

Содержание

1 Математические основы машинного обучения	5
1.1 Методы оптимизации	5
1.1.1 Градиентный спуск	7
1.1.2 Распределенное обучение	10
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	30
2.2.1 Энергетические модели	35
2.3 Обработка естественного языка	37
2.3.1 Формальные языки	37
2.3.2 Представление языка в виде вложений	39
2.3.3 Большие языковые модели	40
3 Математические методы в педагогике	43
3.1 Значение образования	43
3.1.1 Функция блага	44
3.1.2 Механизм	45
3.1.3 Систем наград	48
3.2 Структура методического материала	49
3.2.1 Методы образования	50
3.2.2 Психологические подходы к описанию интеллектуального разви- тия	51
3.3 Методическая задача	54
3.3.1 Тестовые системы	54
4 Описание работы	58
4.1 Подготовка данных для обучения	58
4.1.1 Разметка	59
4.1.2 Распознание текста	59
4.2 Приложение	63

4.2.1	Проект интеллектуального ассистента	64
4.2.2	Подготовка веб-приложения	66
4.2.3	Описание разработанных игр	67
4.3	Рейтинговая система	68
4.3.1	Адаптация рейтинговой системы	69
4.3.2	Численные эксперименты	73
5	Заключение	79
5.1	Итоги работы	79
5.2	Дальнейшая работа	79
5.2.1	Благодарности	81
Список литературы		81
6	Дополнительные материалы	82
6.1	Численное моделирование	82
6.1.1	Приложение	83

Генеративное моделирование в сфере образования активно развивается и является одним из актуальных направлений научной работы. С 2019 года UNESCO регулярно публикует аналитические доклады [[unesco2019beijing](#)][[annuvs2024education](#)], содержащие обзоры применения искусственного интеллекта в образовании. Одним из наиболее активно развивающихся 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 [[lhamatouvron2023](#)]. 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 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 1 работы представлено теоретическое описание аппарата оптимизации и статистического моделирования. Рассмотрены методы градиентного стохастического спуска, обучения графовых вероятностных моделей и транспортных задач. Выполнен анализ ключевых теорем градиентного спуска и стохастической аппроксимации.

Во второй главе 2 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 3 посвящена математическому описанию задач в системе образования. Описаны психометрические 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 раз-

вития, разработанные психологами Жаном Пиаже [piaget1952origins] и Львом Выготским [РўСКРуР«СЂСѓР҃РЧР€2014РёСЌСЇРъРхР„РЧРх]. Описана качественная модель выбора сложности заданий согласно Михею Чиксентмихай [chen2007flow].

В четвертой главе 4 описан порядок исследований и вывод ключевых теорем работы. Приведено описание подхода к сбору данных из открытых источников, количественное описание полученных корпусов. Описан подход к совершенствованию системы оптического распознавания символов на русском языке. Выполнен синтез оптимального алгоритма спуска к заданному параметру бернуллевского 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 [kantorovich1960mathematical], В. В. Наумовичем и А. Я. Червоненкисом [РўРєР»Р,,РЧРЭ1974СЂРХР«СГРЧCS] в теории минимизации эмпирического риска и Л. В. Кантаровичем в теории оптимального планирования [kantorovich1960mathematical]. Современные достижения во многом являются результатом адаптации разработанного теоретического аппарата к применению мощных вычислительных систем, ставших доступными 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 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 специфичной постановки, как правило, существует специальная теория, адаптирующая базовый аппарат оптимизации под задачу. Общими же являются методы, использующие производные оптимизируемой функции для поиска численной схемы, обеспечивающей эффективный поиск экстремума.

Современная задача оптимизации с учетом случайности в спуске записывается как [nesterov2015universal]:

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

где пара (x, ξ) является результатом наблюдения с детерминированным параметром x и случайной величиной ξ . В секции для упрощения будут разобраны 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 $\mu > 0$.

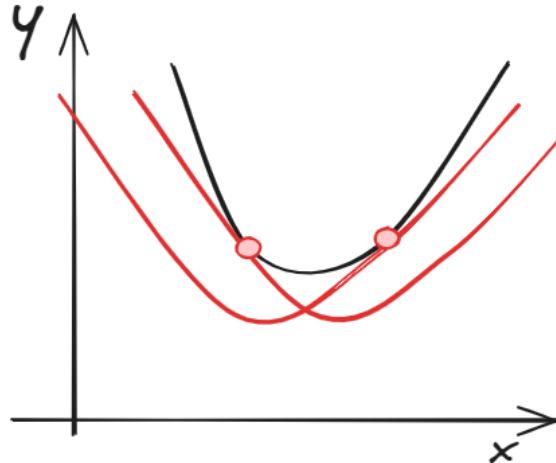


Рис. 1.1: Сильная выпуклость позволяет выполнить квадратичную оценку снизу

Для выпуклых функций справедливо неравенство Йенсена, широко применяю-

щееся для задания оценки снизу на скорость сходимости численной схемы:

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

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 отлика является усреднение по подвыборке (от англ. batch).

Определение: Усредненным по подвыборке мощностью 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 [polyak1990new], 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: 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: В силу 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 продвинутые 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 тя-

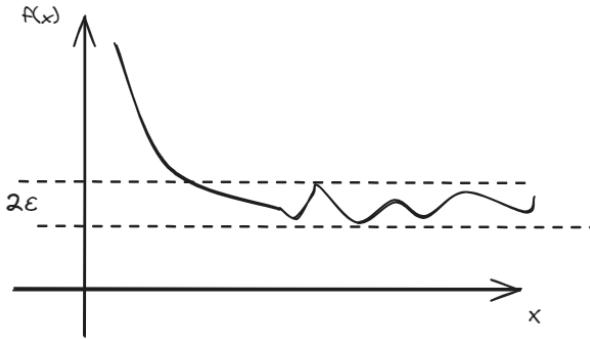


Рис. 1.2: Устойчивость метода при выходе на плато

желого шарика:

1. AdaGrad [duchi2011adaptive] задает адаптивный шаг спуска для каждого параметра в отдельности с учетом градиента $\nabla_{\theta}\mathcal{L}(\theta)$ на каждом шаге:

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

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

2. RMSProp [krizhevsky2012imagenet] 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 [kingma2014adam] совмещающий метод скользящего среднего по градиенту и параметру накопления градиента:

$$\begin{aligned} x_{t+1} &= x_t - \frac{\eta}{\sqrt{g_t + \epsilon}} \cdot \frac{v_{t+1}}{1 - \beta^t} \\ v_{t+1} &= \beta v_t + (1 - \beta) \nabla L(x_t) \\ g_t &= \mu g_{t-1} + (1 - \mu) \nabla L(x_t) \cdot \nabla \mathcal{L}(\theta)^2 \end{aligned} \quad (1.19)$$

Отметим, что метод не обязательно сходится даже в выпуклой постановке [reddi2019convex]. 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 алгоритмы 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.

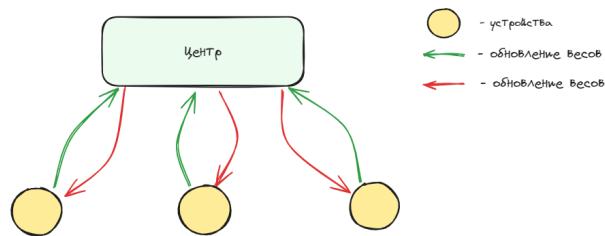


Рис. 1.3: Распределенное обучение включает локальные и глобальные ресурсы, разделяемые раундами коммуникации

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.

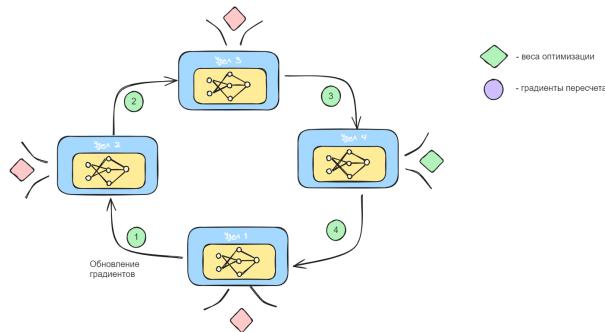


Рис. 1.4: Одной из распространенных архитектур взаимодействия между вычислительными узлами является кольцевая.

