкновенное дифференциальное уравнения $\frac{dz_m}{dt} = g(z_m(t))$ с начальными условиями $z(t_m) = x_m$ аксиоматически аппроксимирует численную схему вблизи бесконечности $x_{n+1} = x_n + a_n [g(x_n) + \varepsilon_n]$:

$$\lim_{m \rightarrow \infty} \sup_{t \in [t_m, t_{m+1}]} |x(t) - z_m(t)| = 0,$$

при условиях липшицевости сужающей функции $\exists L \forall x, y : |g(x) - g(y)| < L|x - y|$.

Доказательство: Для доказательства утверждения рассмотрим связь между интегральной формой дифференциального уравнения и численной аппроксимацией алгоритма. Для этого запишем заданное дифференциальное уравнение в интегральной форме:

$$z(t_m) = z(t_n) + \int_{t_n}^{t_m} g(z(u)) du \quad (1.36)$$

Стохастический ряд приближений согласно алгоритму Роббинса-Монро запишется

как:

$$x(t_n) = x_m + \sum_{k=m}^{n-1} k = ma_k g(x_k) + \sum_{k=m}^{n-1} a_k \varepsilon_k \quad (1.37)$$

С учетом 1.36 и $z(t_m) = x_m$:

$$x(t_n) = x_m + \int_{t_m}^{t_n} g(x([u]_a)) + \sum_{k=m}^{n-1} a_k \varepsilon_k, \quad (1.38)$$

где $[u]_a$ ступенчатая функция $\max_k(k : t_k \leq u)$.

Рассмотрим искомое выражение как разность полученных представлений:

$$\|z(t_n) - x(t_n)\| \leq \left\| \sum_{k=m}^{n-1} a_k \varepsilon_k \right\| + \int_{t_m}^{t_n} \|g(z(u)) - g([u]_a)\| \quad (1.39)$$

Воспользуемся липшицевостью функции $g(z)$:

$$\|z(t_n) - x(t_n)\| \leq \left\| \sum_{k=m}^{n-1} a_k \varepsilon_k \right\| + L \int_{t_m}^{t_n} |z(u) - x([u]_a)| du \quad (1.40)$$

Используем лемму Гронуолла для перехода к произведению вида:

$$\|z(t_n) - x(t_n)\| \leq \left\| \sum_{k=m}^{n-1} a_k \varepsilon_k \right\| \exp\left(L \int_{t_m}^{t_n} du\right) \leq C \left\| \sum_{k=m}^{n-1} a_k \varepsilon_k \right\| \exp(L(t_n - t_m)), \quad (1.41)$$

где C — положительные константы. Оценим сверху ряд $\sum_{k=m}^{n-1} a_k \varepsilon_k$ используя аддитивность матожидания:

$$\left\| \sum_{k=m}^{n-1} a_k \varepsilon_k \right\| \leq \sum_{k=1}^n a_k \mathbb{E} \varepsilon_k^2 \leq \sigma^2 \sum_{k=1}^{\infty} a_k^2 \leq \infty \quad (1.42)$$

Тогда в силу ограниченности заданного ряда получим:

$$\lim_{t_n - t_m \rightarrow 0} \|z(t_n) - x(t_n)\| = 0 \quad (1.43)$$

Или, что эквивалентно:

$$\lim_{m \rightarrow \infty} \sup_{t \in [t_m, t_{m+1}]} |x(t) - z_m(t)| = 0, \quad (1.44)$$

■

Существенный вклад в развитие методов стохастической аппроксимации внёс Б. Т. Поляк [10], предложивший метод усреднения управляющего параметра θ :

$$\hat{\theta}_{n+1} = \sum_{i=0}^n \theta_i. \quad (1.45)$$

Такой подход позволяет подавлять высокочастотные шумовые компоненты при аппроксимации ряда, что позволяет использовать методы с большим шагом спуска. Условия эффективной применимости состоят в малом изменении коэффициентов a_n :

$$\frac{a_n - a_{n+1}}{a_n} = o(a_n). \quad (1.46)$$

1.2 Оптимальный транспорт

Задача оптимального транспорта (Optimal Transport)[26] является одной из ключевых в области теории вероятностей и машинного обучения. Она представляет собой проблему определения оптимального способа перемещения вероятностной массы из одной распределенной системы в другую с минимальными затратами или стоимостью. Формально задача состоит в составлении транспортного плана $T : \mathcal{X} \rightarrow \mathcal{Y}$, который переводит распределение $\mu \subset \mathcal{X}$ в распределение $\nu \subset \mathcal{Y}$, минимизируя некоторую функцию стоимости. Функция стоимости $c : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ обычно является мерой сходства между элементами из \mathcal{X} и \mathcal{Y} , такой как квадрат расстояния.

Определение (Монже): **Оптимальный транспорт** по Монже вводится путем рассмотрения вероятностных распределений μ и ν на метрических пространствах \mathcal{X} и \mathcal{Y} :

$$\inf_{\gamma \in \Pi(\mu, \nu)} \int_{\mathcal{X} \times \mathcal{Y}} c(x, y) d\gamma(x, y), \quad (1.47)$$

где $\Pi(\mu, \nu)$ обозначает множество всех возможных совместных распределений γ на $\mathcal{X} \times \mathcal{Y}$ с фиксированными маргинальными распределениями μ и ν , а $c(x, y)$ — функция стоимости перевозки массы из x в y .

Определение (Канторович): **Оптимальный транспорт** по Канторовичу вводится через потенциал ϕ , который минимизирует функционал стоимости:

$$\inf_{\phi} \left(\int_{\mathcal{X}} \phi(x) d\mu(x) + \int_{\mathcal{Y}} \psi(y) d\nu(y) \right), \quad (1.48)$$

где ψ — обратная функция к ϕ . Таким образом, отображение $T : \mathcal{X} \rightarrow \mathcal{Y}$ вытекает из градиента потенциала.

Заметим, что постановка Канторовича обобщает постановку Монже 1.6. В отличие от постановки Монже оптимальный транспорт по Канторовичу допускает распределение вероятностной массы в непрерывном случае.

Итоговая стоимость оптимального транспортного плана называется метрикой Васерштейна.

Определение: Пусть (X, d) — метрическое пространство и $P(X)$ — множество всех вероятностных мер на X . Для двух вероятностных мер μ и ν на X **метрика Вас-**

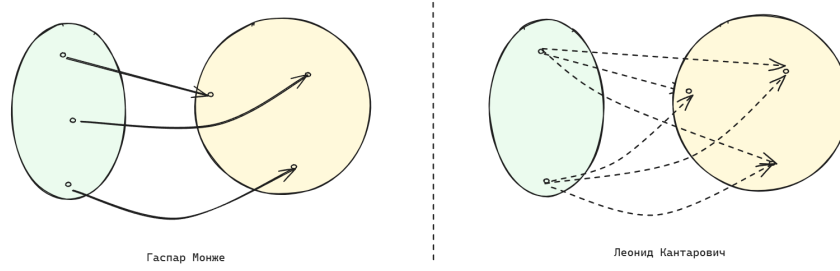


Рис. 1.6: Различие в подходе по Монже и Кантаровичу. В постановке Канторовича задача релаксирует до непрерывного распределения

серштейна порядка p , где $p \geq 1$, определяется как:

$$W_p(\mu, \nu) = \left(\inf_{\gamma \in \Gamma(\mu, \nu)} \int_{X \times X} d(x, y)^p d\gamma(x, y) \right)^{1/p}, \quad (1.49)$$

где $\Gamma(\mu, \nu)$ — множество всех сопряжённых мер γ на $X \times X$ с маргиналами μ и ν .

Метрика имеет практическое применение для задач физики, биологии и машинного обучения, поскольку задает дифференцируемую разность между распределениями.

Определение: **Метрическая производная** кривой $\rho_t, t \in [0, T]$ в вероятностном пространстве $\mathcal{P}_2(\mathbb{R}^N)$ запишется как:

$$|\rho'_t| = \lim_{dt \rightarrow 0} \frac{\mathcal{W}_2(\rho_t, \rho_{t+dt})}{dt} \quad (1.50)$$

Метод оптимального транспорта также активно применяется для анализа стохастических процессов. Базовой моделью, описывающей стохастическое движение с смещением, является процесс Ланжевена.

Определение: Процесс Ланжевена называется случайный процесс вида

$$dX_t = -\nabla\Phi(x)dt + \sqrt{2\beta^{-1}}dW_t, \quad (1.51)$$

где $\Phi(X)$ — потенциал задающий снос частицы, β - масштаб блуждания и dW_t - процесс Винера.

Стохастическое усреднение процесса Ланжевена можно описать с помощью уравнения Колмогорова-Фоккера-Планка, задающего эволюцию вероятностной массы в дифференциальной форме $\rho_t(x)$:

$$\frac{\partial \rho_t}{\partial t} = \text{div}(\nabla\Phi(x)\rho_t) + \beta^{-1}\Delta\rho_t. \quad (1.52)$$

Для естественной работы в энергетических постановках водится функционал, задающий коэффициента сноса с потенциалом Φ . Таким образом, исходное уравнение можно переписать в вариационной постановке 1.8.

Определение: **Функционал Фоккера-Планка** для распределения ρ записыва-

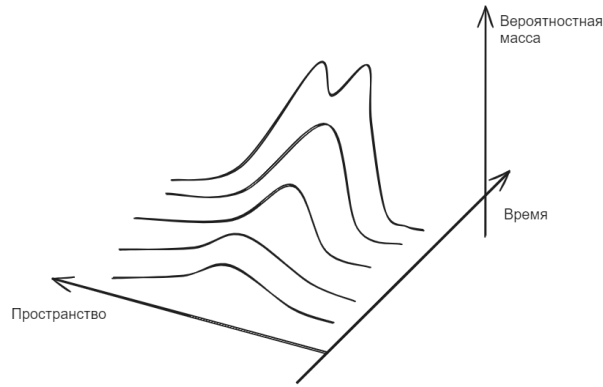


Рис. 1.7: Эволюция вероятностной массы в уравнение Ланжевена

ется как:

$$\mathcal{F}_{FP}(\rho) = \int \Phi(x) d\rho(x) + \beta^{-1} \int \log \rho(x) d\rho(x). \quad (1.53)$$

Научная группа Йордана-Кинана-Отто в работе [27] показала, что маргинальные

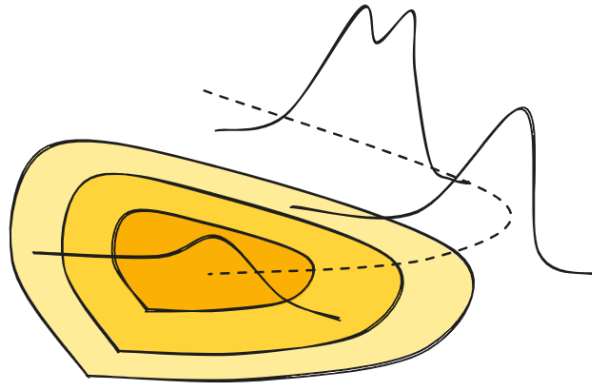


Рис. 1.8: Визуализация постановки уравнения Фоккера-Планка в вариационной форме

вероятностные меры процесса Ланжевена подчиняются уравнению градиентного потока Вассерштейна относительно функционала Фоккера-Планка.

Определение: **Схема Йордана-Кинана-Отто** задает правило обновления уравнения вероятности в виде минимизации функционала энергии и расстояния:

$$\rho^{n+1} = \operatorname{argmin}_{\rho} \left(\frac{1}{2\tau} W_2^2(\rho, \rho^n) + \mathcal{F}(\rho) \right), \quad (1.54)$$

где:

- $\tau > 0$ — шаг по времени;
- $W_2(\rho, \rho^n)$ — метрика Вассерштейна 2-го порядка между плотностями ρ и ρ^n ;
- $\mathcal{F}(\rho)$ — функционал свободной энергии, который может включать в себя энтропийный член и потенциальную энергию системы.

Функционал свободной энергии $\mathcal{F}(\rho)$ задается в виде:

$$\mathcal{F}(\rho) = \int_V f(\rho(x)) dx + \int_V V(x)\rho(x) dx, \quad (1.55)$$

где $f(\rho)$ — внутренний энергетический член, зависящий от плотности, а $V(x)$ — внешний потенциал.

Глава 2

Методы машинного обучения

2.1 Нейросетевые подходы

Изначально искусственные нейронные сети строились как модель коры головного мозга человека. В отличие от своего биологического аналога нейронная сеть, как правило, имеет дифференцируемые функции активации, необходимые для эффективного обучения в ходе обратного распространения ошибки.

Определение: **Функцией активации** в нейронной сети называется *нелинейная* функция, связывающая выходной сигнал и активацию нейрона.

На практике широко используются функции активации в виде сигмоидов $\sigma(x)$, ReLU [28] и GeLU [29]:

$$\begin{aligned}\sigma(x) &= \frac{1}{1 + \exp(-x)} \\ \text{ReLU}(x) &= \min(0, x) \\ \text{Tanh}(x) &= \frac{e^x - e^{-x}}{e^x + e^{-x}} \\ \text{Gelu}(x) &= \sigma(x)x\end{aligned}\tag{2.1}$$

Функция активации вводится для добавления в модель нелинейности, что позволяет нейронной сети моделировать сложные нелинейные зависимости в данных. Некоторые из распространенных функций активации включают в себя сигмоидальную функцию (σ), гиперболический тангенс (\tanh), ReLU (Rectified Linear Unit) и их вариации.

Определение: Перцептроном называется параметрическая математическая модель нейрона. Перцептрон задается матрицей весов W , смещением b и функцией активации σ :

$$\mathbf{y} = \sigma(W\mathbf{x} + \vec{b})\tag{2.2}$$

Определение: **Нейронные сети** — параметрическая аппроксимирующая данные модель, состоящая из упорядоченных слоев обучаемых слоев перцептронов:

$$f^i(W^i f^{i-1}(W^{i-1} \dots f^1(W^1(x))))\tag{2.3}$$

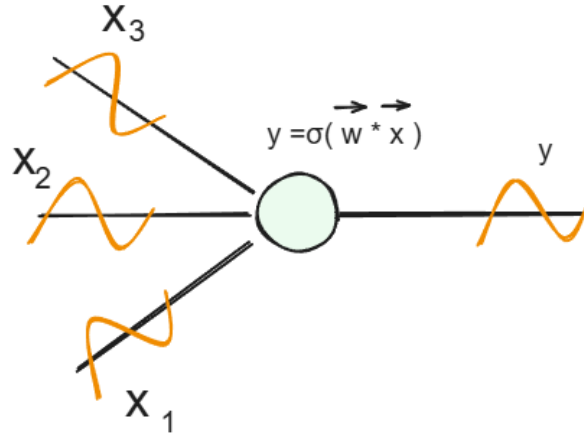


Рис. 2.1: Перцептрон — базовый элемент нейронной сети

где f^i — функция активации i -ого слоя, W^i — веса активации i -ого слоя.

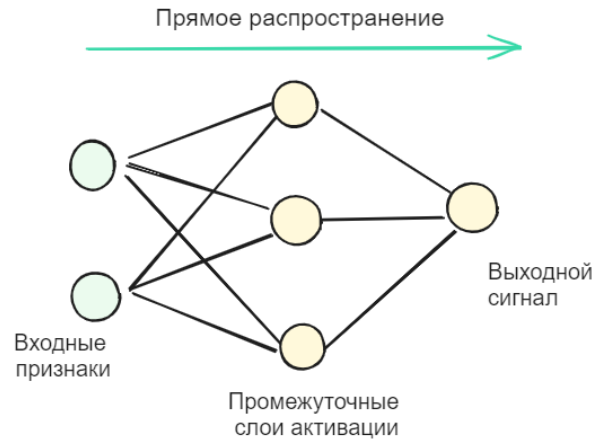


Рис. 2.2: Преобразование сигнала выполняется в промежуточных слоях активации в ходе прямого распространения сигнала

В случае многослойной нейронной сети, выходы нейронов одного слоя становятся входами для следующего слоя, образуя цепочку преобразований. Процесс распространения через нейроны последовательных слоев называется прямым распространением (от *англ.* forward propagation).

Во время обучения модель минимизирует функцию потерь \mathcal{L} , которая оценивает отличие предсказанного результата y_i от истинного значения y_i :

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^N L(y_i, \hat{y}_i). \quad (2.4)$$

Значительное число параметров в современных нейронных сетях требует выбора численно эффективных методов оптимизации. Базовым и наиболее распространенным алгоритмом на практике является градиентный спуск, выполняемый с помощью методов автоматического дифференцирования [30][31]. Общая техника называется обратным распространением ошибки [32] и основана на правиле дифференцирова-

ния сложной функции:

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial w_{jk}^{(L-1)}} &= \frac{1}{N} \sum_{i=1}^N \frac{\partial a_j^{(L)}}{\partial w_{jk}^{(L-1)}} \frac{\partial L}{\partial a_j^{(L)}} \\ \frac{\partial \mathcal{L}}{\partial \alpha_k^{L-1}} &= \sum_{j=1}^{N_L} \frac{\partial a_j^{(L)}}{\partial \alpha_k^{(L-1)}} \frac{\partial L}{\partial a_j^{(L)}}\end{aligned}\quad (2.5)$$

где α_k^{L-1} — активации k -перцептрона на слое L , $w_{jk}^{(L-1)}$ — вес нейрона jk на слое L .

Кодировщики и декодировщики

Архитектура кодировщик-декодировщик широко распространена в машинном обучении, кодирования информации, компрессии и криптографии. В моделировании нейросетей также используют этот подход для задач:

- генерации перевода и пересказа;
- классификации текста;
- распознавания частей речи и выделения имен собственных.

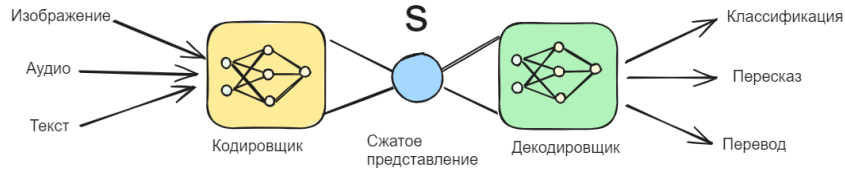


Рис. 2.3: Преобразование сигнала выполняется в промежуточных слоях активации в ходе прямого распространения сигнала

Определение: Рекуррентными нейронными сетями называют модели, использующие скрытые состояния предыдущих слоев для расчета следующих.

$$\begin{aligned}h_t &= \sigma(W_x x_t + W_h h_{t-1} + b_h) \\ y_t &= \sigma(W_y h_t + b_y),\end{aligned}\quad (2.6)$$

где

- x_t, h_t, y_t — векторы входного, скрытого и выходного слоя;
- W_x, W_h, W_y — матрицы обновления состояния.

Ошибка модели для случая обработки последовательностей равной длины запишется как:

$$\mathcal{L}(\hat{y}, y) = \sum_{t=1}^{T_y} \mathcal{L}(\hat{y}_t, y_t) \quad (2.7)$$

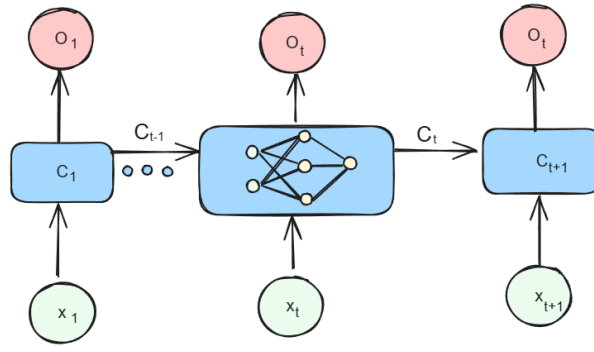


Рис. 2.4: Каждая ячейка рекуррентной нейронной сети выполняет обновление представления.

На практике матрицы обновления постоянны для каждой ячейки, поэтому правило обновления матрицы весов W запишется как:

$$\frac{\partial \mathcal{L}^{(T)}}{\partial W} = \sum_{t=1}^T \frac{\partial \mathcal{L}^{(t)}}{\partial W} \quad (2.8)$$

Такой подход называется распространением ошибки во времени. Также существуют модификации механизма нейронной сети, заключающиеся в добавлении параллельного блока памяти:

- долгая короткая память [33];
- управляемый рекуррентный блок [34].

Механизм внимания, основанный на модели рабочей памяти [35], позволил значительно улучшить результаты решения задач перевода, пересказа и "понимания" языка. Численный блок рассчитывает вектор внимания α , который определяет важность каждого элемента входной последовательности на текущем временном шаге [36]:

$$\alpha_t = \text{softmax}(f(h_t, X)), \quad (2.9)$$

где f - функция, определяющая важность каждого элемента входной последовательности, а softmax применяется для расчета нормированных весов внимания.

Механизм внимания позволяет модели "сосредоточиться" на наиболее значимых компонентах входных данных в каждый момент времени, что делает его наиболее эффективным для задач, требующих адаптивности и контекстного понимания, таких как машинный перевод, генерация текста и вопросно-ответные системы. Этот механизм стал ключевым инструментом в области генеративного моделирования естественного языка, позволяя моделям эффективно работать с различными типами данных и контекстами.

Transformer. Представление внимания

Модель Transformer является архитектурой глубокого обучения, предназначенной для обработки последовательных данных, таких как тексты или временные ряды. Она была предложена в работе [37] и стала одной из наиболее инновационных архитектур в области обработки естественного языка.

Основной компонент модели Transformer это механизм внимания. Он позволяет сфокусировать модель на наиболее важных компонентах входных данных при выполнении таких задач, как машинный перевод или обработка текста.

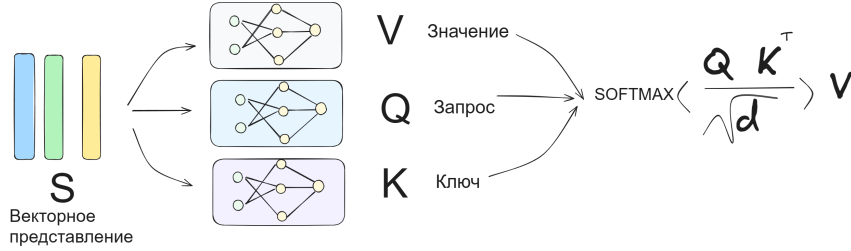


Рис. 2.5: Механизм внимания в архитектуре Transformer [37].

Расчет механизма внимания в архитектуре Transformer состоит из трех основных этапов:

1. Расчет векторов запроса q , ключа k и значения v . Они используются для вычисления весов входных данных и определения их важности для каждого элемента:

$$\begin{aligned} q &= W_q x \\ k &= W_k x, \\ v &= W_v x, \end{aligned} \tag{2.10}$$

где W_q , W_k , W_v — матрицы весов, обучаемые моделью.

2. Расчет логитов e_{ij} :

$$e_{ij} = \frac{q \cdot k_j}{\sqrt{d_k}}, \tag{2.11}$$

где d_k — длина запроса,

3. Преобразование логитов в веса внимания α_{ij} с помощью функции softmax:

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{j'} \exp(e_{ij'})}. \tag{2.12}$$

2.2 Генеративные подходы

Порождающие модели — современное и быстро развивающееся направление работы с данными, позволяющее создавать искусственные данные и наращивать вероятностную массу. Ряд ключевых достижений направления:

1. порождающие грамматики [38];
2. графические вероятностные модели [39];
3. состязательные порождающие модели [40];
4. диффузионные порождающие модели [41].

Порождающие модели задают совместное распределение наблюдаемого объекта x и его свойств $y — p(x, y)$. В этом состоит основное различие между порождающими и дискриминирующими моделями $p(y|x)$ 2.6.

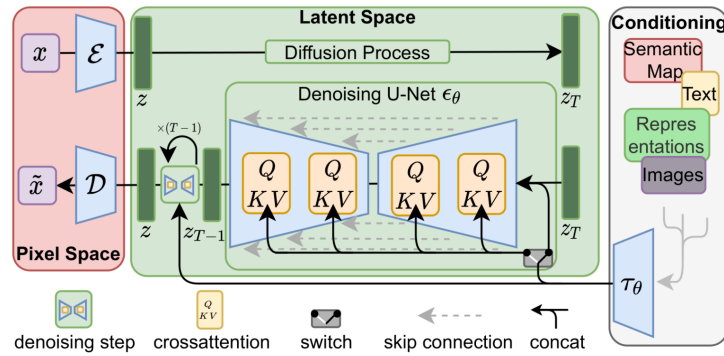


Рис. 2.6: Архитектура современной модели Stable Diffusion.

Порождающие модели используют параметрические модели p_θ для аппроксимации истинных функций распределений на наборе обучающих данных. Выбор параметрической функции аппроксимации, как правило, зависит от числа образцов в корпусе данных. Для больших корпусов, как правило, используют нейросети. Простейшим видом порождающей модели является авторегрессионная модель, использующая предшествующий контекст для предсказания следующего элемента.

Определение: Авторегрессионные модели представляют собой класс порождающих моделей, с вычислимой вероятностью, выполняющие генерацию с помощью цепи последовательных преобразований:

$$p(x^{(1)}, \dots, x^{(t)}) = \prod_{t=1}^T p_\theta(x^{(t)} | x^{(1)}, \dots, x^{(t-1)}) \quad (2.13)$$

Как правило, авторегрессионные модели используются для генерации последовательностей, временных рядов и текста. В то же время с данными, не допускающими однозначного упорядочивания или данными с неравномерным шагом, такой класс моделей работает недостаточно эффективно.

Для оценки различий между вероятностными распределениями используются **дивергенции** с набором правил:

1. $\forall p, q \in M \rightarrow D(p, q) \geq 0$
2. $p = q \leftrightarrow D(p, q) = 0$

3. $\forall p \rightarrow D(p, p + dp)$ — положительно определенная квадратичная форма.

В отличие от метрики дивергенции могут быть несимметричны. На практике используются специальный класс f -дивергенций, заданных матожиданием.

Определение: f -дивергенцией называется выпуклая функция, удовлетворяющая равенству $f(1) = 0$.

$$D_f(\pi \parallel \rho) = \mathbb{E}_{\rho(x)} f\left(\frac{\pi(x)}{\rho(x)}\right).$$

Семейство f -дивергенций включает функции:

1. дивергенция Кульбака-Лейбнера: $u \ln u$;
2. обратная дивергенция Кульбака-Лейбнера: $-\ln u$;
3. дивергенция Йенсена-Шэннона: $\frac{1}{2} (u \ln u - (u + 1) \ln(\frac{u+1}{2}))$.

Нижней вариационной оценкой называется техника максимизации подпирающей границы параметрического распределения $p(\mathbf{x}, \mathbf{z})$ вторым $q(\mathbf{x}, \mathbf{z})$, где переменная \mathbf{z} называется скрытой. В аналитической форме нижняя граница записывается как:

$$\mathcal{L}(\phi, \theta; x) = \mathbb{E}_{z \sim q_\phi(z|x)} \left[\ln \frac{p_\theta(x, z)}{q_\phi(z|x)} \right]. \quad (2.14)$$

Перепишем в виде KL-дивергенции:

$$\begin{aligned} & D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z}|\mathbf{x})) \\ &= \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}|\mathbf{x})} d\mathbf{z} \\ &= \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x}) p_\theta(\mathbf{x})}{p_\theta(\mathbf{z}, \mathbf{x})} d\mathbf{z} \\ &= \int q_\phi(\mathbf{z}|\mathbf{x}) \left(\log p_\theta(\mathbf{x}) + \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}, \mathbf{x})} \right) d\mathbf{z} \\ &= \log p_\theta(\mathbf{x}) + \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}, \mathbf{x})} d\mathbf{z} \\ &= \log p_\theta(\mathbf{x}) + \int q_\phi(\mathbf{z}|\mathbf{x}) \log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{x}|\mathbf{z}) p_\theta(\mathbf{z})} d\mathbf{z} \\ &= \log p_\theta(\mathbf{x}) + \mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \left[\log \frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z})} - \log p_\theta(\mathbf{x}|\mathbf{z}) \right] \\ &= \log p_\theta(\mathbf{x}) + D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z})) - \mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) \end{aligned} \quad (2.15)$$

Следовательно:

$$\log p_\theta(\mathbf{x}) - D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z}|\mathbf{x})) = \mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \parallel p_\theta(\mathbf{z})) \quad (2.16)$$

Таким образом, из неравенства Йенсена получаем:

$$\ln p_\theta(x) \leq \mathcal{L}(\phi, \theta; x)$$

Базовым алгоритмом оптимизации вариационных моделей является ЕМ-алгоритм, состоящий из последовательного обновления скрытых представлений и максимизации правдоподобия с заданным параметрами.

Определение: **ЕМ-алгоритм** — алгоритм для нахождения оценок максимального правдоподобия параметров вероятностных моделей с скрытыми переменными θ . Аналитически шаги алгоритма записываются как:

- расчет матожидания при заданном на шаге t параметре $\theta^{(t)}$. Шаг обновления:

$$Q(\theta|\theta^{(t)}) = \mathbb{E}_{\mathbf{Z} \sim p(\mathbf{Z}|\mathbf{X}, \theta^{(t)})} [\log p(\mathbf{X}, \mathbf{Z}|\theta)] \quad (2.17)$$

- максимизации полученного выражения для нового шага $\theta^{(t+1)}$:

$$\theta^{(t+1)} = \arg \max_{\theta} Q(\theta|\theta^{(t)}). \quad (2.18)$$

2.2.1 Автокодировщики и модели потоков

Модель автокодировщика впервые была представлена в работе [42] для задач компрессии данных с возможными потерями. В качестве модели были использованы ассоциативные нейронные сети, выполняющие задачу снижения размерности входных данных.

Определение: Пусть $X \in \mathbb{R}^n$ и $Z \in \mathbb{R}^m$ евклидовы пространства, соответствующие закодированному и раскодированному сообщению. **Автокодировщик** является парой параметрических функций (E_{ϕ}, D_{θ}) , где $E_{\phi} : X \rightarrow Z$ задает функции кодирования входного сообщения, $D_{\theta} : Z \rightarrow X$ — декодирования.

Обучение модели выполняется путем минимизации среднеквадратичной разности между входом и выходом модели:

$$\mathcal{L}_{\text{AE}}(\theta, \phi) = \frac{1}{n} \sum_{i=1}^n (\mathbf{x}^{(i)} - f_{\theta}(g_{\phi}(\mathbf{x}^{(i)})))^2. \quad (2.19)$$

Такая модель особенно удобна для работы с изображениями, заданными сеткой скалярных выражений $\mathbb{R}^{n \times m}$. Современные методы с использованием квантильного представления позволяют сжимать изображения в десятки раз эффективнее классических алгоритмов кодирования [43].

Аппарат автокодировщика может быть усовершенствован путем учета распределения ошибки кодирования. В предметной области, преимущественно, используются параметрические распределения p_{θ} , поскольку они могут быть численно адаптированы под произвольную постановку при наличии достаточного объема данных. Поиск параметров выполняется путем вариации функционала:

$$\mathcal{L}_{\text{VAE}} = \log p_{\theta}(\mathbf{x}) - D_{\text{KL}}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p_{\theta}(\mathbf{z}|\mathbf{x})) \leq \log p_{\theta}(\mathbf{x}) \quad (2.20)$$

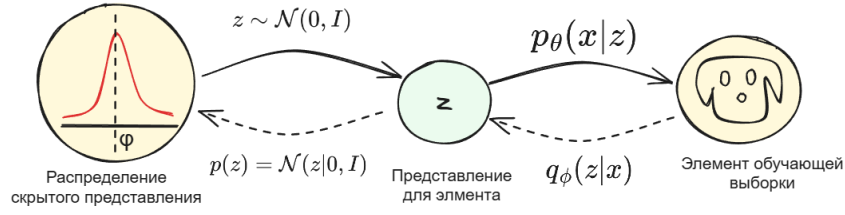


Рис. 2.7: Вариационный кодировщик задает ошибку кодирования в виде параметрического нормального распределения.

В работе [44] предложен численно эффективный учет ошибки, заданной нормальным распределением. Обучение происходит путем автоматического дифференцирования по параметрам распределения:

$$\begin{aligned} \mathbf{z} &\sim q_\phi(\mathbf{z}|\mathbf{x}^{(i)}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}^{(i)}, \boldsymbol{\sigma}^{2(i)} \mathbf{I}); \\ \mathbf{z} &= \boldsymbol{\mu} + \boldsymbol{\sigma} \boldsymbol{\epsilon}, \end{aligned} \quad (2.21)$$

где $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$. Функция оптимизации с учетом репараметризации запишется как:

$$\max_{\phi, \theta} \mathbb{E}_{\mathbf{x} \sim \mathcal{D}} [\mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z})], \quad (2.22)$$

с регуляризационным условием на дивергенцию $D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z})) < \delta$ для обучения декодировщика. Тогда оптимизационный Лангражиан \mathcal{L} с множителем β для условия примет вид:

$$\begin{aligned} \mathcal{L}(\theta, \phi, \beta) &= \mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - \beta(D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z})) - \delta) \\ &= \mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - \beta D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z})) + \beta \delta \\ &\geq \mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - \beta D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z})) \end{aligned} \quad (2.23)$$

Итоговая оптимизационная функция с гиперпараметром β записывается как:

$$\mathcal{L}(\phi, \beta) = -\mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) + \beta D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z})) \quad (2.24)$$

Вариационные кодировщики позволяют лишь выполнить оценку снизу на плотность вероятностной массы. Получения вероятности в явной форме можно выполнить, используя модели потоков.

Определение: Моделями нормализационных потоков называют генеративные модели, основанные на последовательных обратимых дифференцируемых преобразованиях ¹:

$$\log p_K(z_K) = \log p_0(z_0) - \sum_{i=1}^K \log \left| \det \frac{df_i(z_{i-1})}{dz_{i-1}} \right| \quad (2.25)$$

¹"Нормальность" потока связана с явным введением параметра сжатия фазового объема в выражение

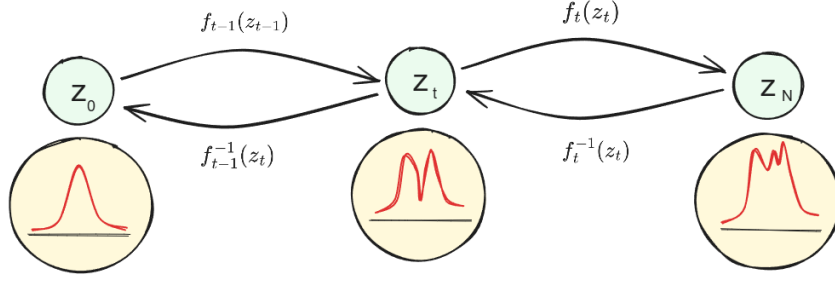


Рис. 2.8: Модели потока используют последовательные обратимые преобразования для приведения исследуемого распределения к известному.

Ключевым для численной разрешимости постановки является простота вычисления логарифма детерминанта и невырожденность $\det \frac{df_i(z_{i-1})}{dz_{i-1}}$. Для исследований класса функции с заданными требованиями, как правило, используются перестановки слагаемых и матричные преобразования.

Обратимые дифференцируемые преобразования позволяют использовать теорему о замене переменных для связи плотности вероятности в ходе преобразований:

$$\begin{aligned} \mathbf{z} &\sim \pi(\mathbf{z}), \mathbf{x} = f(\mathbf{z}), \mathbf{z} = f^{-1}(\mathbf{x}); \\ p(\mathbf{x}) &= \pi(\mathbf{z}) \left| \det \frac{d\mathbf{z}}{d\mathbf{x}} \right| = \pi(f^{-1}(\mathbf{x})) \left| \det \frac{df^{-1}}{d\mathbf{x}} \right|, \end{aligned} \quad (2.26)$$

где $\det \frac{\partial f}{\partial \mathbf{z}}$ — якобиан функции f .

В методе RealNVP [45] в качестве преобразования предлагается аффинное преобразование последних $D - d$ компонент вектора генерации:

$$\begin{aligned} \mathbf{y}_{1:d} &= \mathbf{x}_{1:d}; \\ \mathbf{y}_{d+1:D} &= \mathbf{x}_{d+1:D} \odot \exp(s(\mathbf{x}_{1:d})) + t(\mathbf{x}_{1:d}). \end{aligned} \quad (2.27)$$

Детерминант полученного выражения запишется как:

$$\mathbf{J} = \begin{bmatrix} \mathbb{I}_d & \mathbf{0}_{d \times (D-d)} \\ \frac{\partial \mathbf{y}_{d+1:D}}{\partial \mathbf{x}_{1:d}} & \text{diag}(\exp(s(\mathbf{x}_{1:d}))) \end{bmatrix}. \quad (2.28)$$

Авторы подхода NICE [46] предложили использовать одномерные сверточные слои, по сути являющимися функциями сдвига и масштабирования:

$$\begin{cases} \mathbf{y}_{1:d} &= \mathbf{x}_{1:d}; \\ \mathbf{y}_{d+1:D} &= \mathbf{x}_{d+1:D} + m(\mathbf{x}_{1:d}) \end{cases} \Leftrightarrow \begin{cases} \mathbf{x}_{1:d} &= \mathbf{y}_{1:d}; \\ \mathbf{x}_{d+1:D} &= \mathbf{y}_{d+1:D} - m(\mathbf{y}_{1:d}) \end{cases}. \quad (2.29)$$

Логарифм детерминанта преобразования в этом случае запишется как:

$$\log \left| \det \frac{\partial \text{conv2d}(\mathbf{h}; \mathbf{W})}{\partial \mathbf{h}} \right| = \log(|\det \mathbf{W}|^{h \cdot w}) = h \cdot w \cdot \log |\det \mathbf{W}| \quad (2.30)$$

2.2.2 Энергетические модели

Энергетические подходы приобрели популярность с публикацией прорывных работ по диффузионным порождающим моделям [41]. Ключевым отличием данного подхода является неявная работа с вероятностной массой с использованием ненормированного потенциала. Такой подход позволяет исследователям адаптировать физические модели для обучения и генерации данных. Так, например, процесс Ланжевена, описывающий смещение частицы с случайной составляющей, может быть использован для генерации из распределения, заданного исключительно потенциалом, без необходимости в численно затратной операции нормировки.

Определение: **Энергетическими** подходами в машинном обучении называют класс статистических моделей, параметризующих вероятность состояния согласно энергии E :

$$p(x) \sim \frac{\exp(-E(x))}{Z}, \quad (2.31)$$

где $Z = \int \exp(-E(x))$.

На практике энергетический потенциал задается через параметрическую функцию $E(\mathbf{x}, \theta, x)$, позволяющую выполнять адаптацию модели с помощью методов градиентного спуска [47].

Определение: **Диффузионные модели** представляют собой класс вероятностных моделей, использующих уравнение Ланжевена ?? для генерации элементов x из вероятного распределения p .

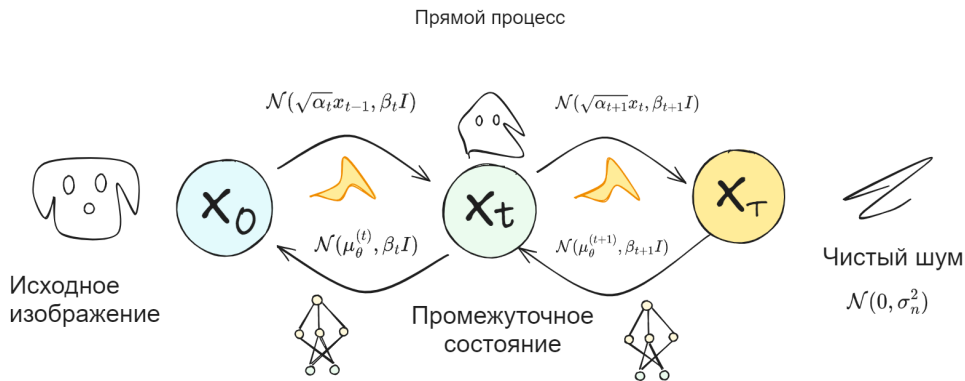


Рис. 2.9: Прямой процесс зашумления и обратный процесс коррекции ошибки в диффузионных моделях

В диффузионных моделях генерация выполняется в виде последовательных шагов, каждый из которых компенсирует ошибку начального изображения. Обучение исправлению ошибок выполняется путем предсказания шума между текущим и предыдущим шагом.

Прямым процессом диффузионной модели называется постепенное зашумление:

$$x_t = \sqrt{1 - \beta_t} x_{t-1} + \sqrt{\beta_t} z_t, \quad (2.32)$$

где $z_t \sim \mathcal{N}(0, \mathbf{I})$. Тогда итоговое вероятностное распределение:

$$q(x_T) = q(x_0)q(x_1|x_0) \dots q(x_T|x_{T-1}) = q(x_0)\mathcal{N}(x_1|\sqrt{1-\beta_1}x_0, \beta_1 I) \dots \mathcal{N}(x_T|\sqrt{1-\beta_T}x_0, \beta_T I) \quad (2.33)$$

В компактной форме:

$$\ln q(x_T) = \ln q(x_0) - \sum_{t=1}^T \frac{1}{2\beta_t} \|x_t - \sqrt{1-\beta_t}x_{t-1}\|^2 + C. \quad (2.34)$$

Обратный процесс состоит в построении цепочки: $\mathcal{N}(x_{t-1}|\mu(x_t, t), \Sigma(x_t, t))$, восстанавливающей элемент из шума:

$$p_\theta(x_T) = \mathcal{N}(x_T|0, I); p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}|\mu_\theta(x_t, t), \Sigma_\theta(x_t, t)). \quad (2.35)$$

Для введения параметрической модели используют нижнюю вариационную границу ELBO:

$$\mathbb{E}_{x_0 \sim q} \ln p_\theta(x_0) \geq \mathbb{E} \ln p_\theta(x_T) - \ln(). \quad (2.36)$$

Тогда функция ошибки запишется как:

$$L(\theta) = - \sum_{t=1}^T \mathbb{E}_{x_{t-1}, x_t \sim q} [-\ln p_\theta(x_{t-1}|x_t)] + \mathbb{E}_{x_0 \sim q} \mathbf{D}_{KL}(q(x_T|x_0) \parallel p_\theta(x_T)). \quad (2.37)$$

На практике функция ошибки упрощается до предсказания шума:

$$\mathbb{E}_{x_0 \sim q; z \sim \mathcal{N}(0, I)} [\|\varepsilon_\theta(x_t, t) - z\|^2]. \quad (2.38)$$

2.3 Обработка естественного языка

Анализ естественного языка это междисциплинарная дисциплина, включающая теорию алгоритмов, лингвистику и машинное обучение.

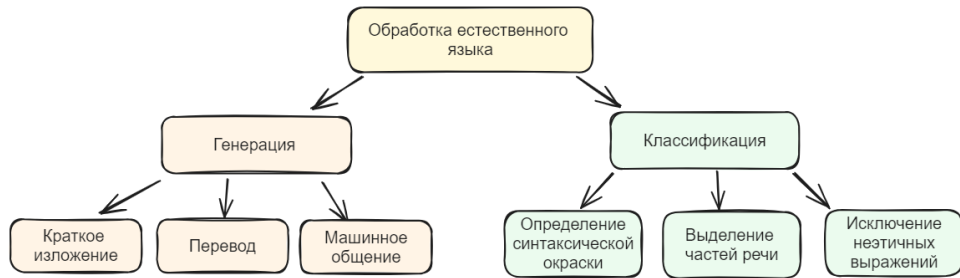


Рис. 2.10: Таксономия современных подходов обработки естественного языка.

Формальные языки широко используются в математике, логике, лингвистике и компьютерных науках. В программировании, например, формальные языки вклю-

чают языки программирования и описания данных. Синтаксис таких языков строго определён для обеспечения корректности и детерминированности выполнения программ. В областях обработки естественного языка приобрела популярность аналитическая форма механизма внимания [37], сделавшая возможным создание больших лингвистических моделей [48].

2.3.1 Формальные языки

В данной секции будет рассмотрена формальная вычислительная теория языка. Подход был предложен Н. Хомски и представлен его работе "Синтаксические структуры" [38]. Направление изучает алгоритмические методы по изменению морфемного состава слова и формированию представления о связи слов в тексте.

Введем ключевые предметные определения, позволяющие формализовать анализ и синтез предложений в естественном языке, что важно для многих приложений в области обработки естественного языка и вычислительной лингвистики.

Определение: Формальный язык \mathcal{L} является совокупностью:

- алфавита Σ — конечного множества символов, из которых строятся строки языка;
- строки N — последовательности символов N из алфавита Σ ;
- грамматики Ξ — набора правил, которые определяют, какие строки являются допустимыми в языке.

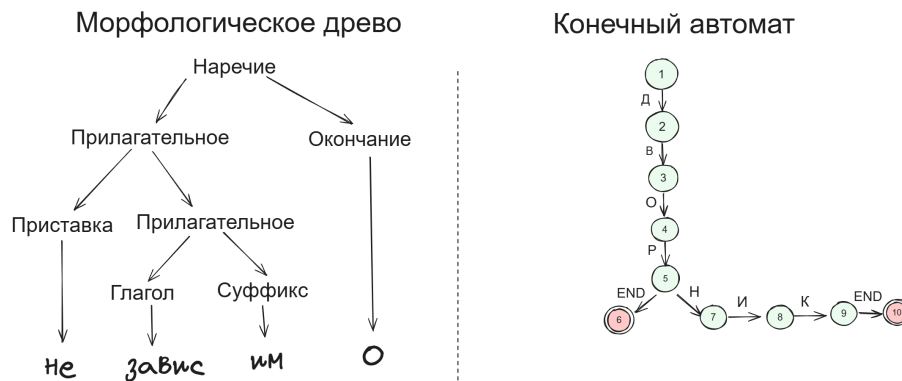


Рис. 2.11: Морфологическое дерево задает совместную иерархическую структура слова в виде конечного автомата.

Для обработки языка используются конечные автоматы, позволяющие выделять из набора символов строки в соответствии с запросом. Распространенным примером являются регулярные выражения, широко используемые для выделения ключевых слов в информационном поиске [49].

Для обработки естественного языка методами формальных языков из слов выделяются нормальные формы и словоизменительные парадигмы, являющиеся образцами для склонения или спряжения. Такие структуры создаются лингвистами и

позволяют путем склонения, спряжения и добавления приставки получать необходимые формы слов. Открытый российский проект OpenCorpora предоставляет наиболее полный словарь форм современного русского языка, описанного в работе [50].