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** [richtarik2016parallel] 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 $\hat{\downarrow}_2$ 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:

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

Теорема: [stich2019unified] 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 [lamport2019time] и Paxos [pease1980reaching]. 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 любое 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 несмешеным откликом. Наиболее известным результатом теории стохастической аппроксимации является алгоритм Роббинса-Монро, задающий численную схему, обеспечивающую сходимость к корню заданного уравнения.

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 [blum1954approximation], используем рекуррентную схему связи между ошибками на каждом шаге алгоритма $\gamma_t = d_t - d^*$. В этом случае правило обновления Роббинса-Монро запишется как:

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

Раскрываем квадрат разности:

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

Используем несмешенность оценки $E_{x \sim p(x|s)} x = s$:

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

Получаем связь между ошибками на каждом шаге γ_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 $\lim_{n \rightarrow \infty} \gamma_n$ нулю следует из $\sum_{j=1}^{\infty} a_j = \infty$. Действительно, зададим ряд $m_n = \frac{\gamma_n}{a_n}$, 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 $\lim_{n \rightarrow \infty} \gamma_n = 0$. ■

В работе [sacks1958asymptotic] показано, что в случае если функции отклика s строго выпукла и дважды дифференцируема, то асимптотическая скорость сходимости линейна $\mathcal{O}(\frac{1}{n})$

Выбор коэффициента a_n в численной схеме существенно влияет на число шагов, необходимых для того, чтобы последовательность соплась. 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 существует модификации схемы, не требующие подбора гиперпараметра. Например, в работе [lai1979adaptive] был разработан метод выбора коэффициентов исходя из требований несмещенностии оценки $Ed_n = d$ и минимизации дисперсии $Dd_n = 0$ на каждом шаге. Такой подход позволил получить оптимальные численные коэффициенты для случая:

1. и цепи гауссовых распределений [liu2024robbins].

Современные исследователи дополняют оригинальную схему [РэСГ'РэСFCТнРъРЧР,,Рэ2007] заменой члена s^* на параметр шага b_n :

Случай $b_n \neq s^$* Модифицированная схема Монро-Роббинса $x_{n+1} = x_n - a_n(y_n - b_n)$ сходится при:

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

Доказательство: Принципиальная схема доказательства значительно изменяется

$$(d_n - d^*)^2 + a_t^2 E_{x_t \sim p(x|s(d))} (b_t - x_t)^2 - 2a_t E_{x_t \sim p(x|s(d))} [(b_t - x_t)(d_n - d^*)]. \quad (1.28)$$

■

Теорема для многомерного случая [xiong2018efficient] 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

$$\begin{aligned}
 & \mathbb{E} \left\{ (\mathbf{x}_{n+1} - \boldsymbol{\theta})^\top (\mathbf{x}_{n+1} - \boldsymbol{\theta}) \right\} \\
 &= \mathbb{E} \left(\mathbb{E} \left[\{ \mathbf{x}_n - \boldsymbol{\theta} - A_n (\mathbf{y}_n - \boldsymbol{\alpha}) \}^\top \right. \right. \\
 &\quad \times \{ \mathbf{x}_n - \boldsymbol{\theta} - A_n (\mathbf{y}_n - \boldsymbol{\alpha}) \} \mid \mathbf{x}_n] \left. \right) \\
 &= \mathbb{E} (\mathbf{x}_n - \boldsymbol{\theta})^\top (\mathbf{x}_n - \boldsymbol{\theta}) - 2 \mathbb{E} (\mathbf{x}_n - \boldsymbol{\theta})^\top A_n (\mathbf{m}_n^x - \boldsymbol{\alpha}) \\
 &\quad + \mathbb{E} \left\{ (\mathbf{y}_n - \boldsymbol{\alpha})^\top A_n^\top A_n (\mathbf{y}_n - \boldsymbol{\alpha}) \right\} \tag{1.29} \\
 &= \mathbb{E} (\mathbf{x}_1 - \boldsymbol{\theta})^\top (\mathbf{x}_1 - \boldsymbol{\theta}) - 2 \sum_{i=1}^n \mathbb{E} (\mathbf{x}_i - \boldsymbol{\theta})^\top A_i (\mathbf{m}_i^x - \boldsymbol{\alpha}) \\
 &\quad + \sum_{i=1}^n \mathbb{E} \left\{ (\mathbf{y}_i - \boldsymbol{\alpha})^\top A_i^\top A_i (\mathbf{y}_i - \boldsymbol{\alpha}) \right\} \\
 &\geq 0,
 \end{aligned}$$

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

- 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. $\sum_{n=1}^{\infty} \mathbb{E} \left\{ \mathbf{x}_n - \boldsymbol{\theta}^\top A_n (\mathbf{m}_n^x - \boldsymbol{\alpha}) \right\} < \infty$.
- Во вторых для $j = 1, \dots, p$, поскольку M_j функция распределения и $M'_j(\theta_j) > 0$, то 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 $u = \max \{ u_j : j = 1, \dots, p \}$. Let $\mathbf{a}_{j,n}$ be the j th column of A_n . Then,

:

$$\begin{aligned}
 & (\mathbf{x}_n - \boldsymbol{\theta})^\top A_n (\mathbf{m}_n^x - \boldsymbol{\alpha}) \\
 &= \sum_{j=1}^p (\mathbf{x}_n - \boldsymbol{\theta})^\top \mathbf{a}_{j,n} (x_{jn} - \theta_j) \frac{M_j(x_{jn}) - \alpha_j}{x_{jn} - \theta_j} \\
 &\geq \sum_{j=1}^p (\mathbf{x}_n - \boldsymbol{\theta})^\top \mathbf{a}_{j,n} (x_{jn} - \theta_j) \delta_{jn} \\
 &= (\mathbf{x}_n - \boldsymbol{\theta})^\top A_n \Delta_n (\mathbf{x}_n - \boldsymbol{\theta}),
 \end{aligned} \tag{1.30}$$

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

$$\delta_{jn} = \begin{cases} \ell, & \text{если } (\mathbf{x}_n - \boldsymbol{\theta})^\top \mathbf{a}_{j,n} (x_{jn} - \theta_j) \geq 0 \\ u, & \text{иначе} \end{cases} \tag{1.31}$$

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

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

Заметим, что для второго члена выполняется:

$$\begin{aligned}
 & \sum_{j \neq k} \sum_{n=1}^{\infty} \frac{1}{2} |a_{jk,n} \delta_{kn}| \{E(x_{jn} - \theta_j)^2 + E(x_{kn} - \theta_k)^2\} \\
 &= \frac{1}{2} \sum_{j=1}^p \sum_{n=1}^{\infty} E(x_{jn} - \theta_j)^2 \sum_{k \neq j} \{|a_{jk,n} \delta_{kn}| + |a_{kj,n} \delta_{jn}|\}.
 \end{aligned} \tag{1.32}$$

Предположения $|a_{jk,n}/a_{jj,n}| \rightarrow 0$ 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 :

$$\begin{aligned}
 & \sum_{k \neq j} |a_{jk,n} \delta_{kn}| + |a_{kj,n} \delta_{jn}| < \sum_{k \neq j} \left\{ \frac{a_{jj,n} \delta_{kn}}{2c(p-1)} + \frac{a_{jj,n} \delta_{jn}}{2c(p-1)} \right\} \\
 &< \frac{a_{jj,n} u}{c}
 \end{aligned} \tag{1.33}$$

Тогда сумма запишется как:

$$\begin{aligned}
 & \sum_{j=1}^p \sum_{n>N}^{\infty} a_{jj,n} \delta_{jn} E(x_{jn} - \theta_j)^2 \\
 & + \sum_{j \neq k} \sum_{n>N}^{\infty} a_{jk,n} \delta_{kn} E(x_{jn} - \theta_j)(x_{kn} - \theta_k) \\
 & \geq \sum_{j=1}^p \sum_{n>N}^{\infty} a_{jj,n} \delta_{jn} E(x_{jn} - \theta_j)^2 - \frac{1}{2} \sum_{j=1}^p \sum_{n>N}^{\infty} E(x_{jn} - \theta_j)^2 \\
 & \times \sum_{k \neq j} \{|a_{jk,n} \delta_{kn}| + |a_{kj,n} \delta_{jn}|\} \\
 & > \sum_{j=1}^p \sum_{n>N}^{\infty} \left(\delta_{jn} - \frac{u}{2c} \right) a_{jj,n} E(x_{jn} - \theta_j)^2
 \end{aligned} \tag{1.34}$$

Заметим, что оценка положительна исходя из выбора c . Это означает ограниченность. Следовательно, исходя $E(x_{jn} - \theta_j)^2$ сходится к 0 $\forall j = 1, \dots, p$, что обуславливает 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:

$$x_{n+1} = x_n + a_n [g(x_n) + \varepsilon_n], \tag{1.35}$$

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

$$t_n = \sum_{k=0}^n a_k \tag{1.36}$$

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 аппроксимирующая схему обыкновенное дифференциальное уравнение.