Определение: **Морфологическим анализом** называют процесс разложения слова w на его морфемы, такие как префикс P , корень R и суффикс S , из словаря Σ

Определение: **Морфологическим синтезом** называется функция f , формирующая слова w из леммы l и морфологических характеристик m , таких как число, род и падеж.

Автоматическая обработка предложений выполняется путем выделения синтаксических связей.

Определение: **Синтаксическое дерево** описывается как упорядоченное дерево T с вершинами N , соответствующими синтаксическими категориями (например, S, NP, VP), либо терминальному символу.

Получаемое дерево в качестве листьев представляет слова в предложении, а корневой узел r , включает структуру предложения. Тогда дерево разбора T для строки $w = w_1 w_2 \dots w_n$ запишется как четверка $T = (N, E, r, L)$, где:

- N — конечное множество узлов;
- $E \subseteq N \times N$ — множество ребер, каждое из которых соединяет пару узлов (родитель — потомок);
- $r \in N$ — корень дерева;
- $L = \{w_1, w_2, \dots, w_n\} \subseteq N$ — множество листьев, соответствующих словам в предложении.

2.3.2 Представление языка в виде вложений

Определение: **N-граммы** представляют собой последовательности из n элементов в тексте или последовательности символов, где n обозначает количество элементов в последовательности. Элементы могут быть символами, словами или более крупными фрагментами текста в зависимости от контекста применения. Анализ n-грамм является важным методом в обработке естественного языка (Natural Language Processing, NLP) для изучения частотности последовательностей слов или символов в текстовых данных.

Формально, n-грамма ngram_n длины n в тексте T определяется как последовательность n элементов, где каждый элемент x_i может быть символом, словом или другими составляющими текста:

$$\text{ngram}_n = (x_1, x_2, \dots, x_n)$$

Использование n -грамм в анализе текста позволяет оценивать частотность последовательностей слов или символов и анализировать лингвистические характеристики текста, такие как структура, стиль и тематика. Кроме того, n -граммы могут использоваться в задачах моделирования языка, предсказания следующего слова в предложении, а также в машинном переводе и других приложениях обработки естественного языка.

Определение: **Токенизация** — процесс в котором текст разбивается на токены. Это позволяет применить лемматизацию к каждому слову в тексте независимо от контекста. Лемматизация часто используется в различных областях NLP, включая информационный поиск, анализ тональности, машинный перевод и другие.

Определение: **Векторное вложение** (*англ* embedding) — векторное представление \mathbb{R}^N слова w , проявляющее семантические и синтаксические свойства при операциях сложения и взятия косинусного расстояния.

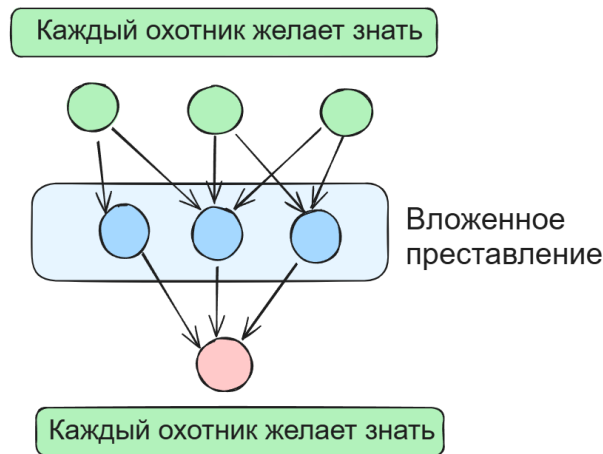


Рис. 2.12: Векторное вложение позволяет выполнять семантические операции

Практически востребованной оказалась дистрибутивная гипотеза [51], легшая в основу алгоритма Word2Vec [52]:

$$\cos(\theta) = \frac{w_1 \cdot w_2}{\|w_1\| \|w_2\|}. \quad (2.39)$$

Векторные вложения слов играют важную роль в генеративном моделировании естественного языка, так как они позволяют моделям представлять слова в виде непрерывных числовых значений, которые могут быть использованы как входные данные для алгоритмов машинного обучения. Это позволяет моделям эффективно сопоставлять зависимости между словами и генерировать семантически богатые и лингвистически осмысленные тексты. Известны и другие подходы GloVe [53], FastText [54], улучшающие базовый подход работой с разбиением слова на части. Появление трансформерной архитектуры нейросетей позволило улучшить результаты семантического представления [55]. Для сравнения стали доступны не только слова, но целые предложения и параграфы. Результат был достигнут путем специального подхода к обучению, заключающегося в случайном исключении до 20% слов во фрагменте текста,

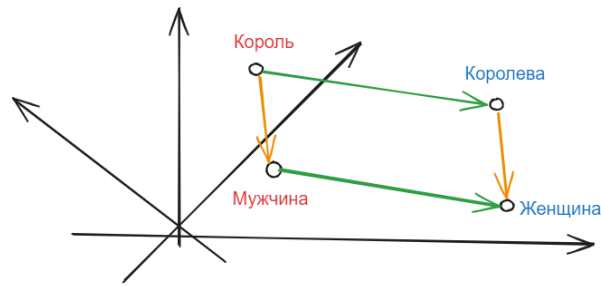


Рис. 2.13: Векторное представление позволяет выполнять семантические операции сложения и вычитания

который требуется восстановить. Сложность задачи заключается в обширном окне контекста, необходимого для учета, при заполнении пропусков. Помимо восстановления слов, модель BERT также обучалась на задачах классификации и выделения ключевых слов в предложении. Совмещение техник обучения позволяет улучшить метрические показатели моделей до нескольких десятков процентных пунктов [56].

2.3.3 Большие языковые модели

Постановка авторегрессионного моделирования, включающая предсказание следующего слова по предшествующему контексту, может быть использована для решения ключевых для обработки языка задач перевода, пересказа и выделения ключевых слов. Данная техника позволила исследователям [57] обучить общую модель работы с текстом GPT (генеративный обучающий трансформер). Отличительной особенностью исследования была его масштабность. Модель обучена на объеме более триллиона слова, полученного с помощью выделения естественного языка из источников в internet. Таким образом, в краткие сроки были приобретены навыки распознавания и генерации на более чем ста языках мира и базовые знания об естественных и социальных науках. Для прохождения испытаний по метрической оценке навыков модели были использованы корпуса высокого качества.

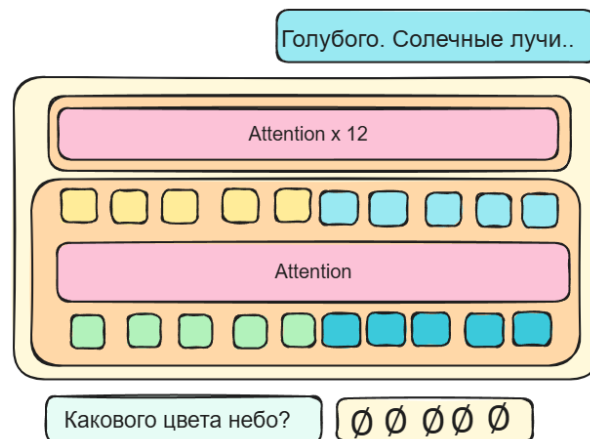


Рис. 2.14: Большая языковая модель GPT обучается как авторегрессионная модель на задаче предсказания следующего слова

Обучение языковых моделей для задач ассистирования разделяется на три этапа [58]:

1. предобучение, включающее формирование представлений о синтаксической структуре языков мира, модели мира и разнообразии формы выражения мысли;
2. адаптацию под задачу, знакомящую модель с терминами и связями между ними;
3. инструкционное дообучение, заключающееся в обучении модели выполнять задачи, необходимые в практике специалистов.

В ходе первого этапа модель обучается на обширных разнородных корпусах текста. Предобучение требует значительных вычислительных ресурсов, не всегда доступных в образовательных и научных приложениях, поэтому в исследованиях используются открытые языковые модели, опубликованные коммерческими организациями [59] [60] [touvron2023llama]. Второй этап обучения выполняется на экспертно отобранном корпусе предметных документов. В сфере образования таковыми являются цифровая учебная литература, вопросы и ответы на предметных форумах и научные статьи. Данные для эффективного обучения должны иметь максимально возможное качество: необходимо исправлять ошибки набора и неструктурированные в соответствии с правилами языка фрагменты текста. В противном случае, модель может запомнить неверные правила форматирования [61]. Финальный этап начинается с обучения на стандартном корпусе инструкций, включающих базовые задачи структурированных определений, изложение информации в требуемой стилистической форме и поиск ключевых слов [62]. На данном этапе модель уже способна эвристически решать большое число задач, но может ошибаться в специфических постановках. Для их устранения этих недостатков эксперты корректируют ошибочные решения модели. Таким образом модель одновременно приобретает навыки правильного решения и получает информацию о видах нежелательных ответов.

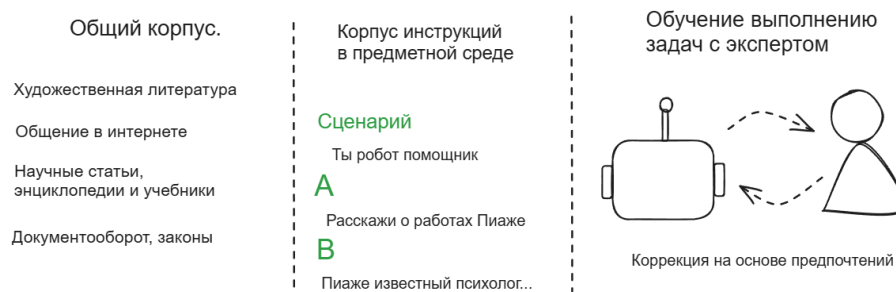


Рис. 2.15: Обучение разбито на три ключевых этапа: подготовка, адаптация на корпусе и экспертная коррекция

Для оценки способностей языковых моделей создаются специальные системы тестирования, количественно оценивающие способности моделей к:

- владение языками мира [63];
- способности к эффективному ведению диалога [64];
- пониманию животного и растительного мира, социальных правил и законов [65] [66];
- решения аналитических и абстрактных логических задач [67] [68].

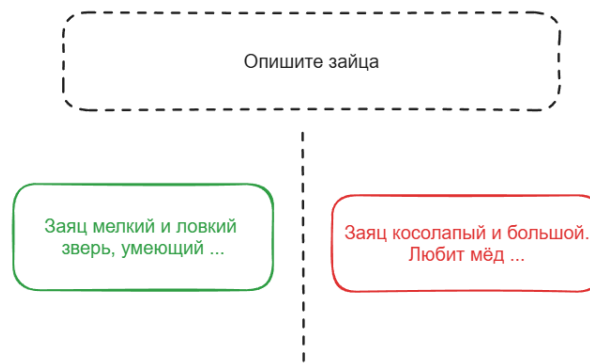


Рис. 2.16: Адаптация к задачам выполняется путем взаимодействия с пользователем и работой с экспертом

Отметим, что современные большие языковые модели не гарантируют корректное исполнение даже базовых арифметических операций. В обзоре [61] показано, что подобные проблемы возникают во многих строгих постановках, где соблюдение формы требует более глубокого чем эвристического понимания задачи:

- написание исполняемого языком программирования кода;
- стратегические игры [69];
- соблюдение корректности математических выражений при алгебраических преобразованиях.

Для решения проблемы исследователи предложили использование инструментов, которые используются моделью для качественного выполнения инструкции. На данный момент сложились следующие подходы:

- обращения к информационным системам (от *англ.* RAG - retrieval augmented generation)[70];
- работа с средой исполнения программного кода [71];
- генерация сопровождающей иллюстрации [72].

Глава 3

Психометрия в образовании

Создание и передача научных знаний — основа общественного развития [73]. Повсеместное распространение знания в профессиональных коллективах позволяет вести эффективную совместную деятельность, автоматическим образом выполнять рутинную работу и системно оценивать риски новых проектов. Одной из важных проблем передачи знаний является их объективная оценка. Современный путь разрешения проблемы заключается в количественной оценке знания с применением контрольно-методических материалов. Анализ и совершенствование систем оценки знаний изучается дисциплиной психометрии. Специалисты в этой области создают и апробируют измерительные инструменты, включающие опросники, тесты и методики описания личности. Также изучаются модели отклика на контрольно-методические и образовательные материалы. Направление включает в себя три основные исследовательские задачи:

- создание инструментов и построение процедур измерения;
- создание новых математических моделей вероятности наблюдения определенных элементов поведения;
- развитие и усовершенствование теоретических подходов к измерению.

В главе описаны современные подходы к созданию систем тестирования. Также представлен раздел с изложением общих положений из психологических теорий Льва Выгосткого, Жана Пиаже и Михаи Чиксентмихайи, задающие порядок воспитательно-образовательного процесса и рекомендации по выбору задач оптимального уровня сложности.

3.1 Значение образования

Введение изменений в методику преподавания требует обоснования, выраженного в положительном изменении общественного блага. Одним из подходов к его определению связан является функция суммарной пользы или счастья для всех членов общества. Термин утилитарности и его экономически-философская трактовка

описана в работах Иериямья Бентам [74] в XVIII веке. Популяризация и развития подхода связано с появлением теории игр в 20 веке [75] [76] [77]. В теории игр мотивы действий связывают с максимизацией скалярного потенциала называемого функцией утилитарности. Полученные теорией игр результаты применяются при создании систем голосования [78], проектировании транспортных систем [79] и организации плана операций в сфере медицины [80].

Определение: **Функция утилитарности** это концепция, разработанная в области экономики и используемая в теории принятия решений. Она описывает способность индивида оценивать полезность или степень удовлетворения от различных вариантов действий или состояний.

Существует два принципиальных подхода к определению функцию утилитарности:

- в ординальном подходе утилитарное значение устанавливается относительно других альтернатив в порядке предпочтений;
- в абсолютном — утилитарное значение измеряется в конкретных единицах, что позволяет проводить количественные сравнения между альтернативами и оценивать уровень удовлетворения количественно.

Определение: **Равновесие по Нэшу** [75] — состояние, в котором ни один игрок не может получить дополнительную выгоду от своих измененных действий, если другие игроки продолжают свои стратегии.

$$\forall s_i \in S_i \rightarrow u_i(s_i^*, s_{-i}^*) \geq u_i(s_i, s_{-i}^*), \quad (3.1)$$

где u - функция утилиты, S_i — заданный набор игроков, s_i^* - решение игрока i , соответствующее равновесию по Нэшу, $s_{-i} = S \setminus \{i\}$ — обозначение исключения элемента множества i из множества S .

Изучение равновесий по Нэшу в математических постановках позволяет задавать оптимальные системы правил, пресекающий недобросовестный сговор. Классическим примером равновесия по Нэшу является дилемма заключенного, удобная к записи в виде матрицы решений двух игроков (рис. 3.1). Модель определяет функцию полезности для совместных решений игроков. При оптимальном задании правил наказания сообщника предпочтут не скрывать злодеяния сообщника.

$$\forall s_i \in S_i u_i(s_i^*, s_{-i}^*), \quad (3.2)$$

3.1.1 Механизм

Оптимальное распределение ресурсов в аналитических постановках изучается с помощью дизайна механизмов, предложенных и развитых в работах нобелевского ла-

		Игрок 2	
		Кооперация	Обвинение
Игрок 1	Кооперация	3 6	6 6
	Обвинение	6 6	9 9

Рис. 3.1: Дилемма заключенного.

урейта Л. А. Гурвича [77]. Теоретический аппарат механизма широко распространен в экономике и социальных науках, и используется в создании форматов справедливой и эффективной коллективной деятельности, включающей аукционы, выборы и право благосостояния.

Теория механизмов изучает проблемы доказательной эффективности организации рабочих процессов с учетом скрытых потребностей исполнителей. Исходя из предположений о рациональности агентов, механизм определяет правила их поведения для приведения коллективной системы в состояние равновесия с оптимальной производственной функцией. Аналитическое описание механизма выполняется через теорию байесовых игр с неполной информацией.

Определение: **Байесова игра** это набор исходов (N, O, Θ) таких что:

- N конечное множество агентов n ;
- O множество исходов;
- $\Theta = \Theta_1 \times \Theta_2 \dots \Theta_n$ множество;
- $u = (u_1, \dots, u_N)$, где $u_i : O \times \Theta \rightarrow \mathbb{R}$ функция полезности для игрока i .

Определение: **Механизм** для байесовой игры это пара (A, M) , где:

- $A = A_1 \times \dots \times A_n$ набор действий доступный агенту i ;
- $M : A \rightarrow \Pi(O)$ соединяет действия с распределением возможностей.

В заданной игре игроки стремятся к поиску стратегии, ведущей к безусловному преимуществу независимо от действий прочих игроков. Такие стратегии называются доминантными и являются предметом изучения исследований. анализе доминантных стратегий и равновесных состояний, к которым они приводят.

Определение: В заданной байесовой игре, механизм является **воплощением доминантной стратегии** социального выбора функции C , если для любого вектора

полезности u у игры есть равновесие в доминантной стратегии, и для любого равновесия a^* выполняется $M(a^*) = C(u)$.

Теория механизмов используется в постановках с асимметричной информацией и достаточно большим числом игроков, что, как правило, исключает возможность наивного перебора возможных исходов.

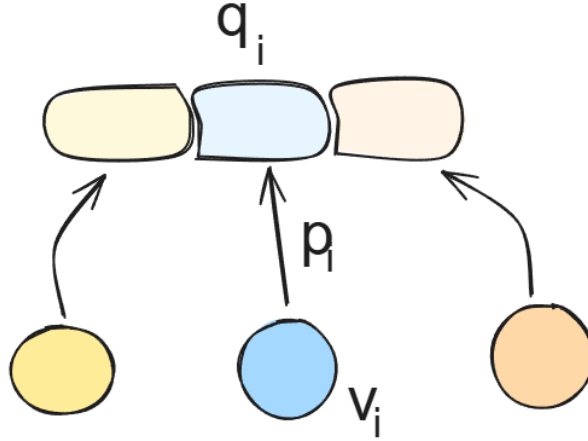


Рис. 3.2: Механизм определяет распределение ресурса \mathbf{x} и платежа \mathbf{p} в зависимости от ставок \mathbf{b}

Каждый игрок i имеет четверку (v_i, b_i, p_i) , соответствующую:

- случайная величина v_i с функцией распределения F . Обозначим полученный квантиль распределения $q(v_i) = 1 - F(v_i)$;
- ставка на лот b_i ;
- доля полученного ресурса x_i ;
- p_i итоговый платеж.

Тогда функция полезности для каждого игрока i при заданном векторе ставок запишется как:

$$u_i(\mathbf{b}; q_i) = v(q_i) \cdot x_i(\mathbf{b}) \quad (3.3)$$

Определение: Байес-Нэшевым равновесием называют результат распределения, в котором каждый участник аукциона максимизирует свою функцию полезности u_i как матожидание от квантилей q_{-i} :

$$\max_{\mathbf{q}, \mathbf{b}} E_{\mathbf{q}_{-i}} [u_i(\mathbf{b}(\mathbf{q}); q_i)] \quad (3.4)$$

При изучении свойств механизмов разделяется фактически (от *англ.* ex-post) распределенное число ресурсов $\tilde{x}_i(\mathbf{q}) = x_i(\mathbf{b}(\mathbf{q}))$ на игрока i и в среднем ожидаемое (от *англ.* interim) $\hat{x}_i(q_i) = E_{\mathbf{q}} [\tilde{x}_i(\mathbf{q}) | q_i]$. Специфика связана с тем, что среднее ожидаемое доступно к расчету прочими игроками и потому может учитываться при принятии решений [81].

Лемма: Набор имеет Байес-Нэшево равновесие при условии:

- $\hat{x}_i(q_i)$ монотонно убывает по q_i ;
- $\hat{p}_i(q_i) = \mathbb{E}_{\mathbf{q}}[p_i(\mathbf{b}(\mathbf{q}))|q_i] = v(q_i) \cdot \hat{x}_i(q_i) + \int_{q_i}^1 \hat{x}_i \cdot v'(z)dz$, где \hat{p}_i соответствует ожидаемому платежу игрока.

Заданные условия достаточны для достижения уникального равновесия \mathbf{b} . Доказательство приведено в [82]. Одним из примером механизма являются рейтинг-системы, широко распространенные в спортивных интеллектуальных соревнованиях.

Определение: Рейтинг-система — модель, ранжирующая n участников в единый линейный порядок $i_1 \succ i_2 \succ \dots i_n$ по данным сравнений небольших подмножеств этих игроков.

Статистический подход к введению рейтинг-моделей заключается [83] в введении вероятности победы в ходе отбора объекта i над объектов j :

$$p(x_i \succ x_j) = \zeta(x_i, x_j), \quad (3.5)$$

где функция ζ , принимает на вход скалярные потенциалы x_i и x_j , соответствующие силе объектов.

Вид функции сравнения ζ , в общем случае, не определен и выбирается на основании статистических исследований. В работе [84] Арпад Эло на основании данных шахматной федерации США предложил для стратегических игр использование логистической функции:

$$p(x_i \succ x_j) = \frac{1}{1 + \exp(-\frac{x_i - x_j}{\sigma})}, \quad (3.6)$$

где дисперсия σ фиксирована для всех игроков. Отличительной чертой такой модели является гарантия победы игрока с значимо большим навыком $x_i - x_j \gg \sigma$.

Отметим, что система Эло также задает правило обновления рейтинга обеспечивающих эволюций системы рейтинга во времени. Обновление рейтинга R игрока выполняется по исходу игры S в соревновании согласно правилу:

$$R' = R + \sigma(S - p), \quad (3.7)$$

где R' — новый рейтинг игрока, p — вероятность победы в матче согласно 3.6. Новые очки рейтинга поступают в систему с появлением новых участников. В шахматной практике игроки начинают с рейтингом 1600.

3.1.2 Систем наград

Награды могут иметь разное происхождение, предназначение и формы. Например, в российском школьном образовании отличная учеба награждается медалями,

а физкультурно-спортивные достижения отмечаются знаками отличия. Награды поощряют вовлеченную учебу и совокупно позволяют выделять талантливых учащихся.

Известные компании, предоставляющие инструменты разработки, такие как GitHub¹ и Google Cloud², также используют систему наград в виде знаков — *бейджей* для поощрения пользователей к эффективному использованию ресурсов платформы и прохождению сертификации. Предполагается, что наличие таких наград может стать причиной для делового сотрудничества или предложения работы. Другим примером является платформа Stack Overflow³, на которой обсуждаются вопросы, преимущественно, технической тематики. Согласно ее правилам бейдж присваивается за наилучший ответ на популярный вопрос. Такая награда поощряет положительную конкуренцию, и мотивирует конструктивные и этичные ответы [85].



Рис. 3.3: Упорядоченная система наград

Теория бейджей основана на системе упорядоченных наград. Опишем постановку для линейного упорядоченного набора бейджей $m > m - 1 > \dots > 0$ и n пользователей системы с оценкой полезности бейджа $S(t)$.

Определение: Механизм бейджа задается как функция $r : \mathbb{R}_+ \times \mathbb{R}_+^{n-1} \rightarrow \{1, \dots, m\}$ от вкладов игроков b_i :

$$S(t_i) = \frac{b_i}{v_i}. \quad (3.8)$$

Задача механизма обеспечить максимизацию суммарного вклада участников:

$$\max_{P(m)} \sum_{i=1}^n b_i. \quad (3.9)$$

Определение: **Механизмом отсечки по порогу** называется механизм распределения бейджа $j \in \{0, \dots, m\}$, исходя из преодоления линейно упорядоченных порогов $\theta = (\theta_1, \dots, \theta_m)$, таким образом, что $b_i \in [\theta_j, \theta_{j+1})$.

Механизм порогов прост в использовании и понимании для агентов. Предварительное задание уровня награды позволяет планировать путь к достижению, сосредотачиваясь на индивидуальном развитии. Тем не менее такой подход не поддерживает соревновательный дух, потому не всегда мотивирует кооперацию среди агентов. Альтернативой является задание рейтинга согласно порядку вклада:

Определение: **Механизмом таблицы лидеров** называется механизм распределения бейджей $j \in \{0, \dots, m\}$, в порядке убывания по вкладу b_i . Число бейджей и

¹<https://github.com/>

²<https://cloud.google.com/>

³<https://stackoverflow.com/>

участников должно быть одинаково.

Такой подход естественно ограничивает кооперацию: агенты стремятся внести больший вклад чем прочие, максимизируя функцию утилиты. Тем не менее такой подход стимулирует лишь относительный рост, что может быть недостаточным в задачах, требующих постоянной отдачи от участников. Также слишком большое число бейджей неудобно для разработки на практике.

Работа [86] предлагает объединение двух подходов для разрешения перечисленных проблем.

Определение: **Механизмом таблицы лидеров с порогом отсечки** называется механизм распределения бейджей в порядке убывания по вкладу b_i при условии $b_i > \theta$, где θ — уровень отсечки.

Такая постановка одновременно стимулирует относительный и абсолютный рост. Единый порог θ вводится исходя из простоты анализа постановки.

Теорема: *Об оптимальном механизме распределения наград* [86] Оптимальный порог отсечки θ для таблицы лидеров с порогом отсечки задается как $\theta = v(k^*)S_n(k^*)$, где $S_n(q_i) = \sum_{\nu=0}^{n-1} S\left(\frac{\nu}{n-1}\right) \beta_{\nu,n-1}(q_i)$ — n полином Бернштейна и k^* — пороговый квантиль полезности, заданный уравнением $b(k^*) = v(k^*)S_n(k^*)$. Заданный порог обеспечивает уникальное равновесие Байеса-Нэша с наибольшей суммой вкладов $\sum_i b_i$.

3.2 Структура методического материала

Дидактический педагогический метод — наиболее распространенный метод обучения, основанный на закономерностях усвоения знания и формирования убеждений. Такой подход задает системные практики организации учебно-познавательной деятельности учащихся, определяет объем и содержание занятия. Ключевым инструментом такого подхода является количественная оценка успеваемости, выводимая на основании контрольных работ. Решение об оценке может выноситься как преподавателем, так и автоматически, с использованием алгоритма. Оценка решает множество задач:

- мотивирует обучающегося к познанию;
- определяет перспективные направления для дальнейшего обучения для родителей;
- позволяет сравнить различные подходы к изложению материала для педагога.

Известным представителем дидактического подхода является психолог и педагог Б. Блум. В своей работе [87] Блум изучает проблемы массового образования. Согласно его исследованиям, обучающиеся по программам индивидуального образования имели результаты статистически лучше посещающих общеобразовательные учреждения. Для разрешения проблемы автор, опираясь на анализ педагогической практики своих ассистентов, выделяет наиболее успешные подходы (табл. 3.1).

Объект изменения	Название практики	Результат
Учитель	Курсы повышения специализации	2.00
Учитель	Обеспечение	1.2
Ученик	Пороговый балл прохождения курса	1.00
Учитель	Предоставление частичного решения, подсказки	1.00
Учитель, Ученик	Вовлеченность в занятие	1.00
Ученик	Задание строго время выполнения задания	1.00
Ученик	Упор на навыки письма и чтения	1.00
Домашнее окружение	Обучение с родителями	0.80
Учитель	Балловая оценка домашнего задания	0.80
Учитель	Поддержание духа класса	0.60
Ученик	Отбор учащихся	0.60
Домашнее окружение	Изменение домашней обстановки	0.50

Таблица 3.1: Влияние образовательных практик согласно [87]

Противопоставлением дидактическому методу обучения является проблемно-ориентированный подход. Эмпирический подход акцентируется на активном участии обучающегося в исследовании и поиске знаний в виде решений проблем и практической деятельности. Этот подход поддерживает самостоятельное мышление, исследовательские навыки и обучение через самостоятельное осмысление опыта. Эмпирический подход часто ассоциируется с *активными методами обучения*, описанными в работах Кругликова [88]. В таком подходе учащиеся самостоятельно формулируют и анализируют проблемы, разрабатывая стратегии исследования. К методам обучения относится регулярное решение актуальных постановок, пришедших из индивидуальной практики, организация исследовательских проектов с акцентом на командной работе.

3.2.1 Психологические подходы к описанию интеллектуального развития

Одной из наиболее проработанных научных психологических теорий интеллектуального развития является теория когнитивного развития, разработанная Жаном Пиаже [4]. Когнитивное развитие происходит в форме стадий, соответствующих развитию интеллектуальных способностей. Таковыми являются все виды мыслительных процессов, таких как восприятие, память, формирование понятий, решение задач, воображение и логика.

Этапы развития детской психики по Жану Пиаже:

1. Этап сенсомоторного интеллекта соответствует возрасту от рождения до 2 лет. Он характеризуется развитием моторной памяти и овладением основными чувствами: зрением, слухом и балансом.
2. Этап дооперационального представления происходит с двух до семи лет. На этой стадии дети еще не могут применять конкретные когнитивные операции,

не могут использовать математический счет и язык для практических целей. Объекты мира представляются идеальными согласно желаниям ребенка. Тем не менее текущих навыков достаточно для самостоятельного ухода за собой и одеждой, совместной социальной деятельности и детского творчества.

3. Период конкретных операций от семи до одиннадцати лет. Ребенок способен на дедуктивное и индуктивное мышление.
4. Конкретное и абстрактное логическое мышление соответствует ментальному развитию взрослого человека. Подросток понимает литературный язык, способен на логическое оформление творческой мысли.

Пиаже обосновывает свою теорию этичными экспериментами, легко доступными к воспроизведению педагогами. Инструментами научного эксперимента являются зеркала, куклы, пластилин и игры для детей. Изучаются реакция ребенка на сокрытие куклы за тканью, понимание сохранения материи на примере лепки и осознание своей личности при наблюдении за отражением в зеркале.

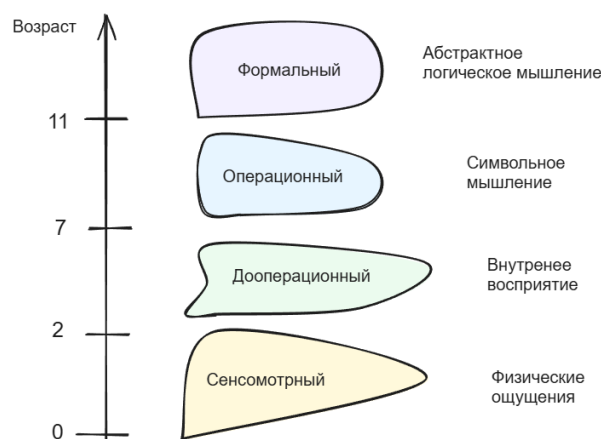


Рис. 3.4: Этапы развития детской психики по Жану Пиаже [4]

Лев Семёнович Выготский оспаривает этот подход, приводя в пример детей, которые в ходе занятий со взрослыми продемонстрировали высокий уровень освоения шахмат 3.5. Выполненные Выготским эксперименты стали основой направления культурно-исторической психологии, признанной в мировом профессиональном сообществе. Согласно Выготскому образование имеет социальный и целенаправленный характер и связано с развитием *высших психических функций* — сложных самоорганизующихся процессов в деятельности человека, опосредованных знаками по своему строению и произвольных по способу своего осуществления. В качестве таких навыков Выготский определяет мышление, речь, произвольное внимание, волю, воображение и др.

Определение: Зона ближайшего развития — расхождение между уровнем актуального психического развития ребёнка и уровнем его потенциального развития, достигаемого при решении задач в сотрудничестве со взрослым или сверстниками.

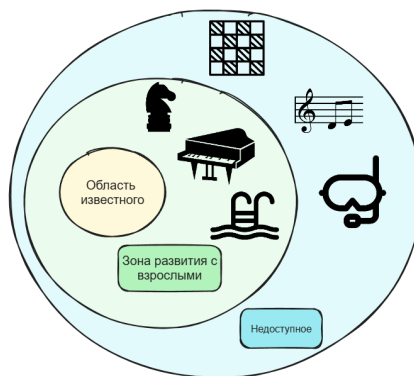


Рис. 3.5: Иллюстрация границ зон развития согласно Л.С. Выготскому

Согласно Выготскому ключевым для образования является процесс рефлексии учащегося. Такой процесс включает в себя осознанный анализ своего учебного опыта, рассмотрение своих мыслей, чувств и действий в контексте обучения. Механизмом рефлексии является *интериоризация*. Процесс заключается в переходе внешней речи во внутренний способ мышления. Отметим, что процесс возможен когда ребёнок уже способен планировать свою деятельность.

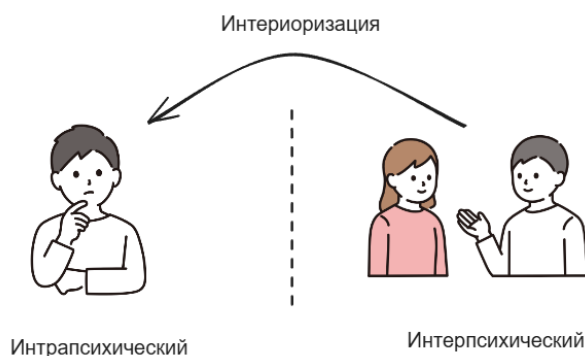


Рис. 3.6: Рефлексия по Выготскому [5]

Рефлексия способствует развитию метакогнитивных навыков:

- оценки своей деятельности и планирования;
- осознания ценности знания и социума;
- ведения высшей когнитивной деятельности по организации социальной деятельности.

Такие навыки позволяют справляться с практическими заданиями, обучать и осваивать неизвестные области знания, выявлять пробелы в знаниях и развивать критическое мышление.

Определение: **Поток** (от лат. *influent*) по Чиксентмихайи — это психическое состояние, характеризующееся полной вовлеченностью и концентрацией на текущей деятельности, сопровождающееся ощущением контроля, потерей чувства времени и высокой внутренней мотивацией.

Теория потока активно используется в играх [6] и программах непрерывного обучения [89] для задания оптимальной сложности образовательной программы.

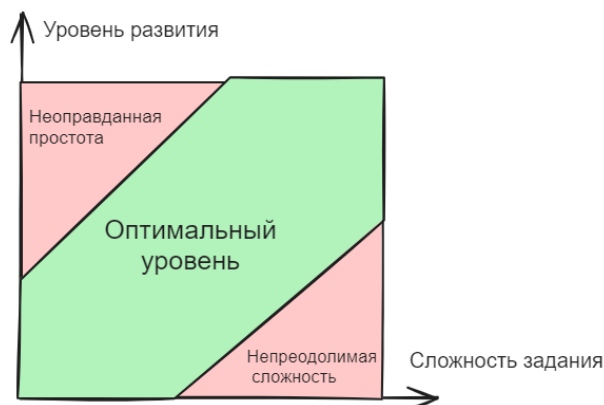


Рис. 3.7: Модель оптимальной сложности согласно Михаяу Чиксентмихайи

Характеристиками потока согласно [90] являются:

- Ясные цели и обратная связь. Деятельность имеет чёткие цели и предоставляет немедленную обратную связь.
- Баланс между сложностью задачи и навыками. Задача должна быть достаточно сложной, чтобы представлять собой вызов, но при этом соответствовать уровню навыков человека, чтобы не вызывать скуки и тревоги.
- Слияние действия и сознания. Человек настолько поглощён деятельностью, что все его действия становятся почти автоматическими и интуитивными.
- Потеря самосознания. В потоке человек забывает о себе, своих заботах и тревогах, полностью сосредоточившись на задаче.
- Искажение восприятия времени. Время может казаться летящим быстро или, наоборот, замедляться; часы могут пройти как минуты.

Определение: Гипотеза о врожденных знаниях (от *англ.* Innateness hypothesis) — теория в лингвистике и когнитивной науке, которая утверждает, что способность к языку является врождённой характеристикой человеческого мозга.

Гипотеза постулирует наличие:

- врождённого языкового аппарата, способствующего естественному освоению языка;
- универсальной грамматики, состоящей из набора грамматических принципов и структур, присутствующих во всех языках и врождённых для каждого человека;
- основных аспектов языковой способности, являющихся врождёнными и не зависящими от окружающей среды.

Гипотеза имеет статистические подтверждения, описанные в работах Хоам Номски, использовавшего статистический аппарат синтаксических деревьев для анализа массивного корпуса естественного языка [91][92].

3.3 Методическая задача

Основой обучения является последовательное усвоение и обсуждение образовательных материалов с их закреплением путем самостоятельной работы. Как правило, практика организована в виде решения задач с четко заданной целью, интуитивно понятной проблемой и сроком выполнения. При составлении образовательной задачи необходимо учитывать, что она должна:

- предоставлять возможность для развития критического мышления и применения знаний на практике;
- быть структурированной;
- иметь ясные и объективные критерии оценки.

По ответу задачи разделяются на открытые, предполагающие развернутый ответ с демонстрацией приобретенных знаний, и закрытые, подразумевающие лишь выбор правильного ответа. Современное образование активно использует оба подхода, объединяя их в единую контрольно-методическую работу, представляющую многогранную систему проверки знаний с заданными критериями оценки. Для анализа результатов прохождения работы используются статистические методы оценки знания, учитывающие случайность в измерениях [93]. Специалисты предполагают, что знание подлежит лишь косвенному измерению, путем проведения тестирования или конкурсного отбора. Для учета погрешности в измерениях этих инструментов теория использует модели заданий и умений обучающихся. Наиболее совершенной теорией на данный момент считается теория отклика на задание (от *англ.* Item Response Theory).

3.3.1 Тестовые системы

Классическая теория тестирования предполагает, что ответ на вопрос с вариантами ответа может быть представлена в виде случайной бернулевской величины ξ_i [94]. Итоговый результат прохождения тестирования представлен в виде вектора $\xi = (\xi_1, \dots, \xi_n)$. Теория предполагает вопросы независимыми и имеющими равную сложность, поэтому знания учащегося представляет как сумму координат ξ :

$$s \sim N(\theta, \sigma^2), \quad (3.10)$$

где s задает экзаменационный результат обучаемого, параметр θ — истинный уровень знания, σ^2 — задает волатильность измерений. Параметр волатильности считается равным для всех учащихся и подбирается путем максимизации правдоподобия на экспериментальной выборке. Существенным недостатком такой теории является предположение о равной сложности задач для учащегося в контрольной работе. Контрольно-методические материалы, составленные по такой системе, как правило, имеют бинарные ответы и проверяют лишь базовые предметные знания [95].

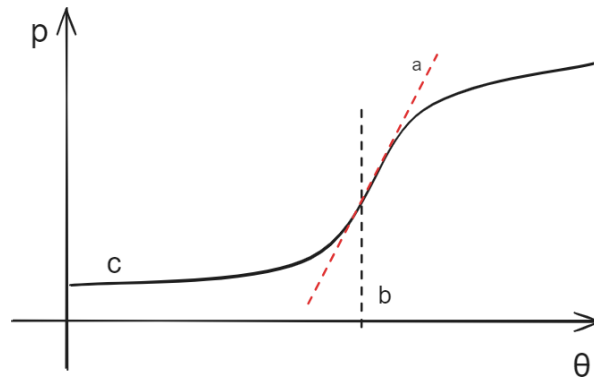


Рис. 3.8: Отклик учащегося на образовательный материал в теории IRT задается смещенной логистической функцией

Для работы с упорядоченным по сложности заданиями институтом Educational Testing Service была создана теория отклика на материал (от *англ.* Item Response Theory)[96]. Наиболее известным результатом теории является 3-х параметрическая логистическая модель 3.8, учитывающая сложность задачи, вероятность угадать правильный ответ и волатильность оценки [97]:

$$p_i(\theta) = c_i + \frac{1 - c_i}{1 + e^{-a_i(\theta - b_i)}}, \quad (3.11)$$

где θ — текущий уровень навыков, b_i — сложность задания, a_i — характерный масштаб, c — вероятность угадать решение. Выбор коэффициентов выполняется на экспериментальных данных путем оптимизации с помощью ЕМ-алгоритма [98]. Таким образом, каждая задача приобретает скалярный уровень сложности, используемый методистами для составления сбалансированных по сложности контрольно-методических работ.

Теория отклика используется при составлении контрольно-методических систем, значительно влияющих на профессиональные возможности экзаменуемого и репутацию образовательных систем,

- международных экзаменов по математике GMAT, английскому языку: IELTS и TOEFL;
- программ оценки образовательных достижений PISA и TIMSS;
- поступлении в американские колледжи SAT.

Также теория отклика активно применяется в современном персональном образовании с использованием информационно-коммуникационных систем (от *англ.* Information and Communication System)[99] [100]. Образовательные платформы по обучению языку Revita [101], математике MathGarden [102] и программированию Stepik используют функцию отклика для адаптивного обучения, подбирая сложность заданий исходя из текущих знаний учащегося. Такой подход, на текущий момент, формирует новое научное направление теории цифрового образования (от *англ.* technology enhanced learning), объединяющей рекомендательные и психометрические подходы для эффективного обучения [100]. Базовой моделью адаптивного алгоритма оценки



Рис. 3.9: Эволюция представлений о знаниях учащегося

знаний является байесов пересчет [103]. Модель учитывает вероятность ошибки и вероятность ошибиться при наличии знания 3.9:

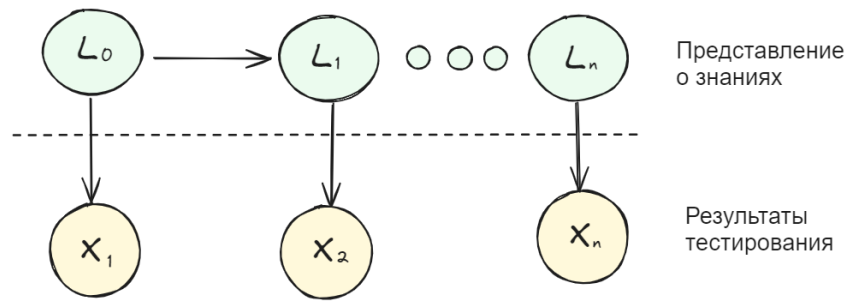
- $P(L_0)$ — начальные знания в предмете;
- $P(S) = P(x = 0|L_t = 1)$ — вероятность просчета при наличии знаний;
- $P(G) = P(x = 1|L_t = 1)$ — вероятность угадать при отсутствии знаний.

Обновление представлений выполняется с учетом ответа согласно:

$$\begin{aligned} P(L_t|obs_t = 1) &= \frac{P(L_t)(1 - P(S))}{P(L_t)(1 - P(S)) + (1 - P(L_t))P(G)} \\ P(L_t|obs_t = 0) &= \frac{P(L_t)P(S)}{P(L_t)P(S) + (1 - P(L_t))(1 - P(G))}. \end{aligned} \quad (3.12)$$

Отметим, что полученный вывод предполагает, что:

- вероятность забыть знание равна нулю $P(L_{t+1} = 0|L_t = 1) = 0$;
- вероятность наличия знаний обновляется по правилу $P(L_{t+1}) = P(L_t|obs_t) + (1 - P(L_t|obs_t))P(T)$.

Рис. 3.10: Матрица исходов модели Байесовской оценки на шаге t

Таким, образом тест можно представить в виде марковской цепи обновления представлений о знаниях учащегося 3.10.

Адаптация для случая IRT позволяет учесть влияние сложности задания [104]:

$$P(Y_{ij} = 1 | \theta_j, a_i, b_i, c_i) = c_i + (1 - c_i) \frac{1}{1 + e^{-a_i(\theta_j - b_i)}} \quad (3.13)$$

Глава 4

Описание работы

В главе описаны шаги адаптации большой языковой модели к задачам образования. Описан формат взаимодействия, способ обучения, алгоритм оценки знаний и задания сложности, необходимые для компенсации текущих недостатков больших языковых моделей. Практическим итогом работы стало веб-приложение, организующее персональное обучение шахматам с поддержкой интеллектуального ассистента. В разработанной системе большая языковая модель отвечает за общение с обучаемым и комментирование игры. Для поддержания интереса к игре алгоритм автоматически адаптируется под уровень игрока таким образом, чтобы доля побед была равна 50%. При необходимости пользователь может сообщить о потребности изменить уровень на более подходящий его предпочтениям. Приложение также имеет многоуровневую систему достижений, поощряющих нестандартную игру и настойчивость в обучении. В главе также приведено описание сбора и подготовки для обучения данных образовательной направленности из открытых источников с использованием современных систем оптического распознавания.

Главным теоретическим результатом работы является адаптация алгоритма Роббинса-Монро под отклик, заданный логистической функцией. В завершающей секции главы описана теорема, задающая в аналитическом виде коэффициенты численной схемы для оптимального спуска к заданному уровню попыток.

4.1 Подготовка данных для обучения

Прогресс в области машинного обучения и разработки интеллектуальных ассистентов ведет к росту потребности в высококачественных корпусах текстов и аннотированных изображений. Обучение на предметном корпусе позволяет улучшать количественные метрические показатели достоверности передаваемых знаний на десятки процентных пунктов [105]. Тем не менее открытые корпуса русского языка [106] [107] [108] почти не содержат образовательной тематики. Представленный в секции корпус является вариантом решения проблемы недостатка данных в направлении образова-

ния. Результаты моделирования предоставлены на открытых ресурсах¹ с указанием источников данных 5.3. Полученный корпус имеет мощность порядка миллиона слов и десятка тысяч изображений. На рисунке 4.1 приведено распределение по источникам данных, включающее открытые цифровые документы, адаптированные с других языков корпусы и естественно-научные журналы.

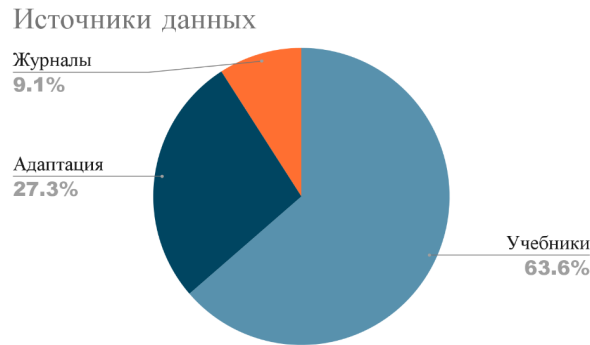


Рис. 4.1: Распределение данных по источникам.

4.1.1 Распознавание текста

Системы оптического распознавания позволяют включать в корпусы текста для обработки цифровые документы с неподготовленным текстовым слоем. Автором работы был разработан открытый программный пакет для языка python — ShuemacherOCR, который позволяет выполнять масштабный анализ русскоязычной естественно-научной методической литературы. Разработка специального пакета связана с особенностью составления печатных изданий. Книги естественнонаучной тематики имеют существенно различающуюся верстку и содержат большое число иллюстраций, графиков и формул. Современные пакеты для выполнения оптического распознавания Nougat [109], Tesseract [110] и LayoutParser [111] не вполне справляются с распознаванием таких документов, поскольку либо не поддерживают русский язык, либо не предназначены для работы с формулами.