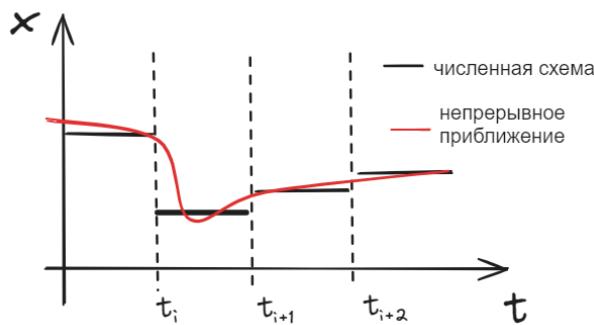


Рис. 1.5: Численная схема стохастической аппроксимации может быть приближена дифференциальным уравнением $\frac{dz}{dt} = g(z(t))$

Теорема Обыкновенное дифференциальное уравнения $\frac{dz_m}{dt} = g(z_m(t))$ с началь-

ным 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

$$z(t_m) = z(t_n) + \int_{t_m}^{t_n} z_m(u) du \quad (1.37)$$

Стохастический ряд приближений согласно алгоритму Роббинса-Монро запишется как:

$$x(t_n) = x_m + \sum_{k=m}^{n-1} a_k g(x_k) + \sum_{k=m}^{n-1} a_k \varepsilon_k \quad (1.38)$$

С учетом 1.37 и $z(t_m) = x_m$:

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

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

Рассмотрим искомое выражение как разность полученных представлений:

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

Воспользуемся липшицевостью функции $g(z)$:

$$\|z(t_n) - x(t_n)\| \leq \left\| \sum_{k=m}^{n-1} a_k \varepsilon_k \right\| + L \int_{t_m}^{t_n} |z(u) - x([u]_a)| du \quad (1.41)$$

Используем лемму Гронуолла для перехода к произведению вида:

$$\|z(t_n) - x(t_n)\| \leq \left\| \sum_{k=m}^{n-1} a_k \varepsilon_k \right\| \exp(L \int_{t_m}^{t_n} du) \leq C \left\| \sum_{k=m}^{n-1} a_k \varepsilon_k \right\| \exp(L(t_n - t_m)) \quad (1.42)$$

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

$$\left\| \sum_{k=m}^{n-1} a_k \varepsilon_k \right\| \leq \sum_{k=1}^n a_k \mathbb{E} \varepsilon_k^2 \leq \sigma^2 \sum_{k=1}^{\infty} a_k^2 \leq \infty \quad (1.43)$$

Тогда в силу ограниченности заданного ряда получим:

$$\lim_{t_n \rightarrow t_m \rightarrow 0} \|z(t_n) - x(t_n)\| = 0 \quad (1.44)$$

Или что эквивалентно:

$$\lim_{m \rightarrow \infty} \sup_{t \in [t_m, t_{m+1}]} |x(t) - z_m(t)| = 0, \quad (1.45)$$

■ Существенный вклад в развитие методов стохастической аппроксимации внёс Б. Т. Поляк [**polyak1990new**], предложивший метод усреднения управляющего параметра θ :

$$\hat{\theta}_{n+1} = \sum_{i=0}^n \theta_i. \quad (1.46)$$

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 :

$$\frac{a_n - a_{n+1}}{a_n} = o(a_n). \quad (1.47)$$

Заданный момент имеет скорость сходимости $\mathcal{O}(\frac{1}{\sqrt{n}})$ для произвольной выпуклой функции.

Зададим через $\sigma = \mathbf{E} [\nabla f(x_*, \xi) \nabla f(x_*, \xi)^T]$ матрицу Гессе в точке сходимости:

$$(\hat{x}^N - x_*) \rightarrow \mathcal{N}(0, \sigma^2) \quad (1.48)$$

Схема Стохастической аппроксимации Поляка-Руперта-Юдицкого, изложенная в работе Робинсона и Монро [**robbins1951stochastic**]:

$$x^{k+1} = x^k - \gamma_k \phi(\nabla_x f(x^k, \xi^k)) \quad (1.49)$$

Шаги $h_k \sim k^{-\alpha}$, $\alpha \in (\frac{1}{2}, 1)$. При этом ошибка рассчитывается для среднего

$$\bar{x}_n = \frac{1}{N} \sum_{k=1}^N x^k \quad (1.50)$$

1.2 Оптимальный транспорт

Задача оптимального транспорта (Optimal Transport)[**villani2009optimal**] 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 μ в распределение ν , минимизируя некоторую функцию стоимости. Функция стоимости с 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} :

$$\inf_{\gamma \in \Pi(\mu, \nu)} \int_{\mathcal{X} \times \mathcal{Y}} c(x, y) d\gamma(x, y), \quad (1.51)$$

, 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:

$$\inf_{\phi} \left(\int_{\mathcal{X}} \phi(x) d\mu(x) + \int_{\mathcal{Y}} \psi(y) d\nu(y) \right), \quad (1.52)$$

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

Заметим, что постановка Канторовича является более общей [**monge_vs_kantarovich**]. В отличие от постановки Монже оптимальный транспорт по Канторовичу допускает распределение вероятностной массы в непрерывном случае.

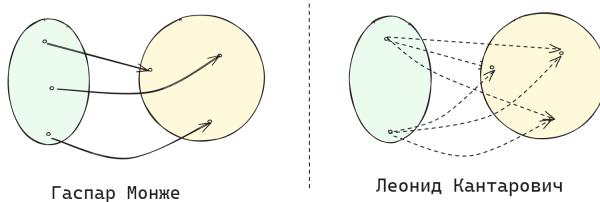


Рис. 1.6: Различие в подходе по Монже и Канторовичу. В постановке Канторовича задача релаксирует до непрерывного распределения

Итоговая стоимость оптимального транспортного плана называется метрикой Вассерштейна.