В состав пакета входят модули как для обработки отдельных изображений, так и полноценных документов, позволяющие извлекать данные в структурированном виде. Для решения задачи пакет использует нейросетевые алгоритмы. Особенностью пакета является возможность выделять в тексте на русском языке строчные математические формулы. Установка выполняется из открытого реестра пакетов PyPI с помощью менеджера pip или из репозитория².

Опишем процесс подготовки данных для обучения модуля оптического распознавания. Начнем с принципиально описания процесса.

¹<https://github.com/NMashalov/Generative-modeling-appliance-for-creating-educational-tasks> и https://huggingface.co/datasets/NMashalov/task_illustrations_dataset

²<https://github.com/NMashalov/SchumacherOCR>

Параметр	Обучение	Валидация
BLEU	83.2	80.4
Edit distance	0.15	0.17

Определение: **Оптическое распознавание символов** (OCR) представляет собой процесс автоматического преобразования текста, представленного в виде изображения или сканированного документа, в текстовый формат.

В ходе оптического распознавания исходное изображение документа подвергается предварительной обработке. Корректируется ориентация изображения, подбирается оптимальная яркость, удаляются шумы. Следующим этапом является сегментация изображения, включающая разделение исходных изображений на отдельные символы или группы символов. Далее при помощи алгоритмов распознавания, включающих методы машинного обучения и компьютерного зрения, символы на изображении анализируются и сопоставляются с соответствующими символами из набора знаков. На завершающем этапе распознавания символы объединяются в слова, предложения и абзацы, формируя полноценный текстовый документ. Точность и эффективность процедуры зависят от качества изображения, используемых алгоритмов распознавания, а также от языка и структуры текста.

Распознавание текста по изображению выполнялось на основе нейросети архитектуры Nougat [109]. Особенностью данной архитектуры является быстрая адаптация под новые виды данных и работа с изображением целиком, без необходимости промежуточного поиска регионов с текстом.

Обучение сети проводилось на корпусе препринтов статей [112], переведенных на русский язык с помощью интеллектуального ассистента ChatGPT [58]. Выбор был связан с возможностью сохранять оригинальную разметку TeX-документов.

Для валидации результатов был разработан общедоступный корпус, позволяющий измерить качество распознавания ³.

Разметка для обучения проводилось с помощью обращения к сервису компании MathPix. Метрики качества приведены в таблице 1 и сопоставимы с результатами, которые дает оригинальная модель Nougat.

Автор продолжает развитие пакета и его адаптацию для среды распределенных вычислений, использующих акторную модель взаимодействия между вычислителями.

4.1.2 Дополнительные подходы к сбору данных

Одним из наиболее эффективных подходов к сбору данных является обращение к международным сайтам исследователей, содержащих большое число структурированных данных. К сожалению, большинство из них представлены на китайском и английском языках. Для их адаптации был предложен подход, использующий боль-

³https://huggingface.co/datasets/NMashalov/ru_educational_book_datasets

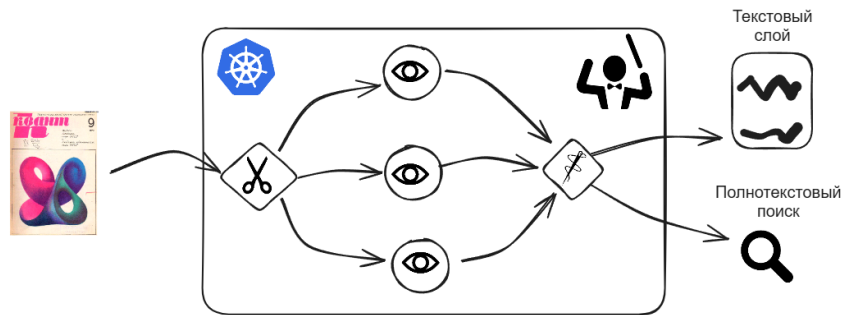


Рис. 4.2: Итоговая разметка выполняется посредством распределенных вычислений

шие языковые модели в качестве средства перевода на русский язык. Для этого был использован открытый языковой ассистент llama3 [3], владеющий более чем ста языками мира и имеющий высокие показатели понимания естественного языка.

Перевод 7500 задач с английского и китайского выполнялся в течении 12 часов. Полученные результаты приведены в виде таблицы в репозитории⁴. Исследование демонстрирует важность выбора оптимальной инструкции для модели. В отсутствие четких требований модель нарушает формат постановки задачи, искажает числа и обозначения в ответах. Таким образом, проведенный машинный перевод требует экспертной адаптации в силу отсутствия явной системны в ошибках. Тем не менее такой подход позволяет существенно сократить время перевода в сравнении с работой со словарем, поскольку суть задачи передается моделью верно.

Цифровые документы также могут быть использованы не только для сбора текста, но и еще для составления аннотированных образовательных иллюстраций. Такие данные особенно ценны для больших языковых моделей, обученных работать с изображениями.

Для выделения иллюстраций и подписей к ним была использована сеть YOLO [113]. Эта архитектура нейронной сети имеет способность эффективно дообучаться на небольших выборках данных, что позволяет в краткие сроки достигать удовлетворительных результатов. Для ситуаций, в которых число аннотаций и число изображений на изображении не совпадало, применялся алгоритм на двудольном графе, направленный на максимизацию числа пар.

Для получения обучающей выборки была проведена разметка части корпуса цифровых учебников. Каждое изображение включает в себя текстовую информацию, а также различные чертежи и формулы, характерные для данной области знаний.

Для расширения корпуса и обеспечения его разнообразия была применена аугментация данных. Применялись повороты, масштабирование, изменение освещения и отражение. Это позволило создать дополнительные вариации входных данных, что способствовало увеличению разнообразия обучающей выборки и повышению устойчивости модели к различным вариациям данных. В реальных условиях это определяет эффективность модели при работе с различными вариантами верстки.

⁴https://huggingface.co/datasets/NMashalov/olympiad_task_translation

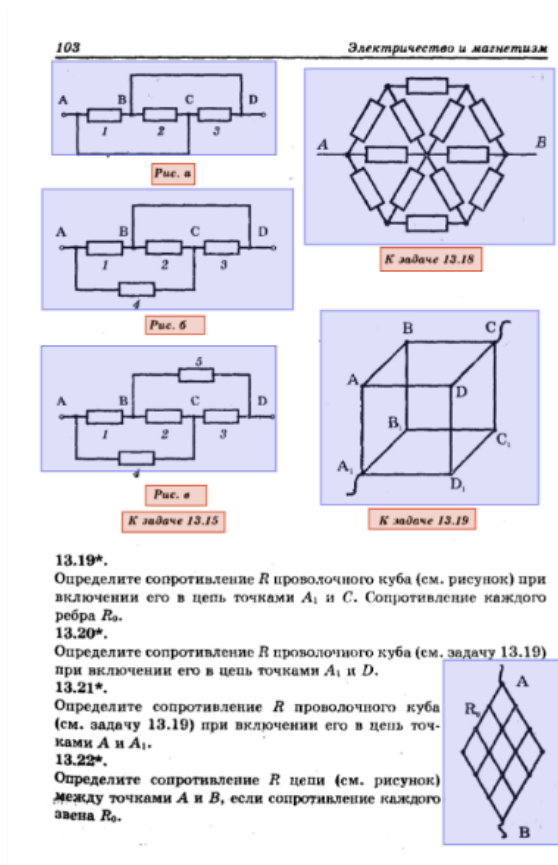


Рис. 4.3: Пример аннотированной иллюстрации из книги Генденштейн, Кирик, Гельфгат: 1001 задача по физике

Метрическая оценка результатов выделения иллюстрации и аннотации:

Параметр	Обучение	Валидация
mAp	78.4	75.4
“изображение-аннотация”	75.2	72.0

4.2 Разработка интерфейса взаимодействия

Современные интеллектуальные ассистенты являются многокомпонентными системами, организующими среду взаимодействия, целевой сценарий диалога, персонализацию и приватность переписки. Конкретные компоненты выбираются исходя из целей и возможностей команды разработчиков. Для исследовательских целей, как правило, используется открытое программное обеспечение, требующее лишь незначительной адаптации под постановку. В этой секции будет описан подход к созданию цифрового ассистента для проведения исследования.

Основой современных ассистентов является большая языковая модель, обученная посредством техник оптимизации выполнять инструкции, описанные естественным языком. Модель прекрасно справляется с задачами коммуникации, придерживается делового этикета и демонстрирует эмпатическое внимание к проблемам пользователя [59][3].

Языковая модель также помогает в решении повседневных и деловых задач, выделяя наиболее важное из академических статей и помогая составлять планы проектов. Последние достижения позволяют задавать вопросы по изображениям, что существенно облегчает выбор продуктов, например одежды или мебели [114]. Тем не менее современные ассистенты не способны к выполнению формальных логических операций: арифметических действий [115], решению абстрактных логических задач [116], соблюдению правил стратегических игр [69].



Рис. 4.4: Навыки современного ассистента ограничены коммуникацией и решением базовых задач обработки текста

В образовании интеллектуальные ассистенты применяются для обучения языку [117] и рисования поясняющих графиков [118]. Примерами коммерческого использования ассистентов в образовании являются компании Merlin Mind и OpenAI Education. Ключевым преимуществом решений является адаптация к государственным общеобразовательным программам, взаимодействие с интерактивной доской и проприетарно подготовленная база знаний, регулярно обновляемая предметными экспертами.

4.2.1 Проект интеллектуального ассистента

Современный интеллектуальный ассистент выполняет множество задач по коммуникации и обработке информации. Основные функции включают в себя способность поддержать беседу на произвольную тему, информирование о происходящих в мире событиях, консультация по научной литературе и создание отрывков текста и изображений по запросу пользователя.

Определение: **Интеллектуальный ассистент** — прикладное программное обеспечение, выполняющее задачи, поставленные пользователем в виде команд на естественном языке.

Для эффективного поддержания функций ассистента используется модульная система, распределяющая задачи на выделенные компоненты системы. Такой подход облегчает совместную работу разработчиков. Модули, как правило, организованы последовательно, образуя цепь преобразований. Системы обработки естественного

языка включают классификаторы, инструменты информационного поиска и подготовленные шаблоны ответов, заполняемые исходя из контекста беседы. В текущей

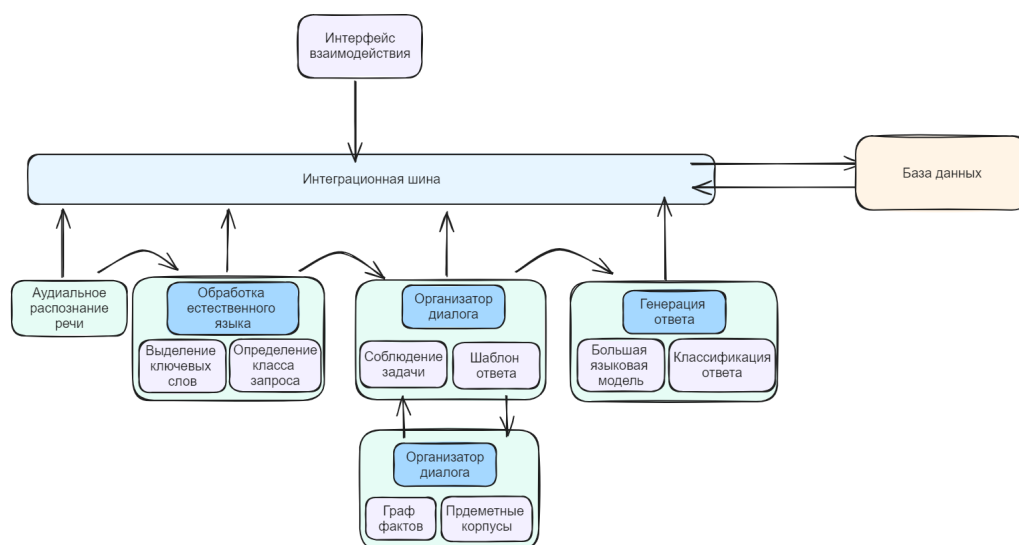


Рис. 4.5: Модульная система позволяет эффективно организовывать командную работу.

практике шаблоны ответов заменяются на большие языковые модели. Такой подход позволяет повысить метрические показатели удовлетворенности диалоговыми системами [61]. Исследователи связывают преимущества с обилием знаний, полученных моделями при обучении. Большие языковые модели эвристически понимают задачи пользователя, при необходимости умеют уточнять его потребности и могут быть адаптированы путем примеров без потребности в обучении [119].

Разработанный в рамках работы ассистент включает в себя три основных компонента:

- модуль исключения ненормативной лексики и неэтичных изображений;
- большая языковая модель;
- модуль принятия решения о сложности задания для пользователя.

Пресечение использования ненормативной лексики было выполнено с помощью морфологического анализатора текста PyMorphu [120]. Для исключения неэтичных по содержанию предложений и изображений был адаптирован бимодальный кодировщик CLIP [57]. Преимущество работы кодировщиком заключается в определении негативного содержания в контексте [121]. Используемые алгоритмы гарантируют исключение бранных слов и выделяют с высоким качеством неприемлемые по смыслу фрагменты текста и иллюстрации (табл. 4.1).

В качестве большой языковой модели была использована архитектура Llama3 [3]. Это наиболее продвинутая открытая модель, способная к общению на более чем ста языках мира. Ассистент вежлив в общении, внимателен к эмоциям собеседника и

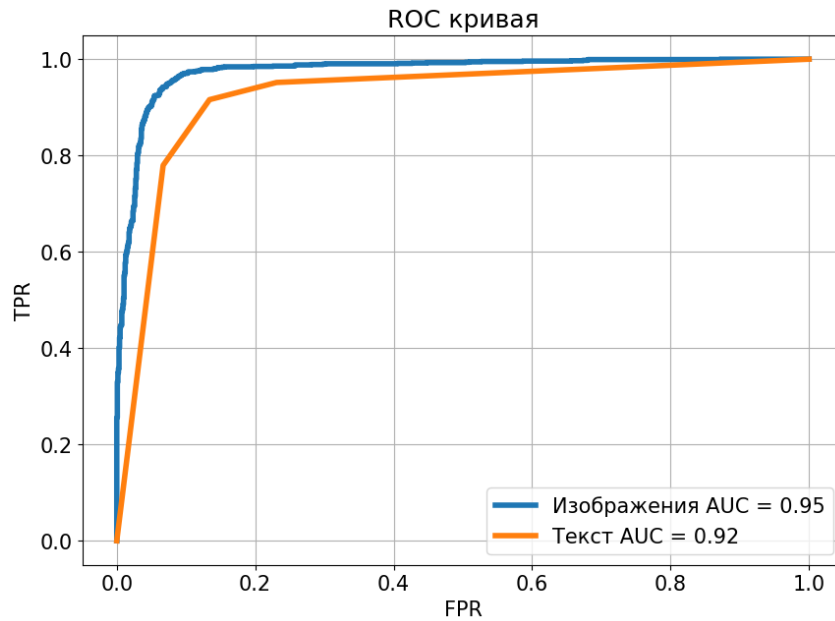


Рис. 4.6: Адаптация модуля CLIP позволила достичь высоких метрических результатов выделения неприемлемых изображений и фрагментов текста

Параметр	Текст	Изображения
ROC AUC, %	92.0	95.0

Таблица 4.1: Метрические показатели

имеет обширные знания о мире. В диалоговой системе ассистент следит за эмоциональным откликом обучающегося на материал, узнает ее причину и в случае фиксации несоразмерной уровню нагрузки сообщает управляющей системе о потребности изменения сложности. Также в задачи ассистента входят рекомендации по обучению и методической литературе для ознакомления.

Для поддержания оптимального уровня сложности используются байесова модели рейтинга, скалярно определяющая сложность задания и уровень знаний учащегося. Принципиально система состоит из определения вероятности решения задачи с заданной сложностью и правила обновления рейтинга согласно результату решения. Преимуществом таких систем является возможность обновления рейтинга учащегося в реальном времени.

Наиболее известной рейтинговой системой является модель Брэдли-Терри [83]:

$$p(i \succ j) = \frac{\theta_i}{\theta_i + \theta_j}, \quad (4.1)$$

где θ_i — текущий уровень знаний учащегося, θ_j — сложность задачи. Строгое правило пересчета требует учета всех экспериментальных данных и может быть вычислительно затратным [122]. Поэтому на практике используется перепараметризация модели

$\theta = \exp(\frac{\nu}{\sigma})$, где σ задает представления о волатильности данных:

$$p(i \succ j) = \frac{1}{1 + \exp(\frac{\nu_i - \nu_j}{\sigma})}. \quad (4.2)$$

Полученная модель соответствует модели Эло [84] с правилом пересчета:

$$\theta_i^{(t+1)} = \theta_i^{(t)} + \sigma, \quad (4.3)$$

где t задает номер пересчета. Таким образом, система стимулирует к решению более сложных задач, требующих освоения новых знаний. Прогресс обучающегося согласно предложенной системе может быть статистически наблюдаем как повышение доли успеха в решении задач.

4.2.2 Подготовка веб-приложения

Интерфейс приложения организует работу с ассистентом. Для реализации доступного в использовании приложения был использован "гештальт-дизайн" [123], задающий правила композиции элементов интерфейса. Также были учтены стандарты WAI-ARIA [124], задающие правила высокого контраста и выраженности контуров, позволяющих выполнять навигацию слабовидящим людям. Функционал приложения ограничен доступен и незрячими людям, использующим специальные приложения для аудио-отображения содержания сайта.

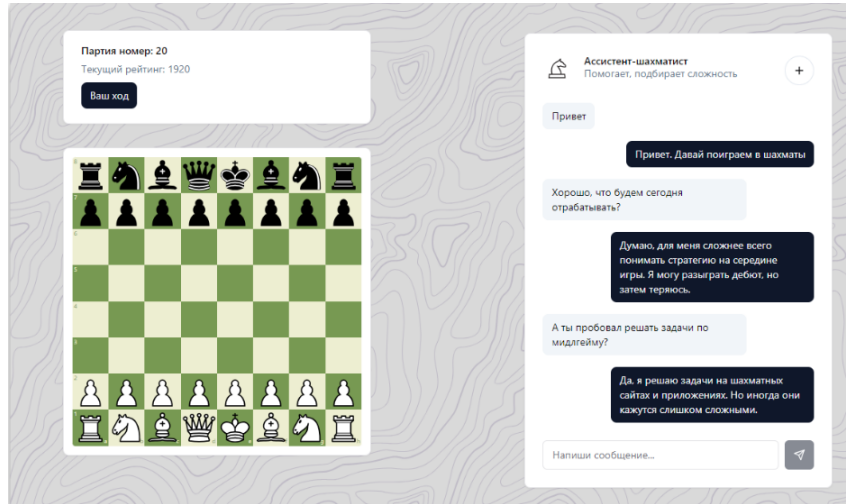


Рис. 4.7: Интерфейс имеет четкие и контрастные элементы взаимодействия. Доска для игры и диалоговое окно размещены совместно

Веб-приложение доступно при подключении через браузер по адресу доменного имени www.mathema-online.xyz. Безопасность соединения обеспечивается криптографической защитой и сертификатами, исключающими возможность подмены доменного имени.

Интерфейс реализован с помощью популярной библиотеки React для языка программирования JavaScript [gackenheim2015introduction]. Такой подход позволяет дескриптивно описывать элементы вебсайта с событийной реакцией на взаимодействие с пользователем. Особенностью подхода является возможность использовать общедоступные профессионально подготовленные схемы интерфейсов в виде css-разметки.

4.2.3 Описание разработанных игр

В секции представлены разработанные игры, использующие современные нейросетевые технологии для персонального обучения. Выбор направления был выполнен с учетом растущих возможностей больших языковых моделей. В качестве предмета обучения были выбраны стратегические игры, развивающие навыки долгосрочного планирования и критического мышления. Также было исследовано применение диффузионных моделей для растрового рисования, позволяющее отработать навыки передачи структуры и природы объектов.

Современные большие языковые модели пока не способны к полноценному ведению игры. В работе [69] проведен анализ силы игры ассистента ChatGPT путем сравнения с шахматной программой Stockfish. Статистические исследования показывают, что текущий уровень игры модели соответствует рейтингу Эло 1600 [84]. Это начальный уровень игрока в шахматы. Исследователи также отмечают неспособность ассистента к строгому исполнению правил игры и наблюдают противоречия в стратегии даже в коротком отрезке партии.

Исходя из текущих возможностей был предложен гибридный подход, заключающийся в совмещении языковой модели с открытой шахматной программой StockFish [125]. В такой постановке интеллектуальный ассистент отвечает на вопросы пользователя по ходу игры и рассказывает о возможных стратегических решениях, исходя из ситуации на доске. Шахматная программа задает уровень оппонента и выдает информацию для обновления рейтинга. Обновление выполняется в соответствии с моделью Эло, в зависимости от уровня игры оппонента по правилу:

$$x_{i+1} = x_i + K(\beta) [X_i - P(X_i = 1)], \quad (4.4)$$

где $P(X_i = 1)$ задает вероятность победы в игре.

Алгоритм базовой игры может быть описан как последовательность шагов:

1. пользователь начинает игру;
2. пользователь по ходу игры консультируется с ассистентом, и при необходимости изменяет сложность;
3. ассистент отслеживает результаты игры и обновляет рейтинг. Пользователь может запрашивать советы.

Для поощрения настойчивости и развития интереса к игре была разработана система достижений, включающая как количественную оценку прогресса, так и выполнение нестандартных задач. Правила получения награды определяются порогом отсечки с помощью алгоритма, описанного в 3.3. Такая система поощряет соревновательный дух.

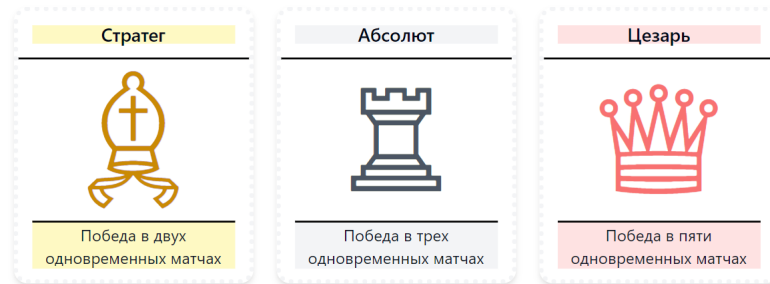


Рис. 4.8: Достижение на примере награды за победу в сеансе одновременной игры

Разработка игры по рисованию была выполнена с использованием открытой диффузионной модели 2.2.2, составляющей рисунок по текстовому запросу. Для простоты выполнения рисунка предложено выполнение мозаики из пикселей, представляющих базовый элемент растровой сетки изображения. Адаптация базовой модели для стилизации рисунка выполняется с помощью открытого низкорангового адаптера. Исходный запрос для модели должен быть сформулирован на английском языке, лаконично описывать объекты на изображении и обстановку. В силу случайности генерации пользователь может подобрать для себя наиболее интересный вариант изображения. Сложность рисунка определяется наличием фона, декораций и сложностью композиции.

Алгоритм выполнения рисунка состоит из трех последовательных этапов:

1. пользователь вводит описание желаемого предмета на русском языке;
2. ассистент переводит запрос на английский язык с учетом его рейтинга и предоставляет пользователю выбор из заданных рисунков;
3. кодировщик сравнивает рисунок с исходным изображением и определяет балл для обновления рейтинга.

Модель дополнительно снабжена фильтром цензуры, позволяющем исключать изображения не соответствующие этике [57]. Уровень сложности регулируется путем изменения композиции и наличия фона.

4.3 Рейтинговая система

Стохастическая аппроксимация — эффективный метод поиска корней уравнения в условиях случайного несмещенного отклика. В работе [126] была предложена адап-

Интерактивное рисование

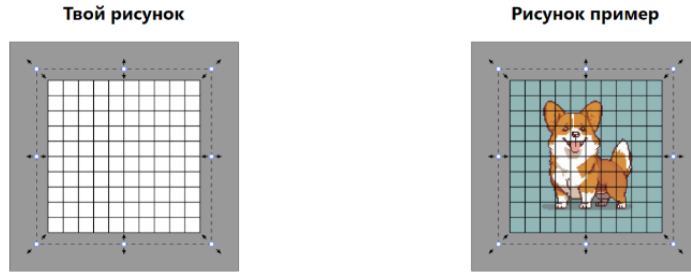


Рис. 4.9: Рисунок выполняется путем сопоставления результата с опорным изображением.

тация методов стохастической аппроксимации к задачам оптимального выбора сложности заданий. Постановка представляет тест как стохастический ряд вида $\{x\}_{t=0}$, каждый элемент которого является случайной бернуллевской величиной с параметром s . Для ввода управляющей переменной задается сложность задачи d , параметризующая, в совокупности с функцией отклика учащегося f , переменную $s_t = f(d)$.

Таким образом, задача алгоритма рассчитать функцию $f(d_{t+1}^t, x_{i=0}^t)$, обеспечивающую оптимальную сходимость $\lim_{t \rightarrow \infty} \rho(s(d_t), s^*) = 1$ согласно с условиями:

- метрика $\rho(x, x') = (x - x')^2$ евклидова;
- предполагается наличие банка W , возвращающего задачу произвольной сложности d ;
- функция отклика $f(d_t)$ ограничена числом M и монотонно убывает.

Авторы выбрали правило обновления сложности согласно правилу:

$$d_{t+1} = \Pi(d_t + \lambda(x(t) - s^*)), \quad (4.5)$$

где функция Π является ограничивающим оператором вида:

$$\Pi_H(d) = \begin{cases} d, & \text{при } 1 < d < 0 \\ 1, & d \geq 1 \\ 0, & \text{при } d \leq 0 \end{cases} \quad (4.6)$$

В секции рассмотрена модификация заданного метода путем явного учета логистического вида отклика учащегося на задание.

4.3.1 Адаптация рейтинговой системы

Принципиальная схема доказательства состоит из трех этапов:

1. используем модифицированный алгоритм Роббинса-Монро;

2. зададим связь между функцией отклика $s(d)$ и параметрами a_n и b_n через условия несмещенности оценки $E(x_n)$ и минимизации дисперсии $\mathbf{D}(x_n) \rightarrow \min$ [127] [128];
3. определим явное выражение a_n и b_n для априорного представления о функции отклика в виде параметрической модели Эло $s(d, \alpha, \beta) = 1 - \frac{1}{1 + \exp(-\beta \cdot (d - \alpha))}$

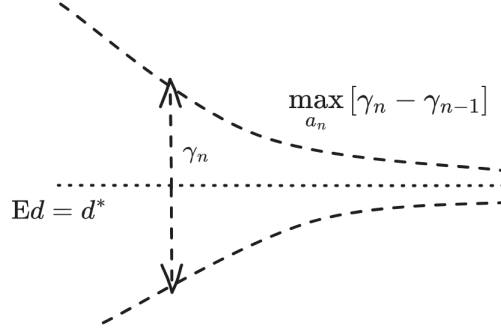


Рис. 4.10: Использование несмещенной оценки с постепенной редукцией дисперсии

Теорема: Адаптированный алгоритм Роббинса-Монро для случая наблюдений, имеющих бернуллево распределение. При условии $\forall n \rightarrow x_n \sim \text{Bern}(s(d_n))$ выполняются следующие утверждения.

1. оптимальная сходимость достигается при $a_n = \frac{\mathbb{E}[d_n s(d_n)]}{\mathbb{E}s(d_n)(1 - \mathbb{E}s(d_n))}$, $b_n = \mathbb{E}s(d_n)$;
2. для функции отклика, представленной параметрической моделью Эло $s(d, \alpha, \beta) = 1 - \frac{1}{1 + \exp(-\beta \cdot (d - \alpha))}$ и в предположении нормальности распределения $d \sim \mathcal{N}(\alpha, \sigma^2)$ получим аппроксимацию рядов a_n и b_n , обеспечивающих оптимальную сходимость как:

- $a_n = \frac{1}{b_n(1-b_n)} \frac{1}{4\beta\sqrt{1+\frac{\pi\gamma_n^2}{8}}} \exp\left(\frac{-\pi(\alpha-d^*)^2}{16\beta^2(1+\frac{\pi\gamma_n^2}{8})}\right)$
- $b_n = \frac{1}{2} + \frac{1}{2} \text{erf}\left(\frac{\sqrt{\pi}(\alpha-d^*)}{4\sqrt{1+\frac{\pi\gamma_n^2}{8}}}\right).$

Доказательство: Анализ сходимости метода приведен в приложении к работе ???. Используем рекуррентное представление для поиска оптимальных параметров a_n, b_n :

$$d_{n+1} = d_n - a_n(x_n - b_n). \quad (4.7)$$

Рассчитаем матожидание $\mathbb{E}_{d_n} p(d_n)$:

$$\mathbb{E}d_{n+1} = \mathbb{E}d_n - a_n(\mathbb{E}x_n - b_n). \quad (4.8)$$

Запишем условия $\mathbb{E}d_{n+1} = d^*$ с учетом $\mathbb{E}d_n = \dots \mathbb{E}d_1 = d^*$:

$$a_n(\mathbb{E}s(d_n) - b_n) = 0. \quad (4.9)$$

Следовательно:

$$b_n = \mathbb{E}s(d_n). \quad (4.10)$$

Коэффициент a_n найдем из минимизации дисперсии $\mathbf{D}d_{n+1}$. Запишем $\mathbf{D}_{x_n \sim \text{Bern}(s(d_n))}$ для выражения 4.7:

$$\mathbf{D}d_{n+1} = \mathbf{D}d_n + a_n^2 \mathbf{D}(x_n - b_n) - 2a_n \mathbb{E}[(x_n - b_n)d_n]. \quad (4.11)$$

Поскольку $x_n \sim \text{Bern}(s(d_n))$, то

$$\mathbf{D}(x_n - b_n) = \mathbb{E}s(d_n)\mathbb{E}(1 - s(d_n)) = \mathbb{E}s(d_n)(1 - \mathbb{E}s(d_n)). \quad (4.12)$$

С учетом $b_n = \mathbb{E}s_d$ и $\mathbb{E}d_n = d^*$:

$$\mathbb{E}[(x_n - b_n)d_n] = \mathbb{E}[x_n d_n] - b_n \mathbb{E}d_n = \mathbb{E}[s(d_n)d_n] - \mathbb{E}s(d)d^*. \quad (4.13)$$

Тогда:

$$\mathbf{D}d_{n+1} = \mathbf{D}d_n + a_n^2 \mathbb{E}s(d_n)\mathbb{E}(1 - s(d_n)) - 2a_n \mathbb{E}[s(d_n)(d_n - d^*)]. \quad (4.14)$$

Из условия $\frac{\partial \mathbf{D}d_{n+1}}{\partial a_n} = 0$ получаем:

$$a_n = \frac{\mathbb{E}[(d_n - d^*)s(d_n)]}{\mathbb{E}s(d_n)(1 - \mathbb{E}s(d_n))}. \quad (4.15)$$

Тогда связь между дисперсиями на каждом шаге запишется как:

$$\mathbf{D}d_{n+1} = \mathbf{D}d_n - \frac{(\mathbb{E}[(d_n - d^*)s(d_n)])^2}{\mathbb{E}s(d_n)(1 - \mathbb{E}s(d_n))}. \quad (4.16)$$

Определим оптимальные коэффициенты a_n и b_n для случая отклика согласно модели Эло: $s(d_n) = \sigma(d, \alpha, \beta) = 1 - \frac{1}{1 + \exp(-\frac{(d_n - \alpha)}{\beta})}$. Исходя из предположения d^* :

$$d^* = \alpha - \beta \log \left(\frac{s^*}{1 - s^*} \right). \quad (4.17)$$

Согласно условию d распределен нормально $\sim \mathcal{N}(\alpha_n, \gamma_n)$. $\alpha_n = d^*$, исходя из несмещенности оценки d_n . γ_n рекуррентно связан с значениям предыдущих операций согласно 4.15:

$$\gamma_n = \gamma_{n-1} - \frac{(\mathbb{E}[(d_n - d^*)s(d_n)])^2}{\mathbb{E}s(d_n)(1 - \mathbb{E}s(d_n))}. \quad (4.18)$$

Найдем b_n из 4.10:

$$b_n = \int_{-\infty}^{\infty} s(x) \mathcal{N}_{d^*, \gamma_n}(x) dx = \int_{-\infty}^{\infty} \sigma(x, \alpha, \beta) \mathcal{N}_{d^*, \gamma_n} dx. \quad (4.19)$$

Для этого используем аппроксимацию логнормального интеграла через функцию

ошибки $\text{erf}(x) = \int_{-\infty}^x \exp(-t^2) dt$:

$$\sigma(x, \alpha, \beta) \approx \frac{1}{2} + \frac{1}{2} \text{erf} \left(\frac{(x - \alpha)\sqrt{\pi}}{4\beta} \right). \quad (4.20)$$

Свертка $\text{erf}(x)$ с плотностью вероятности гауссового распределения $\mathcal{N}(d^*, \gamma_n)$ является табличным интегралом [129]:

$$b_n \approx \int_{-\infty}^{\infty} \left[\frac{1}{2} + \frac{1}{2} \text{erf} \left(\frac{\sqrt{\pi}}{4\beta} (x - \alpha) \right) \right] \mathcal{N}_{d^*, \gamma_n}(x) dx = \frac{1}{2} + \frac{1}{2} \text{erf} \left(\frac{\sqrt{\pi}(\alpha - d^*)}{4\beta\sqrt{1 + \frac{\pi\gamma_n^2}{8}}} \right). \quad (4.21)$$

Найдем a_n из 4.15:

$$\begin{aligned} a_n &= \frac{1}{b_n(1 - b_n)} \int_{-\infty}^{\infty} d(\sigma(x\alpha, \beta) - d^*) \mathcal{N}_{d^*, \gamma_n}(x) dx \\ &\approx \int_{-\infty}^{\infty} x \left[\frac{1}{2} - d^* + \frac{1}{2} \text{erf} \left(\frac{(x - \alpha)\sqrt{\pi}}{4\beta} \right) \right] \mathcal{N}_{d^*, \gamma_n}(x) dx \\ &= \frac{d^*}{2\gamma_n^2} \text{erf} \left(\frac{\sqrt{\pi}(\alpha - d^*)}{4\sqrt{1 + \frac{\pi\gamma_n^2}{8}}} \right) + \frac{1}{4\gamma_n^2} \frac{1}{b_n(1 - b_n)} \int_{-\infty}^{\infty} x \text{erf} \left(\frac{(x - \alpha)\sqrt{\pi}}{4\beta} \right) \exp \left(\frac{(x - d^*)^2}{2\gamma_n^2 \frac{1}{b_n(1 - b_n)}} \right) dx \\ &= \frac{d^*}{2\gamma_n^2} \text{erf} \left(\frac{\sqrt{\pi}(\alpha - d^*)}{4\sqrt{1 + \frac{\pi\gamma_n^2}{8}}} \right) + \frac{1}{4\gamma_n^2} \frac{1}{b_n(1 - b_n)} I(\alpha, \beta, d^*, \gamma_n). \end{aligned} \quad (4.22)$$

Заметим, что полученный интеграл $I(\alpha, \beta, d^*, \gamma_n)$ можно связать с табличным $T(\alpha, \beta, d^*, \gamma_n) = \int \text{erf} \left(\frac{\sqrt{\pi}}{4\beta} (x - \alpha) \right) \exp \left(\frac{(x - d^*)^2}{2\gamma_n^2} \right) dx$ через дифференцирование по параметру d^* :

$$T'_{d^*} = -\frac{1}{2\gamma_n^2} I(\alpha, \beta, d^*, \gamma_n) + \frac{d^*}{2\gamma_n^2} \text{erf} \left(\frac{\sqrt{\pi}(\alpha - d^*)}{4\beta\sqrt{1 + \frac{\pi\gamma_n^2}{8}}} \right). \quad (4.23)$$

С другой стороны:

$$\begin{aligned} T'_{d^*} &= \frac{1}{2} \text{erf} \left(\frac{\sqrt{\pi}(\alpha - d^*)}{4\beta\sqrt{1 + \frac{\pi\gamma_n^2}{8}}} \right)'_{d^*} = -\frac{2}{\sqrt{\pi}\beta} \frac{\sqrt{\pi}}{4\sqrt{1 + \frac{\pi\gamma_n^2}{8}}} \frac{1}{2} \exp \left(\frac{-\pi(\alpha - d^*)^2}{16\beta^2(1 + \frac{\pi\gamma_n^2}{8})} \right) = \\ &= -\frac{1}{4\beta\sqrt{1 + \frac{\pi\gamma_n^2}{8}}} \exp \left(\frac{-\pi(\alpha - d^*)^2}{16\beta^2(1 + \frac{\pi\gamma_n^2}{8})} \right). \end{aligned} \quad (4.24)$$

Приравнявая 4.23 и 4.24 получаем:

$$I = \frac{1}{4\sqrt{1 + \frac{\pi\gamma_n^2}{8}}} \exp \left(\frac{-\pi(\alpha - d^*)^2}{16\beta^2(1 + \frac{\pi\gamma_n^2}{8})} \right) - \frac{d^*}{2\gamma_n^2} \text{erf} \left(\frac{\sqrt{\pi}(\alpha - d^*)}{4\beta\sqrt{1 + \frac{\pi\gamma_n^2}{8}}} \right). \quad (4.25)$$

Подставляем I в 4.22 и получаем аппроксимацию a_n :

$$a_n \approx \frac{1}{b_n(1-b_n)} \frac{1}{4\beta\sqrt{1+\frac{\pi\gamma_n^2}{8}}} \exp\left(\frac{-\pi(\alpha-d^*)^2}{16\beta^2(1+\frac{\pi\gamma_n^2}{8})}\right). \quad (4.26)$$

Также получим γ_n из 4.25 и 4.18:

$$\gamma_n = \gamma_{n-1} - \frac{\gamma_{n-1}^2}{1+\gamma_{n-1}} \frac{1}{b_n(1-b_n)} \exp\left(\frac{-\pi(\alpha-d^*)^2}{16\beta^2(1+\frac{\pi\gamma_n^2}{8})}\right)^2. \quad (4.27)$$

■

Явно приведем основные выражения для приближения к заданному значению s^* :

$$d^* = \alpha - \beta \log\left(\frac{s^*}{1-s^*}\right). \quad (4.28)$$

Коэффициенты a_n , b_n , γ_n рассчитываются рекурсивно:

$$\begin{aligned} \gamma_n &= \gamma_{n-1} - \frac{\gamma_{n-1}^2}{1+\gamma_{n-1}} \frac{1}{b_n(1-b_n)} \exp\left(\frac{-\pi(\alpha-d^*)^2}{16\beta^2(1+\frac{\pi\gamma_n^2}{8})}\right)^2 \\ a_n &= \frac{1}{b_n(1-b_n)} \frac{1}{4\beta\sqrt{1+\frac{\pi\gamma_n^2}{8}}} \exp\left(\frac{-\pi(\alpha-d^*)^2}{16\beta^2(1+\frac{\pi\gamma_n^2}{8})}\right) \\ b_n &= \frac{1}{2} + \frac{1}{2} \operatorname{erf}\left(\frac{\sqrt{\pi}(\alpha-d^*)}{4\beta\sqrt{1+\frac{\pi\gamma_n^2}{8}}}\right). \end{aligned} \quad (4.29)$$

Отметим, что процедуру пересчета коэффициентов можно выполнить однократно при инициализации, тем самым ускорив исполнение программы.

4.3.2 Численные эксперименты

Предложенный алгоритм тестируется с помощью численного моделирования для функции логистической регрессии и сравнивается с классическими подходами. Исходный код на языке Python доступен в открытом репозитории диссертации⁵.

Основным критерием для анализа является соотношение изменения параметра сложности задачи Δd к параметру роста β . Таким образом, были исследованы две ключевые краевые постановки:

1. малые изменения $\Delta d/\beta \approx 0$;
2. значительные изменения $\Delta d/\beta \approx 1$.

В каждом эксперименте сравнивалась эффективность предложенного метода с эффективностью классического алгоритма Роббинса-Монро и его аналога с фиксиро-

⁵<https://github.com/NMashalov/EducationGenerativeModelApplication>

Алгоритм	Число шагов
Постоянный $\lambda_n = 0.01$	400 ± 20
Алгоритм Р-М $\lambda_n = 0.1$	Не сошелся
Алгоритм Р-М $\lambda_n = 0.5$	250 ± 30
Адаптированный алгоритм Р-М	200 ± 35

Таблица 4.2: Сравнение числа шагов сходимости в постановке $\Delta d/\beta \approx 1$

ванным коэффициентом λ [126]. Дополнительно изучено влияние модификации по методу Поляка скользящим средним.

Гладкие графики траекторий получены путем визуализации среднего и перцентилей распределения. Для их численного расчета используется метод бутстрэп [130]. Эксперимент проводился $B \gg 1$ раз, после чего путем расчета распределения определялась статистика.

В разделе приведены основные графики. Полный набор параметров доступен в приложении к работе. Отметим, что при анализе сходимости число шагов N определяется аналогично правилу предела. N считается шагом достижения сходимости, если все последующие точки не выходят за границу ϵ . Значение ϵ выбирается индивидуально из соображений статистической значимости результата.

Случай $\Delta d/\beta \approx 1$

Эксперимент проводился для $s(d) = \frac{1}{1+\exp(-5(d-0.6))}$ с начальной сложностью $d_0 = 0.2$ и целевым параметром $s^* = 0.4$

Отметим также чувствительность алгоритма Роббинса-Монро к параметру шага 4.11:

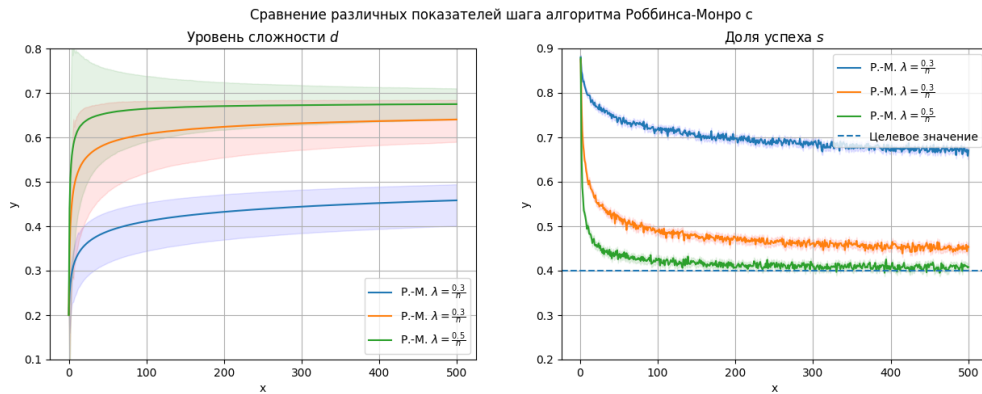
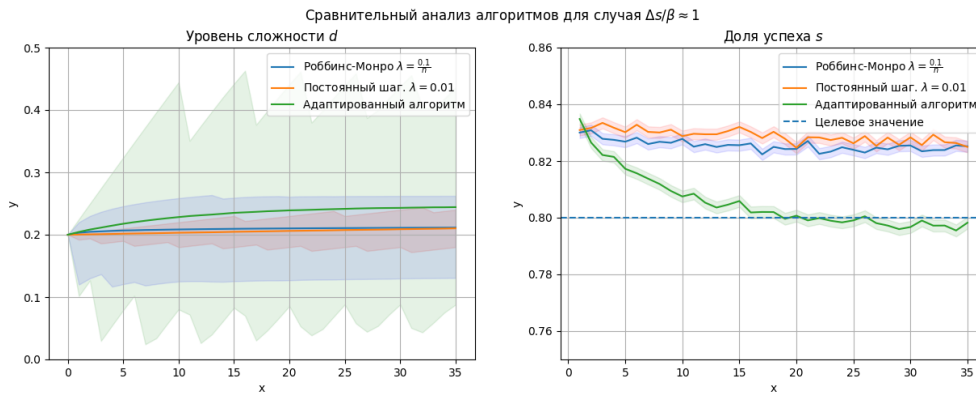


Рис. 4.11: Чувствительность классического алгоритма Роббинса-Монро к параметру λ

Название алгоритма	Число шагов
Постоянный $\lambda_n = 0.01$	400 ± 20
Алгоритм Р-М $\lambda_n = 0.1$	Не сошелся
Алгоритм Р-М $\lambda_n = 0.5$	250 ± 30
Адаптированный алгоритм Р-М	200 ± 35

Таблица 4.3: Сравнение числа шагов сходимости в постановке $\Delta d/\beta \approx 0$.**Случай $\Delta d/\beta \approx 0$**

Эксперимент проводился для $s(d) = \frac{1}{4(d-0.6)}$ с начальной сложностью $d_0 = 0.2$ и целевым параметром $s = 0.8$. Число раундов было выбрано минимальным для⁶

Рис. 4.12: Предложенный алгоритм имеет высокую скорость реакции d **Случай значительного отличия априорных представлений о наклоне кривой от действительного**

Рассмотрен случай, в котором априорное значение β значительно отличается от действительного значения β^* .

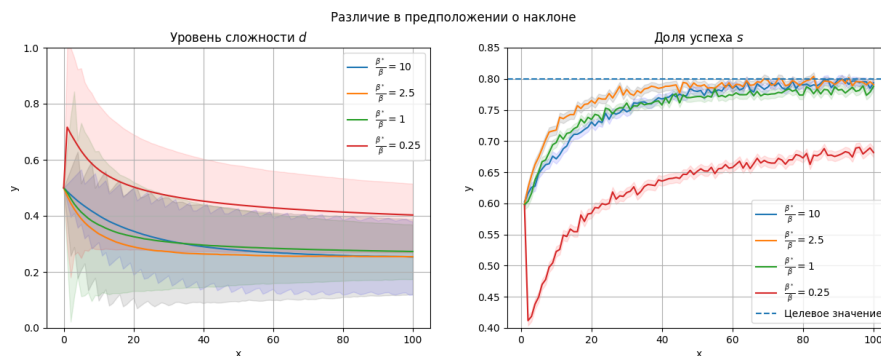


Рис. 4.13: Значительные различия в предположениях о наклоне логистической регрессии приводят к снижению эффективности алгоритма

⁶Для наглядности в таблице классический алгоритм Роббинса-Монро представлен аббревиатурой "Р-М" с указанием параметра шага.