Определение Метрика Вассерштейна Пусть (X, d) — метрическое пространство, $P(X)$ — множество всех вероятностных мер на X . Для двух вероятностных мер μ и ν на X метрика Вассерштейна порядка p , где $p \geq 1$, определяется как:

$$W_p(\mu, \nu) = \left(\inf_{\gamma \in \Gamma(\mu, \nu)} \int_{X \times X} d(x, y)^p d\gamma(x, y) \right)^{1/p}, \quad (1.53)$$

, 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 $\mathcal{P}_2(\mathbb{R}^N)$ записывается как:

$$|\rho'_t| = \lim_{dt \rightarrow 0} \frac{\mathcal{W}_2(\rho_t, \rho_{t+dt})}{dt} \quad (1.54)$$

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

Определение Процесс Ланжевена с коэффициентом сноса, с потенциалом $\Phi(X)$ задается

$$dX_t = -\nabla\Phi(x)dt + \sqrt{2\beta^{-1}}dW_t \quad (1.55)$$

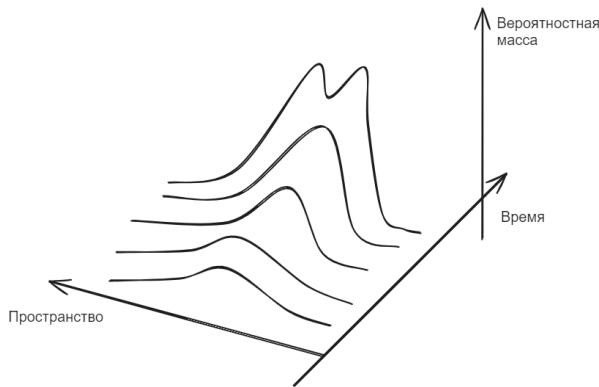


Рис. 1.7: Эволюция вероятностной массы в уравнение Ланжевена

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

$$\frac{\partial \rho_t}{\partial t} = \text{div}(\nabla\Phi(x)\rho_t) + \beta^{-1}\Delta\rho_t \quad (1.56)$$

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

Определение **Функционал Фоккера-Планка** для распределения ρ записывается как:

$$\mathcal{F}_{FP}(\rho) = \int \Phi(x)d\rho(x) + \beta^{-1} \int \log \rho(x)d\rho(x) \quad (1.57)$$

Научная группа Йордана-Кинана-Отто в работе [jordan1998variational] показала, что маргинальные вероятностные меры процесса Ланжевена подчиняются уравнению градиентного потока Вассерштейна относительно функционала Фоккера-Планка.

Определение Схема Йордана-Кинана-Отто (JKO) задает правило обновления урав-

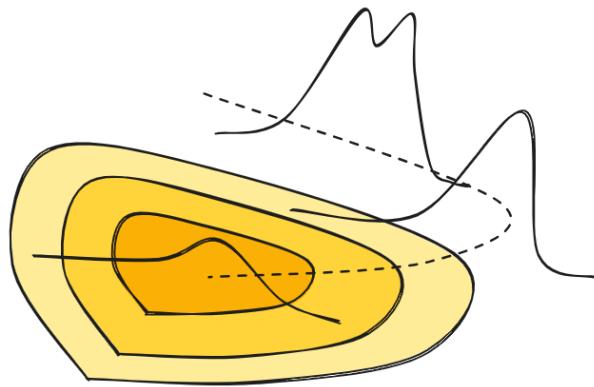


Рис. 1.8: Визуализация постановки уравнения Фоккера-Планка в вариационной форме

нения вероятности в виде минимизации функционала энергии и расстояния:

$$\rho^{n+1} = \operatorname{argmin}_{\rho} \left(\frac{1}{2\tau} W_2^2(\rho, \rho^n) + \mathcal{F}(\rho) \right), \quad (1.58)$$

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

$$\mathcal{F}(\rho) = \int_V f(\rho(x)) dx + \int_V V(x)\rho(x) dx, \quad (1.59)$$

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 σ , ReLU [agarap2018deep] и GeLU [hendrycks2016gaussian]:

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

Функция активации вводится для добавления нелинейности в модель, 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 модель, состоящая из слоев нейронов:

В случае многослойной нейронной сети, выходы нейронов одного слоя становятся

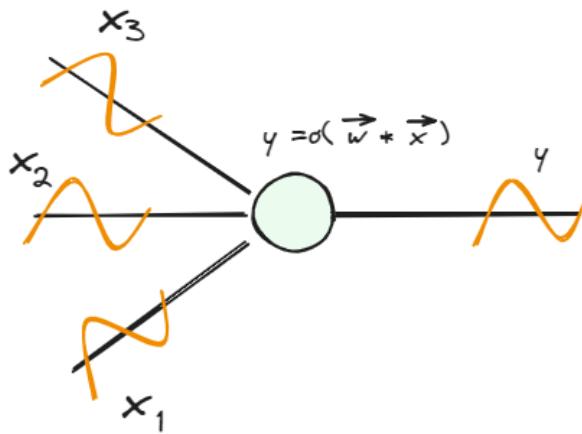


Рис. 2.1: Прямой процесс наложения шума и обратный процесс коррекции ошибки [stablediffusion]

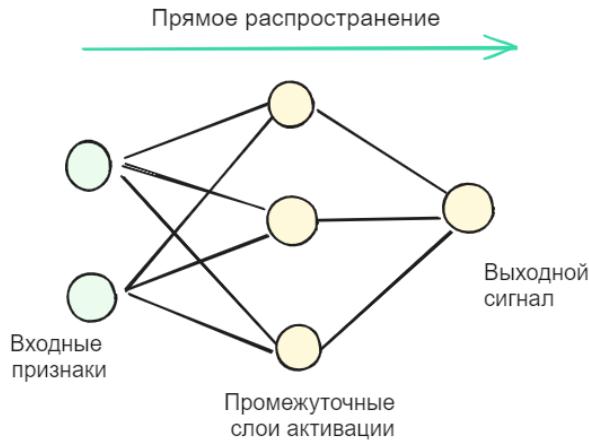


Рис. 2.2: Преобразование сигнала выполняется в промежуточных слоях активации в ходе прямого распространения сигнала

входами для следующего слоя, образуя цепочку преобразований. Процесс распространения через нейроны последовательных слоев называется прямым распространением (forward propagation).

Во время обучения модель минимизирует функцию потерь L , которая оценивает отличие между предсказанным результатом y_i и истинным значением y :

$$L = \frac{1}{N} \sum_{i=1}^N L(y_i, \hat{y}_i), \quad (2.3)$$

Обучение нейронной сети выполняется путем настройки весов w и смещений b с использованием алгоритмов оптимизации, таких как градиентный спуск ??.

Кодировщики и декодировщики

Архитектура кодировщик-декодировщик широко распространена в машинном обучении, теории кодирования, компрессии и криптографии. В моделировании нейросе-

тей также используют этот подход для задач:

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

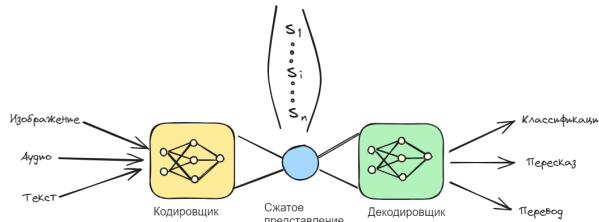


Рис. 2.3: Преобразование сигнала выполняется в промежуточных слоях активации в ходе прямого распространения сигнала.

Такая архитектура

Определение: Рекуррентными нейронными сетями называют такие, в которых используются скрытые состояния предыдущих слоев для расчета следующих.

$$\begin{aligned} h_t &= \sigma(W_x x_t + W_h h_{t-1} + b_h) \\ y_t &= \sigma(W_y h_t + b_y), \end{aligned} \tag{2.4}$$

где

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

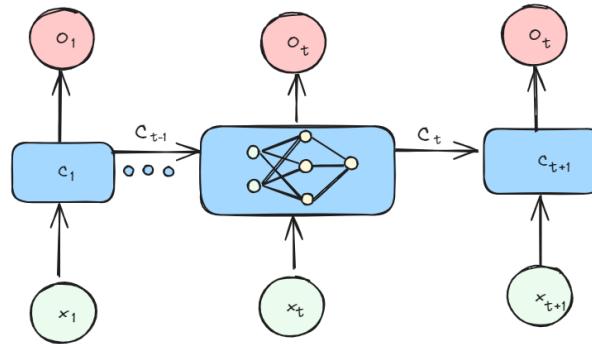


Рис. 2.4: Каждая ячейка рекуррентной нейронной сети выполняет обновление представления.

Ошибка модели для случая обработки последовательностей равной длины запишется как:

$$\mathcal{L}(\hat{y}, y) = \sum_{t=1}^{T_y} \mathcal{L}(\hat{y}_t, y_t) \tag{2.5}$$

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

$$\frac{\partial \mathcal{L}^{(T)}}{\partial W} = \sum_{t=1}^T \frac{\partial \mathcal{L}^{(T)}}{\partial W} \quad (2.6)$$

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

- долгая короткая память [[ochreiter1997long](#)];
- управляемый рекуррентный блок [[chung2014empirical](#)].