Название алгоритма	Число шагов
Адаптированный алгоритм Р-М $\frac{\beta^*}{\beta} = 10$	400 ± 20
Адаптированный алгоритм Р-М $\frac{\beta^*}{\beta} = 2.5$	Не сошелся
Адаптированный алгоритм Р-М $\frac{\beta^*}{\beta} = 1$	250 ± 30
Адаптированный алгоритм Р-М $\frac{\beta^*}{\beta} = 0.25$	200 ± 35

Таблица 4.4: Сравнение числа шагов сходимости в постановке различающихся априорных представлений о наклоне кривой от действительных

Модификация скользящим средним

Численно исследуем применимость метода скользящего среднего к предложенному алгоритму и алгоритму с постоянным шагом.

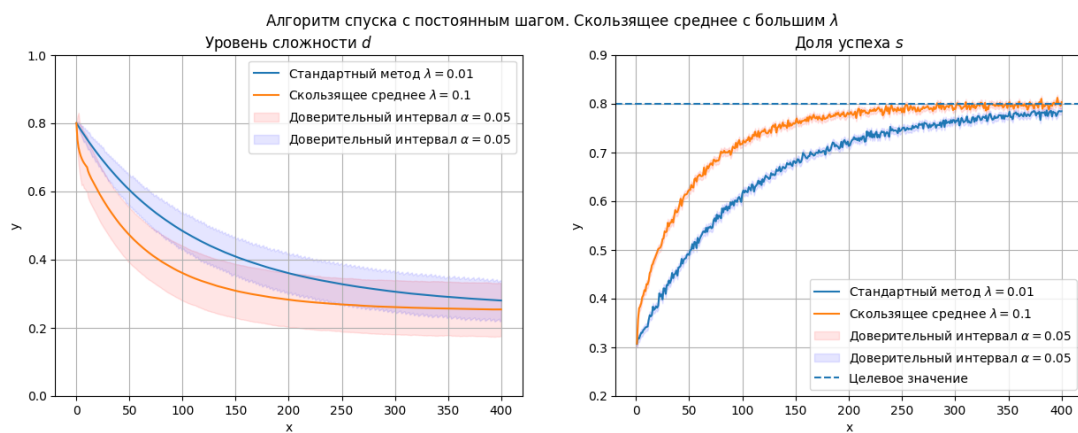


Рис. 4.14: Метод скользящего среднего позволяет использовать больший шаг, не теряя устойчивость метода

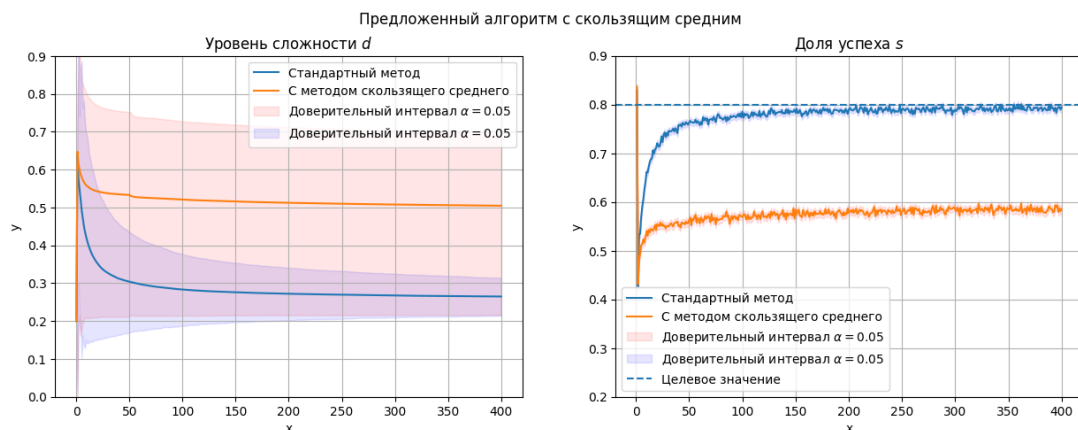


Рис. 4.15: Метод скользящего среднего неприменим к предложенному алгоритму

Расходимость предложенного метода связана с нарушениями условий на инфинитезимальность шага 1.46.

Таким образом, метод скользящего среднего:

1. позволяет использовать увеличенный шаг для обеспечения большей скорости сходимости;

Название алгоритма	Число шагов
Алгоритм Р-М со скользящим средним $\lambda = 0.01$	Не сошелся
Алгоритм Р-М со скользящим средним $\lambda = 0.1$	250 ± 40
Адаптированный алгоритм Р-М со скользящим средним	Не сошелся

Таблица 4.5: Сравнение числа шагов с применением метода скользящего среднего

2. не применим к предложенному алгоритму.

Полный набор задач моделирования приведен в приложении к работе 5.4.

Глава 5

Заключение

5.1 Итоги работы

Поставленная в работе цель по адаптации большой языковой модели к задачам разработки интеллектуального ассистента была успешно выполнена. Разработанный интеллектуальный ассистент сопровождает процесс обучения игре в шахматы и рисованию, задает оптимальный уровень сложности и поддерживает тематический разговор. Полученные в ходе разработки алгоритма адаптивной сложности результаты имеют научную ценность, заключающаяся в повышении скорости спуска к заданному решению. Подтверждение полученного результата выполнено путем проведения численного эксперимента с особым вниманием к крайним случаям.

Для выполнения поставленных задачи были исследованы подходы к созданию и обучению больших языковых моделей, представляющих выделенный класс порождающих моделей. Значительный объем обучающей выборки позволяет таким моделям вырабатывать эвристические правила обработки естественного языка, позволившие расширить ряд задач, выполняемых интеллектуальными ассистентами. Исследователи демонстрируют выдающиеся результаты больших языковых моделей по выполнению инструкций, заданных на естественном языке, включающих задачи перевода, пересказа и информационного поиска. Адаптация моделей для выполнения задач в предметных дисциплинах выполняется путем создания корпусов текстов, включающих примеры использования профессиональных знаний в общении специалистов и в описание решения практических задач. Обучение выполняется экспертом, системно исправляющим ошибки в ответах модели, таким образом совершенствуя ее знания.

Для демонстрации практического применения было разработано интерактивное приложение с учетом текущих возможностей больших языковых моделей. В качестве исследуемой модели была использована архитектура Llama3, к текущему моменту имеющая наилучшие метрические показатели. Для адаптации модели в области естественных наук был собран корпус документов образовательной тематики мощностью порядка миллион слов и десятка тысяч изображений. Также были разработаны модули очистки речи от бранных слов и исключения изображений неприемлемых по

содержанию. В качестве предмета персонального обучения были выбраны шахматы и рисование из соображений развития структурного и стратегического мышления. Заданные предметы имеют ясную проблематику и интерпретируемый уровень сложности.

Аналитический обзор литературы диалоговых систем показал, что практическое применение больших моделей требует создания дополнительных модулей, выполняющих задачи планирования и оптимизации целевых показателей. Поэтому для адаптации большой языковой модели был разработана модификация алгоритма Роббинса-Монро для задания оптимальной сложности задач. Численная схема была получена путем минимизации дисперсии на каждом шаге спуска с явным учетом вида отклика, заданного логистической функцией. Приближения гауссовым интегралом позволило получить аналитический вид коэффициентов схемы. Алгоритм имеет ряд весомых преимуществ для эксперта, заключающихся в ускоренном спуске и возможности выбора параметра шага спуска исходя из априорных представлений о знаниях учащегося. Валидация эксперимента в крайних постановках подтверждает эффективность алгоритма в сравнение с классическими подходами.

Проведенное исследование имеет практическое применение в оптимизации работы банковской коммуникаций с клиентами. Интеллектуальные ассистенты высвобождают операторов поддержки и берут на себя диалог по наиболее часто возникающим вопросам. Вариант разработанного алгоритма используется в системе телефонного информирования клиентов, и позволяет сформировать индивидуальный график контактов с учетом требований федерального закона. В качестве переменных используются длительность и время коммуникации. В совокупности с использованием большой языковой модели коммуникация с клиентами может строиться менее формально и в стиле, учитывающем реакцию клиента. Исследования позволили сократить время обработки контакта оператором и повысить удовлетворенность клиентов.

5.2 Дальнейшая работа

Наиболее перспективным направлением развития исследованных в работе тем является применение адаптивных алгоритмов к задачам группового обучения. Разработанный в работе алгоритм адаптивного подбора сложности пока применим только в постановках индивидуального обучения. Для коллективного обучения такая модель слишком груба, поскольку не учитывает важные для образовательного коллектива процессы соревнования и кооперации учащихся. Для аналитического описания группового обучения необходимо изучить три основных направления:

1. определение правил объединения в группы;
2. подбор сложности задания для группы, учитывающий модель оценки эффективности совместной работы;

3. оценка распределения нагрузки в случае группового задания.

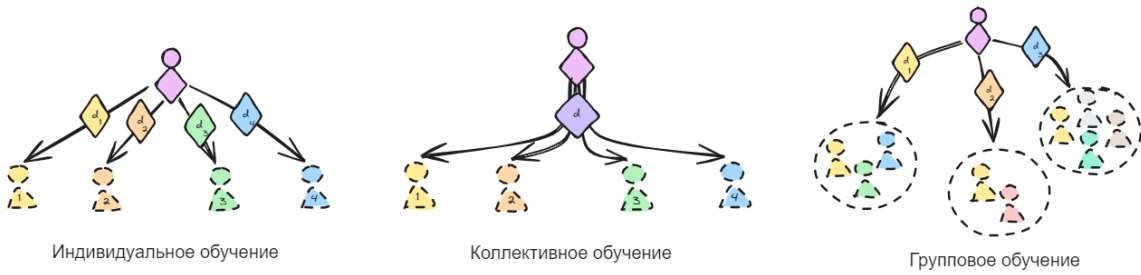


Рис. 5.1: Групповое обучение позволяет задать баланс между объемом проверки и специфичностью задания

При оценке экономической состоятельности корпоративного кодекса и культуры венчурные фонды используют экономические модели организации [131]. Теория охватывает широкий спектр проблем организации совместного труда. В частности, изучаются проблемы определения набора компетенций для членов команды [132], различия в целях между индивидом и коллективом [133] тунеядства (от *англ.* free-riders) и раскрытия корпоративной тайны [134]. Разработанные модели также могут быть эффективно совмещены с алгоритмами распределения нагрузки [135] и дисконтирования вклада в будущем [75].

Материал имеет потенциал для приложения к организации групповых проектов в образовании, поскольку аналитически задает выводы о распределении нагрузки в коллективе. Моделирование выполняется путем введения совместной функции труда $Q(a_1, \dots, a_n)$, где a_i соответствует вкладу i -ого участника. Исследователи вводят разумные предположения о монотонности $\frac{\partial Q}{\partial a_i} > 0$ и порядке эффективности совместного вклада $\frac{\partial^2 Q}{\partial a_i^2} < 0$ и $\frac{\partial^2 Q}{\partial a_i \partial a_j} \geq 0$. Таким образом, труд каждого участника дает вклад в результат, но стратегическое объединение имеет больший эффект.

Изучение оптимального вклада каждого участника выполняется путем оптимизации целевой функции. Существуют различные подходы к её, но для задач образования наиболее актуален баланс между усердиями учащихся и величиной приобретенных знаний $\sum_i \omega_i - g_i(a_i)$, где ω_i соответствует рейтинговому баллу приобретенных знаний, а $g_i(a_i)$ — задает монотонный отклик обучающегося i на величину затраченного времени [136]. Связь между трудом и баллами задается организатором учебного процесса и, как правило, представляется как $\sum_i \omega_i = Q(a_1, \dots, a_n)$. Исследования позволяют получить выводы об оптимальных нагрузках для учащихся в случае логистического отклика. Таким образом, станет возможно аналитическое задание оптимального уровня сложности, ведущего к оптимуму приобретенных знаний.

5.2.1 Благодарности

Автор благодарит кафедру инновационной педагогики Московского физико-технического института за помощь в подготовке работы. Преподавание и академическое взаимо-

действие на кафедре определи направления исследований наиболее актуальные для современного образования. Я благодарю научного консультанта моей работы Дмитрия Евгеньевича Щербакова за рекомендации по академическим материалам в течение подготовки диссертации и введение в практику организации образовательной деятельности, развитой выдающимися педагогами города Москвы. Я признателен Ивану Валерьевича Яценко за внимательное и эффективное руководство научной работой. Курс лекции Ивана Валерьевича по педагогике стал вдохновением к освоению психологического подхода к описанию педагогической деятельности. Также я признателен за поддержку в организации научной работы научному секретарю кафедры Наталье Николаевне Парамоновой.

Список литературы

- [1] Norbert Annuš. «Education in the Age of Artificial Intelligence». В: *TEM Journal* 13.1 (2024), с. 404—413.
- [2] unesco. «Beijing consensus on artificial intelligence and education». В: *International Conference on Artificial Intelligence and Education, Planning Education in the AI Era: Lead the Leap*. 2019.
- [3] Hugo Touvron и др. «Llama: Open and efficient foundation language models». В: *arXiv preprint arXiv:2302.13971* (2023).
- [4] Jean Piaget, Margaret Cook и др. *The origins of intelligence in children*. Т. 8. 5. International Universities Press New York, 1952.
- [5] Лев Семенович Выготский. *Мышление и речь*. Directmedia, 2014.
- [6] Jenova Chen. «Flow in games (and everything else)». В: *Communications of the ACM* 50.4 (2007), с. 31—34.
- [7] Leonid V Kantorovich. «Mathematical methods of organizing and planning production». В: *Management science* 6.4 (1960), с. 366—422.
- [8] Червоненкис АЯ Вапник ВН. «Теория распознавания образов. Статистические проблемы обучения». В: (1974).
- [9] Yu Nesterov. «Universal gradient methods for convex optimization problems». В: *Mathematical Programming* 152.1 (2015), с. 381—404.
- [10] Boris T Polyak. «New stochastic approximation type procedures». В: *Automat. i Telemekh* 7.98-107 (1990), с. 2.
- [11] John Duchi, Elad Hazan, Yoram Singer. «Adaptive subgradient methods for online learning and stochastic optimization.» В: *Journal of machine learning research* 12.7 (2011).
- [12] Alex Krizhevsky, Ilya Sutskever, Geoffrey E Hinton. «Imagenet classification with deep convolutional neural networks». В: *Advances in neural information processing systems* 25 (2012).
- [13] Diederik P Kingma, Jimmy Ba. «Adam: A method for stochastic optimization». В: *arXiv preprint arXiv:1412.6980* (2014).

- [14] Sashank J Reddi, Satyen Kale, Sanjiv Kumar. «On the convergence of adam and beyond». В: *arXiv preprint arXiv:1904.09237* (2019).
- [15] Peter Richtárik, Martin Takáč. «Parallel coordinate descent methods for big data optimization». В: *Mathematical Programming* 156 (2016), с. 433—484.
- [16] Sebastian U Stich. «Unified optimal analysis of the (stochastic) gradient method». В: *arXiv preprint arXiv:1907.04232* (2019).
- [17] Leslie Lamport. «Time, clocks, and the ordering of events in a distributed system». В: *Concurrency: the Works of Leslie Lamport*. 2019, с. 179—196.
- [18] Marshall Pease, Robert Shostak, Leslie Lamport. «Reaching agreement in the presence of faults». В: *Journal of the ACM (JACM)* 27.2 (1980), с. 228—234.
- [19] Jerome Sacks. «Asymptotic distribution of stochastic approximation procedures». В: *The Annals of Mathematical Statistics* 29.2 (1958), с. 373—405.
- [20] Julius R Blum. «Approximation methods which converge with probability one». В: *The Annals of Mathematical Statistics* (1954), с. 382—386.
- [21] Tze Leung Lai, Herbert Robbins. «Adaptive design and stochastic approximation». В: *The annals of Statistics* (1979), с. 1196—1221.
- [22] Siwei Liu, Ke Ma, Stephan M Goetz. «A Robbins–Monro Sequence That Can Exploit Prior Information For Faster Convergence». В: *arXiv preprint arXiv:2401.03206* (2024).
- [23] Татьяна Павловна Красулина. «Об односторонней сходимости процесса Роббинса—Монро при малых шагах». В: *Вестник Санкт-Петербургского университета. Математика. Механика. Астрономия* 4 (2007), с. 67—72.
- [24] V Roshan Joseph. «Efficient Robbins–Monro procedure for binary data». В: *Biometrika* 91.2 (2004), с. 461—470.
- [25] Cui Xiong, Jin Xu. «Efficient Robbins–Monro procedure for multivariate binary data». В: *Statistical Theory and Related Fields* 2.2 (2018), с. 172—180.
- [26] Cédric Villani и др. *Optimal transport: old and new*. Т. 338. Springer, 2009.
- [27] Richard Jordan, David Kinderlehrer, Felix Otto. «The variational formulation of the Fokker–Planck equation». В: *SIAM journal on mathematical analysis* 29.1 (1998), с. 1—17.
- [28] Abien Fred Agarap. «Deep learning using rectified linear units (relu)». В: *arXiv preprint arXiv:1803.08375* (2018).
- [29] Dan Hendrycks, Kevin Gimpel. «Gaussian error linear units (gelus)». В: *arXiv preprint arXiv:1606.08415* (2016).
- [30] Adam Paszke и др. «Automatic differentiation in pytorch». В: (2017).

- [31] Atilim Gunes Baydin и др. «Automatic differentiation in machine learning: a survey». В: *Journal of machine learning research* 18.153 (2018), с. 1—43.
- [32] David E Rumelhart, Geoffrey E Hinton, Ronald J Williams. «Learning representations by back-propagating errors». В: *nature* 323.6088 (1986), с. 533—536.
- [33] Sepp Hochreiter, Schmidhuber. «Long short-term memory». В: *Neural computation* 9.8 (1997), с. 1735—1780.
- [34] Junyoung Chung и др. «Empirical evaluation of gated recurrent neural networks on sequence modeling». В: *arXiv preprint arXiv:1412.3555* (2014).
- [35] Anthony FC Wallace. *Plans and the structure of Behavior*. 1960.
- [36] Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio. «Neural machine translation by jointly learning to align and translate». В: *arXiv preprint arXiv:1409.0473* (2014).
- [37] Ashish Vaswani и др. «Attention is all you need». В: *Advances in neural information processing systems* 30 (2017).
- [38] Noam Chomsky. *Syntactic structures*. Mouton de Gruyter, 2002.
- [39] Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan kaufmann, 1988.
- [40] Ian Goodfellow и др. «Generative adversarial networks». В: *Communications of the ACM* 63.11 (2020), с. 139—144.
- [41] Yang Song и др. «Score-based generative modeling through stochastic differential equations». В: *arXiv preprint arXiv:2011.13456* (2020).
- [42] Mark A Kramer. «Nonlinear principal component analysis using autoassociative neural networks». В: *AIChE journal* 37.2 (1991), с. 233—243.
- [43] João Ascenso, Evgeniy Upenik. «White paper on jpeg ai scope and framework v1.0». В: *ISO/IEC JTC 1/SC 29/WG1 N90049* (2021).
- [44] Diederik P Kingma, Max Welling. «Auto-encoding variational bayes». В: *arXiv preprint arXiv:1312.6114* (2013).
- [45] Laurent Dinh, Jascha Sohl-Dickstein, Samy Bengio. «Density estimation using real nvp». В: *arXiv preprint arXiv:1605.08803* (2016).
- [46] Laurent Dinh, David Krueger, Yoshua Bengio. «Nice: Non-linear independent components estimation». В: *arXiv preprint arXiv:1410.8516* (2014).
- [47] Yann LeCun и др. «A tutorial on energy-based learning». В: *Predicting structured data* 1.0 (2006).
- [48] Alec Radford и др. «Language models are unsupervised multitask learners». В: *OpenAI blog* 1.8 (2019), с. 9.

- [49] Ken Thompson. «Programming techniques: Regular expression search algorithm». В: *Communications of the ACM* 11.6 (1968), с. 419—422.
- [50] ВВ Бочаров, ДВ Грановский. «Программное обеспечение для коллективной работы над морфологической разметкой корпуса». В: *Труды международной конференции «Корпусная лингвистика-2011»*. -СПб.: С.-Петербургский государственный университет. 2011.
- [51] Zellig S Harris. «Distributional structure». В: *Word* 10.2-3 (1954), с. 146—162.
- [52] Tomas Mikolov и др. «Efficient estimation of word representations in vector space». В: *arXiv preprint arXiv:1301.3781* (2013).
- [53] Jeffrey Pennington, Richard Socher, Christopher D Manning. «Glove: Global vectors for word representation». В: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014, с. 1532—1543.
- [54] Piotr Bojanowski и др. «Enriching word vectors with subword information». В: *Transactions of the association for computational linguistics* 5 (2017), с. 135—146.
- [55] Jacob Devlin и др. «Bert: Pre-training of deep bidirectional transformers for language understanding». В: *arxiv preprint arxiv:1810.04805* (2018).
- [56] Timothy Hospedales и др. «Meta-learning in neural networks: A survey». В: *IEEE transactions on pattern analysis and machine intelligence* 44.9 (2021), с. 5149—5169.
- [57] Alec Radford и др. «Learning transferable visual models from natural language supervision». В: *International conference on machine learning*. PMLR. 2021, с. 8748—8763.
- [58] Long Ouyang и др. «Training language models to follow instructions with human feedback». В: *Advances in neural information processing systems* 35 (2022), с. 27730—27744.
- [59] Albert Q Jiang и др. «Mistral 7B». В: *arXiv preprint arXiv:2310.06825* (2023).
- [60] Albert Q Jiang и др. «Mixtral of experts». В: *arXiv preprint arXiv:2401.04088* (2024).
- [61] Wayne Xin Zhao и др. «A survey of large language models». В: *arXiv preprint arXiv:2303.18223* (2023).
- [62] Shengyu Zhang и др. «Instruction tuning for large language models: A survey». В: *arXiv preprint arXiv:2308.10792* (2023).
- [63] Dan Hendrycks и др. «Measuring massive multitask language understanding». В: *arXiv preprint arXiv:2009.03300* (2020).
- [64] Wanjun Zhong и др. «Agieval: A human-centric benchmark for evaluating foundation models». В: *arXiv preprint arXiv:2304.06364* (2023).