Механизм внимания, основанный на модели рабочей памяти [[wallace1960plans](#)], позволил значительно улучшить результаты решения задач перевода, пересказа и "понимания" языка [[bahdanau2014neural](#)]. Механизм внимания вычисляет вектор внимания α , который определяет важность каждого элемента входной последовательности на текущем временном шаге [[bahdanau2014neural](#)]:

$$\alpha_t = \text{softmax}(f(h_t, X)), \quad (2.7)$$

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

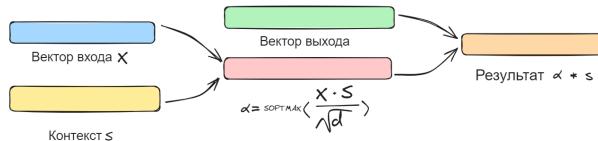


Рис. 2.5: Механизм внимания в архитектуре Transformer (Self Attention) [[vaswani2017attention](#)].

Механизм внимания позволяет модели "сосредоточиться" на наиболее значимых частях 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, предназначеннной для обработки последовательных данных, таких как тексты или временные ряды.

Она была предложена в работе [vaswani2017attention] 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 частях входных данных при выполнении задач, таких как машинный перевод или обработка текста.

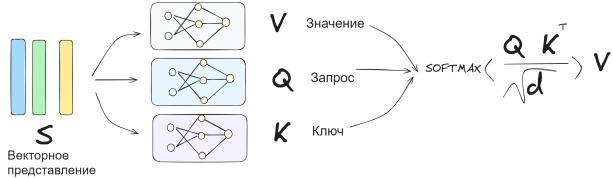


Рис. 2.6: Механизм внимания в архитектуре Transformer [vaswani2017attention].

Механизм внимания в Transformer состоит из трех основных частей:

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

$$\begin{aligned} q &= W_q x \\ k &= W_k x, \\ v &= W_v x, \end{aligned} \tag{2.8}$$

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

2. Вычисление векторов запроса и ключа, для каждого элемента x_i . Рассчитываются логиты e_{ij} :

$$e_{ij} = \frac{q \cdot k_j}{\sqrt{d_k}}, \tag{2.9}$$

где d_k - длина запроса,

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

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{j'} \exp(e_{ij'})} \tag{2.10}$$

Веса определяют важность каждого элемента данных, которые модель учитывает при решении конкретной задачи. После вычисления весов внимания, они умножаются на соответствующие значения (value) и суммируются, чтобы получить итоговый взвешенный вектор, который представляет собой выходной результат механизма внимания.

Применение

Методы решения задач компьютерного зрения могут основываться как классических подходах, так и на применении глубоких нейронных сетей.

Классические методы построены на анализе внутренней структуры и симметрии изображения, его семантике и характеристиках объектов на нем. Эти методы включают в себя алгоритмы обработки изображений, фильтрацию, выделение признаков (например, метод гистограмм градиентов или методы локтевых точек), шаблонное сопоставление и классификацию на основе характеристик объектов.

Определение: Сверточные нейронные сети (CNN) [lecun1989handwritten] представляют собой класс глубоких нейронных сетей, использующих специально разработанные для обработки структурно-связанных данных как изображения.

Эффективность CNN определяется их возможностью извлекать иерархические признаки из входных данных.

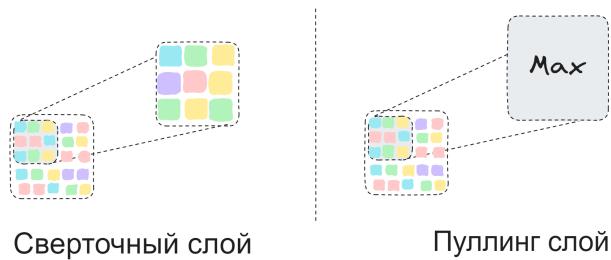


Рис. 2.7: Сверточный слой содержит параметрическое ядро, обеспечивающее выделение ключевых признаков. Пуллинг-слои выполняют заранее заданную аналитическую операцию.

Глубокие методы, основанные на сверточных нейронных сетях (CNN), стали широко распространенными и эффективными в решении задач компьютерного зрения. Эти методы автоматически изучают признаки изображений на различных уровнях абстракции, начиная от низкоуровневых признаков, таких как грани и текстуры, до высокоуровневых семантических признаков, связанных с объектами и их распознаванием.

Основными компонентами сверточных нейронных сетей являются:

- сверточные слои — выполняют операции свертки над входными данными с использованием фильтров или ядер, чтобы извлечь локальные пространственные признаки, такие как грани, углы и текстуры, что позволяет модели обнаруживать абстрактные особенности изображений на разных уровнях детализации;
- пуллинг-слои — предназначены для уменьшения пространственных размеров активаций, полученных после сверточных операций, путем объединения пикселей в заданных областях, что позволяет модели быть устойчивой к небольшим искажениям объектов на изображении и уменьшает количество параметров, снижает переобучение и способствует росту эффективности вычислений;
- полносвязанные слои обычно располагаются в конце архитектуры нейронной сети и используются для объединения высокоуровневых признаков, извлеченных предыдущими слоями, в предсказаниях или классификации.

Во время обучения сверточной нейронной сети параметры каждого слоя оптимизируются с использованием специальных методов, таких как обратное распространение ошибки и стохастический градиентный спуск. Целью является минимизация заданной функции потерь. Этот процесс позволяет настраивать параметры модели для эффективного извлечения признаков и выполнения конкретной задачи, такой как классификация изображений или сегментация объектов.

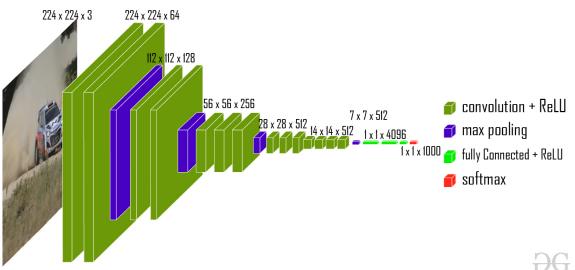


Рис. 2.8: Архитектура VGG16 [simonyan2014very]

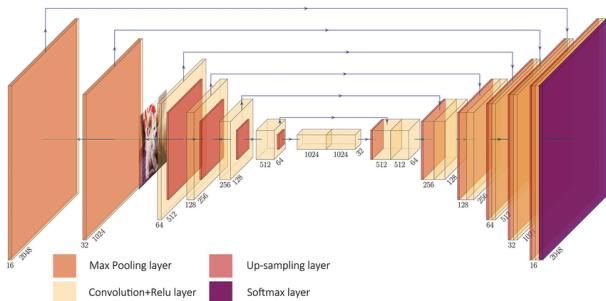


Рис. 2.9: Архитектура Unet [ronneberger2015u].

Архитектуры U-Net и ResNet 2.9 привнесли существенные модификации в оригинальные архитектуры сверточных сетей

- использование симметричной структуры, деконволюционные слои (upsampling) для восстановления пространственного разрешения;
 - блоки с пропуском, обеспечивающие плавное обучение глубоких сетей $x_{t+1} = x_t + \phi_t(x_t)$, где $\phi_t(x)$ — функция активации промежуточного слоя t .

Методы аугментации изображений в компьютерном зрении представляют собой техники, используемые для увеличения размера и разнообразия тренировочного набора данных путем применения различных преобразований к изображениям. Целью аугментации является создание дополнительных вариаций изображений, что помогает улучшить обобщающую способность моделей машинного обучения и снизить риск переобучения.

Основные методы аугментации включают в себя:

- изменение размера изображения (путем масштабирования);
 - поворот и отражение;
 - изменение яркости, контраста и насыщенности цветов;

- добавление шума или размытия.

Дополнительно могут применяться специфические трансформации, такие как сдвиги, обрезка или изменение геометрии изображения.

Применение методов аугментации позволяет модели обучаться получать на более разнообразных данных, что способствует повышению устойчивости модели в различных условиях. Кроме того, аугментация позволяет уменьшить влияние несбалансированности классов и улучшить обобщающую способность моделей.

Модель YOLO (You Only Look Once) представляет собой популярную архитектуру для обнаружения объектов на изображениях. Основной идеей является выполнение обнаружения объектов и их классификации в одной сети, что делает ее быстрой и эффективной [kirillov2023segment].

Порядок работы модели YOLO начинается с входного изображения, которое подается на вход нейронной сети. Затем изображение проходит через сверточные слои, которые извлекают из изображения признаки на различных уровнях абстракции. Далее, полученные признаки проходят через сверточные слои, которые прогнозируют ограничивающие рамки (boxes) для объектов и вероятности их принадлежности к различным классам. Эти сверточные слои выполняют прогнозы на основе якорей (anchors), которые представляют разные размеры и соотношения сторон ограничивающих рамок. После этого выполняется пост-обработка, включающая подавление неоднородных предсказаний (non-maximum suppression), чтобы получить итоговые прогнозы объектов. Этот шаг удаляет дубликаты и прогнозирует объекты с наибольшей уверенностью (confidence).

Результатом работы модели YOLO является набор ограничивающих рамок с идентификацией классов объектов, найденных на изображении и оценками уверенности. Модель обеспечивает возможность работы в масштабе времени близком к реальному.

Для любых матриц A и B выполняется:

$$\text{rang}(AB) \leq \min(\text{rang}(A), \text{rang}(B)) \quad (2.11)$$

Низкоранговый адаптер Lora работает путем сжатия параметров модели с использованием низкоранговых матриц. Основная идея заключается в том, чтобы аппроксимировать исходные параметры модели с помощью матриц меньшего ранга. Это позволяет снизить объем памяти, необходимый для хранения параметров и ускорить вычисления.

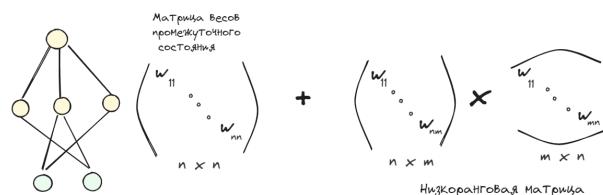


Рис. 2.10: Модель адаптера Lora [hu2021lora]

Наиболее популярным вариантом адаптера является Lora [hu2021lora]:

$$\hat{W} = W + AB, \quad (2.12)$$

где AB — низкоранговая матрица, полученная произведением матриц A размерности $m \times r$ и B размерности $r \times n$, где r — это ранг аппроксимации.

2.2 Генеративные подходы

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

1. порождающие грамматики [chomsky2002syntactic];
2. графические вероятностные модели [pearl1988probabilistic];
3. состязательные порождающие модели [goodfellow2020generative];
4. диффузионные порождающие модели [song2020score].

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 $p(y|x)$ 2.12.

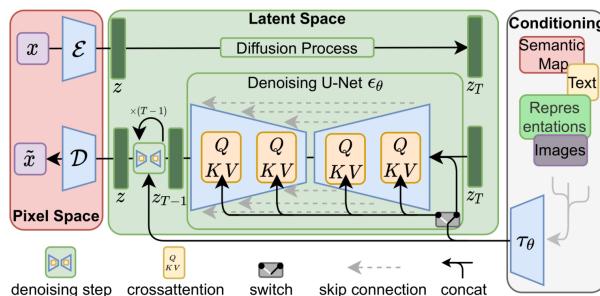


Рис. 2.11: Архитектура современной модели Stable Diffusion.

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 упорядочиванию 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 $f(1) = 0$.

$$D_f \pi \| \rho = E_{\rho(x)} f \left(\frac{\pi(x)}{\rho(x)} \right)$$

Семейство f -дивергенций включает функции

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

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:

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

Перепишем в виде KL-дивергенции:

$$\begin{aligned}
 & D_{\text{KL}}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p_{\theta}(\mathbf{z}|\mathbf{x})) \\
 &= \int q_{\phi}(\mathbf{z}|\mathbf{x}) \log \frac{q_{\phi}(\mathbf{z}|\mathbf{x})}{p_{\theta}(\mathbf{z}|\mathbf{x})} d\mathbf{z} \\
 &= \int q_{\phi}(\mathbf{z}|\mathbf{x}) \log \frac{q_{\phi}(\mathbf{z}|\mathbf{x})p_{\theta}(\mathbf{x})}{p_{\theta}(\mathbf{z}, \mathbf{x})} d\mathbf{z} \\
 &= \int q_{\phi}(\mathbf{z}|\mathbf{x}) \left(\log p_{\theta}(\mathbf{x}) + \log \frac{q_{\phi}(\mathbf{z}|\mathbf{x})}{p_{\theta}(\mathbf{z}, \mathbf{x})} \right) d\mathbf{z} \\
 &= \log p_{\theta}(\mathbf{x}) + \int q_{\phi}(\mathbf{z}|\mathbf{x}) \log \frac{q_{\phi}(\mathbf{z}|\mathbf{x})}{p_{\theta}(\mathbf{z}, \mathbf{x})} d\mathbf{z} \\
 &= \log p_{\theta}(\mathbf{x}) + \int q_{\phi}(\mathbf{z}|\mathbf{x}) \log \frac{q_{\phi}(\mathbf{z}|\mathbf{x})}{p_{\theta}(\mathbf{x}|\mathbf{z})p_{\theta}(\mathbf{z})} d\mathbf{z} \\
 &= \log p_{\theta}(\mathbf{x}) + \mathbb{E}_{\mathbf{z} \sim q_{\phi}(\mathbf{z}|\mathbf{x})} [\log \frac{q_{\phi}(\mathbf{z}|\mathbf{x})}{p_{\theta}(\mathbf{z})} - \log p_{\theta}(\mathbf{x}|\mathbf{z})] \\
 &= \log p_{\theta}(\mathbf{x}) + D_{\text{KL}}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p_{\theta}(\mathbf{z})) - \mathbb{E}_{\mathbf{z} \sim q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z})
 \end{aligned} \tag{2.14}$$

Следовательно:

$$\log p_{\theta}(\mathbf{x}) - D_{\text{KL}}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p_{\theta}(\mathbf{z}|\mathbf{x})) = \mathbb{E}_{\mathbf{z} \sim q_{\phi}(\mathbf{z}|\mathbf{x})} \log p_{\theta}(\mathbf{x}|\mathbf{z}) - D_{\text{KL}}(q_{\phi}(\mathbf{z}|\mathbf{x}) \| p_{\theta}(\mathbf{z})) \tag{2.15}$$

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:

$$Q(\theta | \theta^{(t)}) = \mathbb{E}_{\mathbf{Z} \sim p(\mathbf{Z} | X, \theta^{(t)})} [\log p(\mathbf{X}, \mathbf{Z} | \theta)] \tag{2.16}$$

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

$$\theta^{(t+1)} = \arg \max_{\theta} Q(\theta | \theta^{(t)}) \tag{2.17}$$

Модели потоков

Такой класс моделей

Вариационные кодировщики класс вероятностных моделей, использующие про-

межуточное скрытое представление.

Предшественником вариационного кодировщика был автокодировщик, состоящий из двух частей:

1. модель компрессии изначального изображения g_ϕ ;
2. модель декомпрессии изначального изображения f_θ .

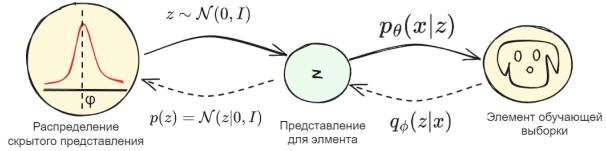


Рис. 2.12: Вариационный кодировщик задает ошибку кодирования в виде параметрического нормального распределения.