- [65] Francois Chollet. «On the measure of intelligence». В: *arXiv preprint arXiv:1911.01547* (2019).
- [66] Neel Guha и др. «Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models». В: *Advances in Neural Information Processing Systems* 36 (2024).
- [67] David Rein и др. «Gpqa: A graduate-level google-proof q&a benchmark». В: *arXiv preprint arXiv:2311.12022* (2023).
- [68] Karl Cobbe и др. «Training verifiers to solve math word problems». В: *arXiv preprint arXiv:2110.14168* (2021).
- [69] Adam Karvonen. *chess_gpt_eval*. 2024.
- [70] Patrick Lewis и др. «Retrieval-augmented generation for knowledge-intensive nlp tasks». В: *Advances in Neural Information Processing Systems* 33 (2020), с. 9459—9474.
- [71] Aaron Parisi, Yao Zhao, Noah Fiedel. «Talm: Tool augmented language models». В: *arXiv preprint arXiv:2205.12255* (2022).
- [72] Robin Rombach и др. «High-resolution image synthesis with latent diffusion models». В: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, с. 10684—10695.
- [73] Daniel Bell. «The coming of post-industrial society». В: *Social Stratification, Class, Race, and Gender in Sociological Perspective, Second Edition*. Routledge, 2019, с. 805—817.
- [74] Jeremy Bentham. *The collected works of Jeremy Bentham: An introduction to the principles of morals and legislation*. Clarendon Press, 1996.
- [75] John F Nash и др. «The bargaining problem». В: *Econometrica* 18.2 (1950), с. 155—162.
- [76] John G Riley, William F Samuelson. «Optimal auctions». В: *The American Economic Review* 71.3 (1981), с. 381—392.
- [77] Leonid Hurwicz. «Optimality and informational efficiency in resource allocation processes». В: *Mathematical methods in the social sciences* (1960).
- [78] Allan Gibbard. «Manipulation of voting schemes: a general result». В: *Econometrica: journal of the Econometric Society* (1973), с. 587—601.
- [79] TE Harris, FS Ross. *Fundamentals of a method for evaluating rail net capacities*. Тех. отч. Rand Corporation, 1955.
- [80] Alvin E Roth, Tayfun Sönmez, M Utku Ünver. «Kidney exchange». В: *The Quarterly journal of economics* 119.2 (2004), с. 457—488.
- [81] Jeremy Bulow, John Roberts. «The simple economics of optimal auctions». В: *Journal of political economy* 97.5 (1989), с. 1060—1090.

- [82] Roger B Myerson. «Optimal auction design». В: *Mathematics of operations research* 6.1 (1981), с. 58—73.
- [83] Ralph Allan Bradley, Milton E Terry. «Rank analysis of incomplete block designs: I. The method of paired comparisons». В: *Biometrika* 39.3/4 (1952), с. 324—345.
- [84] Arpad E Elo. «The proposed uscf rating system, its development, theory, and applications». В: *Chess life* 22.8 (1967), с. 242—247.
- [85] Stav Yanovsky и др. «One size does not fit all: A study of badge behavior in stack overflow». В: *Journal of the Association for Information Science and Technology* 72.3 (2021), с. 331—345.
- [86] David Easley, Arpita Ghosh. «Incentives, Gamification, and Game Theory: An Economic Approach to Badge Design». В: *ACM Transactions on Economics and Computation* 4 (июнь 2016), с. 1—26.
- [87] Benjamin S Bloom. «The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring». В: *Educational researcher* 13.6 (1984), с. 4—16.
- [88] Виктор Николаевич Кругликов, Евгений Владимирович Платонов, Юрий Александрович Шаранов. «Деловые игры и другие методы активизации познавательной деятельности». В: (2006).
- [89] Peter Jarvis. *The Routledge international handbook of lifelong learning*. Routledge London, 2009.
- [90] Mihaly Csikszentmihalyi, Sami Abuhamdeh, Jeanne Nakamura. «Flow». В: *Handbook of competence and motivation* (2005), с. 598—608.
- [91] Martin BH Everaert и др. «Structures, not strings: Linguistics as part of the cognitive sciences». В: *Trends in cognitive sciences* 19.12 (2015), с. 729—743.
- [92] Richard Montague и др. «Universal grammar». В: *1974* (1970), с. 222—46.
- [93] Robert L Brennan, National Council on Measurement in Education и др. *Educational measurement*. Praeger Publishers, 2006.
- [94] Ross E Traub. «Classical test theory in historical perspective». В: *Educational Measurement* 16 (1997), с. 8—13.
- [95] Arnold Van Alphen и др. «Likert or Rasch? Nothing is more applicable than good theory». В: *Journal of advanced nursing* 20.1 (1994), с. 196—201.
- [96] Frederic M Lord. *Applications of item response theory to practical testing problems*. Routledge, 2012.
- [97] Frederic M Lord. «The measurement of growth». В: *ETS Research Bulletin Series* 1956.1 (1956), с. i—22.

- [98] R Darrell Bock, Murray Aitkin. «Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm». В: *Psychometrika* 46.4 (1981), с. 443—459.
- [99] Chris Abbott. *ICT: Changing education*. Routledge, 2003.
- [100] Nikos Manouselis и др. «Recommender systems in technology enhanced learning». В: *Recommender systems handbook* (2011), с. 387—415.
- [101] Anisia Katinskaia, Javad Nouri, Roman Yangarber. «Revita: a language-learning platform at the intersection of ITS and CALL». В: *Proceedings of the eleventh international conference on language resources and evaluation (LREC 2018)*. 2018.
- [102] Marthe Straatemeier и др. *Math Garden: A new educational and scientific instrument*. 2014.
- [103] Albert T Corbett, John R Anderson. «Knowledge tracing: Modeling the acquisition of procedural knowledge». В: *User modeling and user-adapted interaction* 4 (1994), с. 253—278.
- [104] Okan Bulut и др. «An introduction to Bayesian knowledge tracing with pyBKT». В: *Psych* 5.3 (2023), с. 770—786.
- [105] Robert Tinn и др. «Fine-tuning large neural language models for biomedical natural language processing». В: *Patterns* 4.4 (2023).
- [106] Chia-Chien Hung и др. «Multi2WOZ: A robust multilingual dataset and conversational pretraining for task-oriented dialog». В: *arXiv preprint arXiv:2205.10400* (2022).
- [107] Anna Dmitrieva. «Automatic text simplification of Russian texts using control tokens». В: *Proceedings of the 9th Workshop on Slavic Natural Language Processing 2023 (SlavicNLP 2023)*. 2023, с. 70—77.
- [108] Vladimir Ivanov, Elbayoumi Mohamed Gamal. «A new dataset for sentence-level complexity in Russian». В: *Proceedings of the International Conference “Dialogue. T. 2023*. 2023.
- [109] Lukas Blecher и др. «Nougat: Neural optical understanding for academic documents». В: *arXiv preprint arXiv:2308.13418* (2023).
- [110] Ray Smith. «An overview of the Tesseract OCR engine». В: *Ninth international conference on document analysis and recognition (ICDAR 2007)*. Т. 2. IEEE. 2007, с. 629—633.
- [111] Zejiang Shen и др. «Layoutparser: A unified toolkit for deep learning based document image analysis». В: *Document Analysis and Recognition-ICDAR 2021: 16th International Conference, Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part I* 16. Springer. 2021, с. 131—146.
- [112] Colin B Clement и др. «On the use of arxiv as a dataset». В: *arXiv preprint arXiv:1905.00075* (2019).

- [113] Joseph Redmon и др. «You only look once: Unified, real-time object detection». В: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, с. 779—788.
- [114] Jinze Bai и др. «Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond». В: (2023).
- [115] Sebastien Bubeck, Chandrasekaran. «Sparks of artificial general intelligence: Early experiments with gpt-4». В: *arXiv preprint arXiv:2303.12712* (2023).
- [116] Sebastian Bordt, Ulrike von Luxburg. «Chatgpt participates in a computer science exam». В: *arXiv preprint arXiv:2303.09461* (2023).
- [117] Алиса Назаровна Аль-Кайси, Алла Леонидовна Архангельская, Ольга Ивановна Руденко-Моргун. «Интеллектуальный голосовой помощник Алиса на уроках русского языка как иностранного (уровень А1)». В: *Филологические науки. Вопросы теории и практики* 12.2 (2019), с. 239—244.
- [118] Arya Bulusu и др. «An Automated Graphing System for Mathematical Pedagogy». В: ().
- [119] Yaqing Wang и др. «Generalizing from a few examples: A survey on few-shot learning». В: *ACM computing surveys (csur)* 53.3 (2020), с. 1—34.
- [120] M.Korobov. «Morphological Analyzer and Generator for Russian and Ukrainian Languages». В: 2015, с. 320—332.
- [121] Darya Bogoradnikova и др. «Multilingual sentiment analysis and toxicity detection for text messages in russian». В: *2021 29th Conference of Open Innovations Association (FRUCT)*. IEEE. 2021, с. 55—64.
- [122] David R Hunter. «MM algorithms for generalized Bradley-Terry models». В: *The annals of statistics* 32.1 (2004), с. 384—406.
- [123] Max Wertheimer. «Laws of organization in perceptual forms.» В: (1938).
- [124] James Craig и др. «Accessible rich internet applications (WAI-ARIA) 1.0». В: *W3C Working Draft* (2009).
- [125] Mathieu Acher, François Esnault. «Large-scale analysis of chess games with chess engines: A preliminary report». В: *arXiv preprint arXiv:1607.04186* (2016).
- [126] Anis Yazidi и др. «Balanced difficulty task finder: an adaptive recommendation method for learning tasks based on the concept of state of flow». В: *Cognitive Neurodynamics* 14.5 (2020), с. 675—687.
- [127] Inchi Hu. «Strong consistency in stochastic regression models via posterior covariance matrices». В: *Biometrika* 84.3 (1997), с. 744—749.
- [128] Inchi Hu. «On sequential designs in nonlinear problems». В: *Biometrika* 85.2 (1998), с. 496—503.

- [129] Edward W Ng, Murray Geller. «A table of integrals of the error functions». В: *Journal of Research of the National Bureau of Standards B* 73.1 (1969), с. 1—20.
- [130] Bradley Efron, Robert J Tibshirani. *An introduction to the bootstrap*. Chapman и Hall/CRC, 1994.
- [131] Jacob Marschak, Roy Radner. *Economic Theory of Teams*. 1972.
- [132] Roy Radner. «Team decision problems». В: *The Annals of Mathematical Statistics* 33.3 (1962), с. 857—881.
- [133] Theodore Groves. «Incentives in teams». В: *Econometrica: Journal of the Econometric Society* (1973), с. 617—631.
- [134] Jere R Francis, Inder K Khurana, Raynolde Pereira. «Disclosure incentives and effects on cost of capital around the world». В: *The accounting review* 80.4 (2005), с. 1125—1162.
- [135] Harold W Kuhn. «The Hungarian method for the assignment problem». В: *Naval research logistics quarterly* 2.1-2 (1955), с. 83—97.
- [136] Bengt Holmstrom. «Moral hazard in teams». В: *The Bell journal of economics* (1982), с. 324—340.

Дополнительные материалы

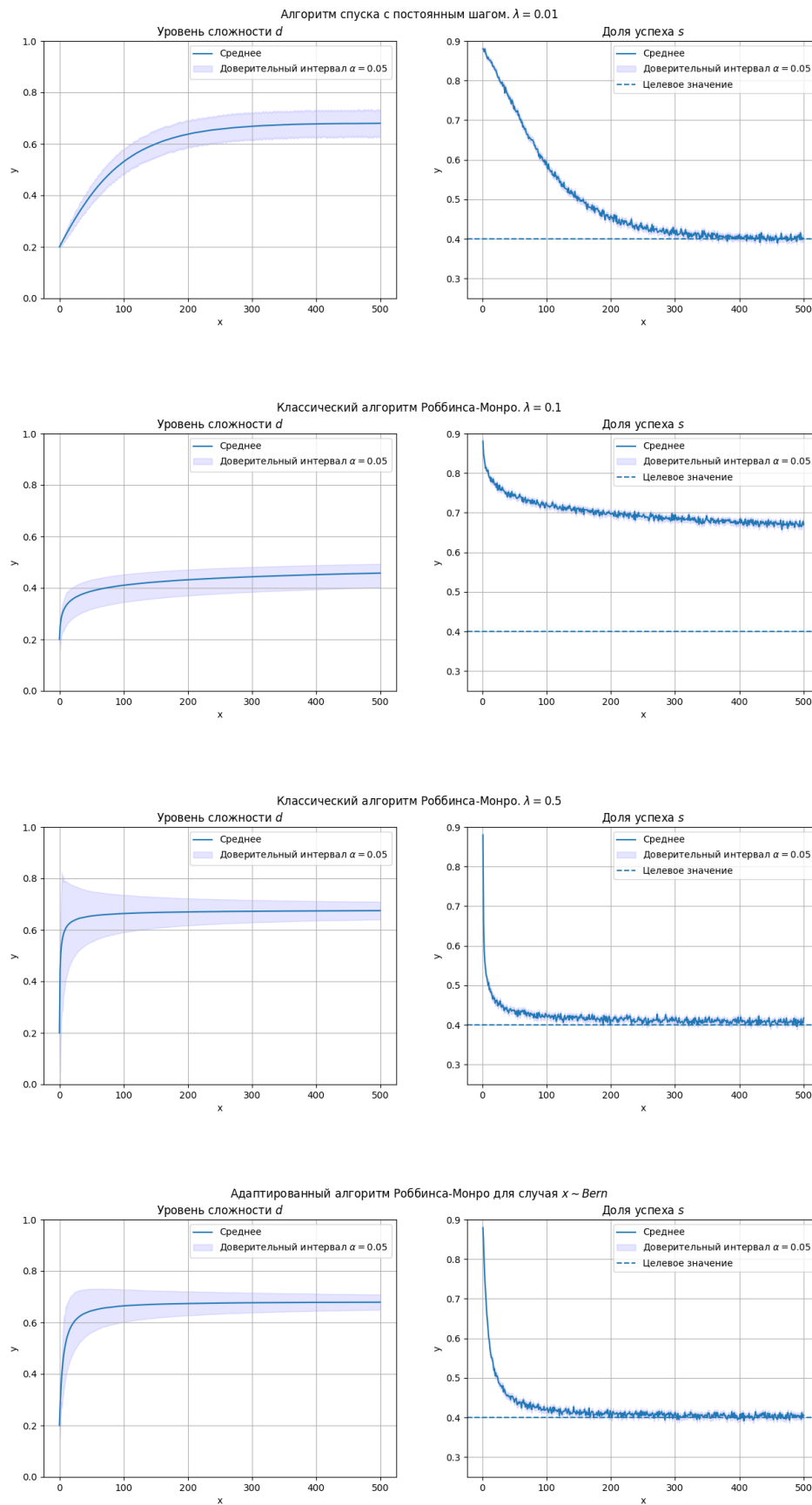
5.3 Используемая литература

1. Колмогоров А. Н. и др. Алгебра и начала анализа : учебное пособие для 10-го класса средней школы;
2. Алгебра и начала анализа, 9 класс / под ред. А. Н. Колмогорова. — 1975 г.;
3. Перельман Я. И. Живая геометрия теория и задачи. — 1930 г.;
4. Извольский Н. А. Геометрия на плоскости (планиметрия);
5. Беляева Э. С., Монахов В. М. Экстремальные задачи. — 1977 г.;
6. Александров А. Д. и др. Геометрия пробный учебник для 9—10 классов средней школы. — 1983 г.;
7. Гарднер М. Есть идея! — 1982 г.;
8. Дорф П. Я. Наглядные пособия по математике и методика их применения в средней школе. — 1960 г.;
9. Лиман М. М. Практические задачи по геометрии. — 1961 г.;
10. Скопец З. А., Жаров В. А. Задачи и теоремы по геометрии (планиметрия). — 1962 г.;
11. Клопский В. М. и др. Геометрия : учебное пособие для 9—10 классов / В. М. Клопский, З. А. Скопец, М. И. Ягодовский 1978 г.;
12. Овчинкин В.А. Сборник задач по общему курсу физики. Часть 1. Механика. Термодинамика и молекулярная физика - 2005 г.;
13. Овчинкин В.А., Раевский А.О., Ципенюк Ю.М. Сборник задач по общему курсу физики. Часть 3. Атомная и ядерная физика. Строение вещества 2009 г.;
14. Козел С.М., Лейман В.Г., Локшин Г.Р., Овчинкин В.А., Прут Э.В. Сборник задач по общему курсу физики. Часть 2. Электричество и магнетизм. Оптика 2000 г.;

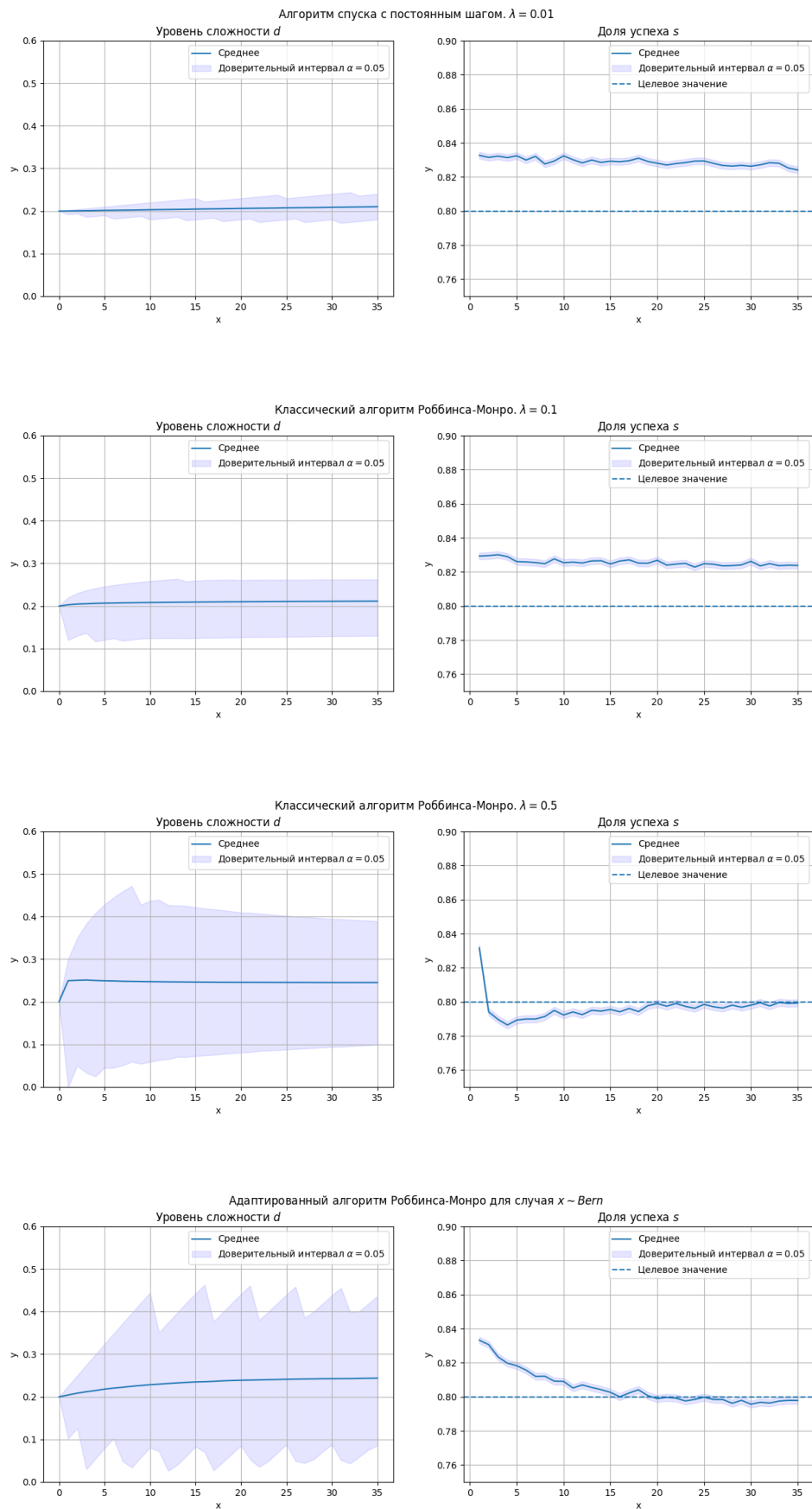
15. Беклемишев Д.В. Курс аналитической геометрии и линейной алгебры 2005 г.;
16. Генденштейн, Кирик, Гельфгат: 1001 задача по физике с ответами, указаниями, решениями 2001г.;

5.4 Численное моделирование

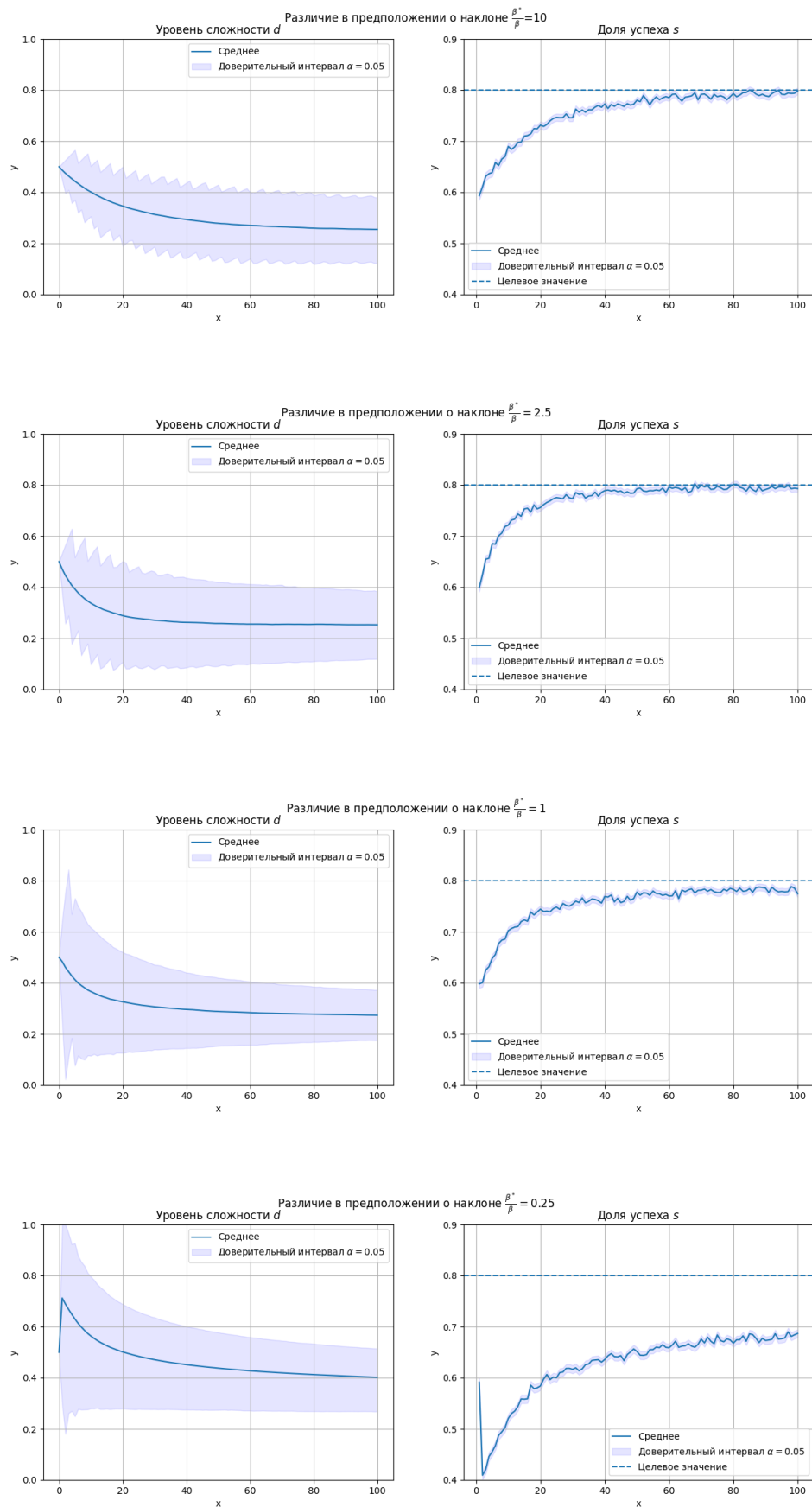
Эксперимент 1



Эксперимент 2



Эксперимент 3



Эксперимент 4