Функция оптимизации такой модели записывается как среднеквадратичная разность между входом и выходом модели:

$$L_{\text{AE}}(\theta, \phi) = \frac{1}{n} \sum_{i=1}^n (\mathbf{x}^{(i)} - f_\theta(g_\phi(\mathbf{x}^{(i)})))^2 \quad (2.18)$$

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:

$$-L_{\text{VAE}} = \log p_\theta(\mathbf{x}) - D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{z}|\mathbf{x})) \leq \log p_\theta(\mathbf{x}) \quad (2.19)$$

В работе [kingma2013auto] предложен численно эффективный учет ошибки, заданной нормальным распределением. Обучение происходит путем автоматического дифференцирования по параметрам распределения:

$$\begin{aligned} \mathbf{z} &\sim q_\phi(\mathbf{z}|\mathbf{x}^{(i)}) = \mathcal{N}(\mathbf{z}; \boldsymbol{\mu}^{(i)}, \boldsymbol{\sigma}^{2(i)} \mathbf{I}) \\ \mathbf{z} &= \boldsymbol{\mu} + \boldsymbol{\sigma} \boldsymbol{\epsilon}, \end{aligned} \quad (2.20)$$

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

$$\max_{\phi, \theta} \mathbb{E}_{\mathbf{x} \sim \mathcal{D}} [\mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z})], \quad (2.21)$$

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:

$$\begin{aligned}\mathcal{L}(\theta, \phi, \beta) &= \mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - \beta(D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})) - \delta) \\ &= \mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - \beta D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z})) + \beta\delta \\ &\geq \mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})} \log p_\theta(\mathbf{x}|\mathbf{z}) - \beta D_{\text{KL}}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{z}))\end{aligned}\quad (2.22)$$

Итоговая оптимизационная функция с гиперпараметром β записывается как:

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

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

$$\log p_K(z_K) = \log p_0(z_0) - \sum_{i=1}^K \log \left| \det \frac{df_i(z_{i-1})}{dz_{i-1}} \right| \quad (2.23)$$

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

$$\begin{aligned}\mathbf{z} &\sim \pi(\mathbf{z}), \mathbf{x} = f(\mathbf{z}), \mathbf{z} = f^{-1}(\mathbf{x}) \\ p(\mathbf{x}) &= \pi(\mathbf{z}) \left| \det \frac{d\mathbf{z}}{d\mathbf{x}} \right| = \pi(f^{-1}(\mathbf{x})) \left| \det \frac{df^{-1}(\mathbf{x})}{d\mathbf{x}} \right|,\end{aligned}\quad (2.24)$$

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

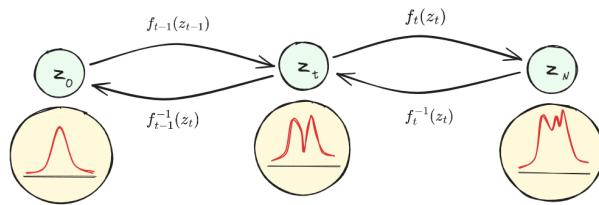


Рис. 2.13: Модели потока используют последовательные обратимые преобразования для приведения исследуемого распределения к известному.

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 функций с заданными требованиями, как правило, используются перестановки слагаемых и матричные преобразования.

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

В методе RealNVP в качестве преобразования предлагается афинное преобразование последних $D - d$ компонент вектора генерации:

$$\begin{aligned} \mathbf{y}_{1:d} &= \mathbf{x}_{1:d} \\ \mathbf{y}_{d+1:D} &= \mathbf{x}_{d+1:D} \odot \exp(s(\mathbf{x}_{1:d})) + t(\mathbf{x}_{1:d}) \end{aligned} \quad (2.25)$$

Детерминант полученного выражения запишется как:

$$\mathbf{J} = \begin{bmatrix} \mathbb{I}_d & \mathbf{0}_{d \times (D-d)} \\ \frac{\partial \mathbf{y}_{d+1:D}}{\partial \mathbf{x}_{1:d}} & \text{diag}(\exp(s(\mathbf{x}_{1:d}))) \end{bmatrix} \quad (2.26)$$

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

$$\begin{cases} \mathbf{y}_{1:d} = \mathbf{x}_{1:d} \\ \mathbf{y}_{d+1:D} = \mathbf{x}_{d+1:D} + m(\mathbf{x}_{1:d}) \end{cases} \Leftrightarrow \begin{cases} \mathbf{x}_{1:d} = \mathbf{y}_{1:d} \\ \mathbf{x}_{d+1:D} = \mathbf{y}_{d+1:D} - m(\mathbf{y}_{1:d}) \end{cases} \quad (2.27)$$

Логарифм детерминанта преобразования в этом случае запишется как:

$$\log \left| \det \frac{\partial \text{conv2d}(\mathbf{h}; \mathbf{W})}{\partial \mathbf{h}} \right| = \log(|\det \mathbf{W}|^{h \cdot w}) = h \cdot w \cdot \log |\det \mathbf{W}| \quad (2.28)$$

2.2.1 Энергетические модели

Энергетические подходы стали популярны с публикацией прорывных работ по диффузионным порождающим моделям [**song2020score**]. 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 :

$$p(x) \sim \frac{\exp(-E(x))}{Z}, \quad (2.29)$$

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 [**lecun2006tutorial**].

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

вероятного распределения p .

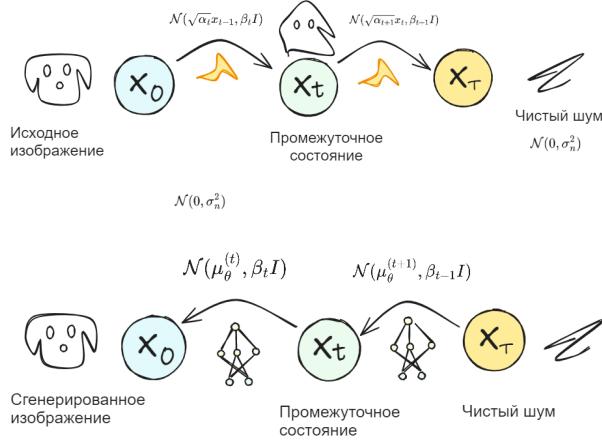


Рис. 2.14: Прямой процесс зашумления и обратный процесс коррекции ошибки [stablediffusion].

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:

$$x_t = \sqrt{1 - \beta_t} x_{t-1} + \sqrt{\beta_t} z_t, \quad (2.30)$$

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

$$q(x_T) = q(x_0)q(x_1|x_0)\dots q(x_T|x_{T-1}) = q(x_0)\mathcal{N}(x_1|\sqrt{1 - \beta_1}x_0, \beta_1 I)\dots\mathcal{N}(x_T|\sqrt{1 - \beta_T}x_0, \beta_T I) \quad (2.31)$$

В компактной форме:

$$\ln q(x_T) = \ln q(x_0) - \sum_{t=1}^T \frac{1}{2\beta_t} \|x_t - \sqrt{1 - \beta_t} x_{t-1}\|^2 + C \quad (2.32)$$

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

$$p_\theta(x_T) = \mathcal{N}(x_T|0, I); p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}|\mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \quad (2.33)$$

Для введения параметрической модели используют нижнюю вариационную границу ELBO:

$$\mathbb{E}_{x_0 \sim q} \ln p_\theta(x_0) \geq \mathbb{E} \ln p_\theta(x_T) - \ln() \quad (2.34)$$

Тогда функция ошибки запишется как:

$$\begin{aligned} L(\theta) = - \sum_{t=1}^T \mathbb{E}_{x_{t-1}, x_t \sim q} [-\ln p_\theta(x_{t-1|x_t})] + \\ + \mathbb{E}_{x_0 \sim q} \mathbf{D}_{KL}(q(x_T|x_0) \| p_\theta(x_T)) \end{aligned} \quad (2.35)$$

На практике функция ошибки упрощается до предсказания шума:

$$\mathbb{E}_{x_0 \sim q; z \sim \mathcal{N}(0, I)} [\|\varepsilon_\theta(x_t, t) - z\|^2] \quad (2.36)$$

2.3 Обработка естественного языка

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

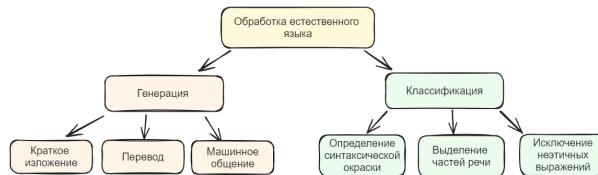


Рис. 2.15: Таксономия современных подходов обработки естественного языка.

Формальные языки широко используются в математике, логике, лингвистике и компьютерных науках. В программировании, например, формальные языки включают языки программирования и описания данных, где синтаксис строго определён для обеспечения корректности и детерминированности выполнения программ. В областях обработки естественного языка приобрела популярность аналитическая форма механизма внимания [vaswani2017attention], сделавшая возможным создание больших лингвистических моделей [radford2019language].

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" [chomsky2002syntactic]. 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: Формальные составляющие языка является совокупностью:

- алфавита Σ — конечного множества символов, из которых строятся строки языка;

- *строки* — последовательности символов из алфавита, которые принадлежат языку;
- *грамматики* — набора правил, которые определяют, какие строки являются допустимыми в языке.

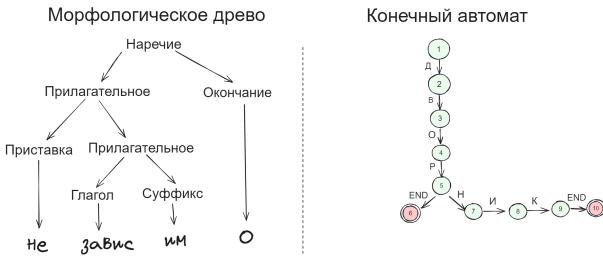


Рис. 2.16: Морфологическое дерево задает совместную иерархическую структуру слова в виде конечного автомата.

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

Для обработки естественного языка методами формальных языков из слов выделяются стемы и словоизменительные парадигмы, являющиеся образцами для склонения или спряжения. Такие структуры создаются специалистами лингвистами и позволяют путем склонения, спряжения и добавления приставки получать необходимые формы слов.

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

Определение: **Морфологическим синтезом** называется функция f , формирующая слова w из леммы l и морфологических характеристик t . Примерами морфологических характеристик являются число, род, 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 коренной узел r , заключает структуру предложения. Тогда дерево разбора T для строки $w = w_1w_2 \dots w_n$ 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 , проявляющие семантические и синтаксические при операциях сложения и взятия косинусного расстояния.

Практически востребованной оказалась дистрибутивная гипотеза [Schutze], легшая в основу алгоритма Word2Vec (Вектор для Слова)[NIPS2013_9aa42b31]. Известны и другие подходы GloVe, FastText, улучшающие подход:

$$\cos(\theta) = \frac{w_1 \cdot w_2}{\|w_1\| \|w_2\|} \quad (2.37)$$

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

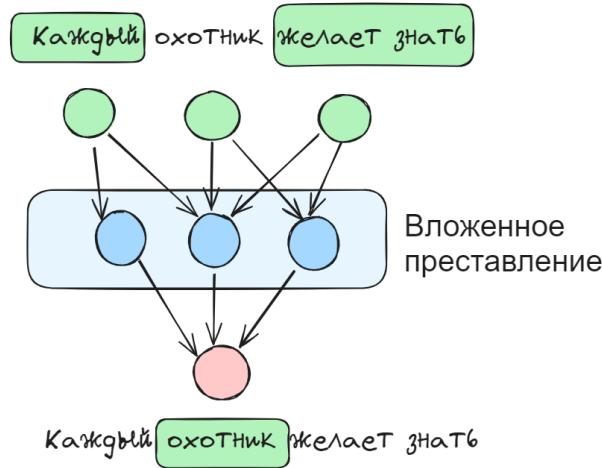


Рис. 2.17: Векторное вложение позволяет выполнять семантические операции

для алгоритмов машинного обучения. Это позволяет моделям эффективно сопоставлять зависимости между словами и генерировать семантически богатые и лингвистически осмыслиенные тексты. Появление трансформерной архитектуры нейросе-

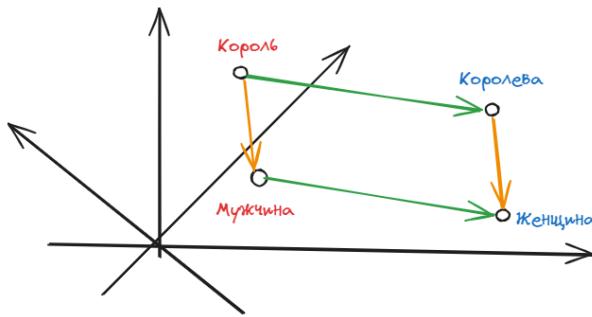


Рис. 2.18: Векторное вложение позволяет выполнять семантические операции.

тей позволило улучшить результаты семантического представления для работы с предложениями и параграфами [devlin2018bert]. Ключевым преимуществом модели является работа с максимированным представлением языка, что позволяет изменить постановку обучения до предсказания промежуточного элемента.

$$P(w_i | w_1, \dots, w_{i-1}, w_{i+1}, \dots, w_n) \quad (2.38)$$

Задача

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. При этом требуется задание необходимой текстовой структуры. Данная техника позволила исследователям [radford2021learning] обучить общую модель

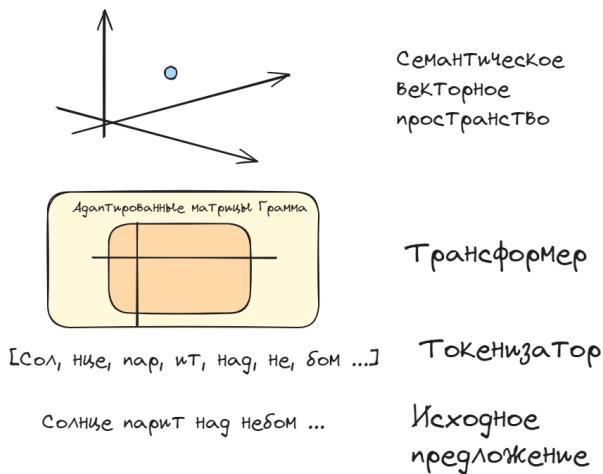


Рис. 2.19: Векторное позволяет выполнять семантические операции

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 языка 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.

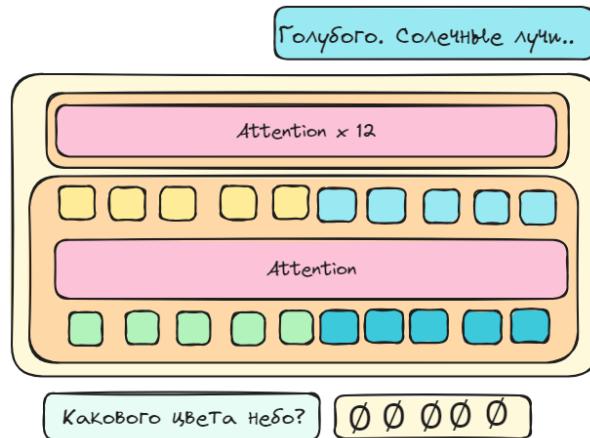


Рис. 2.20: GPT обучается как авторегрессионная модель на ко

Обучение языковых моделей для задач ассистирования разделяется на два этапа предобучение (англ. pretrain) и инструкционное дообучение. В ходе первого этапа модель обучается синтаксической структуре языка. На втором этапе модель обучается под руководством эксперта на специализированных корпусах инструкций. Таковыми, например, могут являться выдача структурированных определений, изложение информации в требуемой стилистической форме, поиск ключевых слов в большом по объему тексте.

Предобучение требует значительных вычислительных ресурсов, не всегда доступных в образовательных и научных приложениях. В связи с этим компании, обладаю-

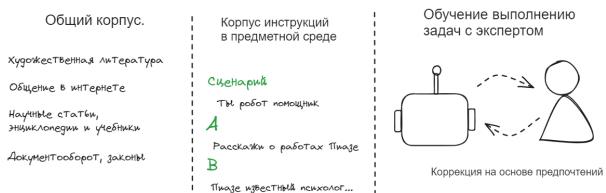


Рис. 2.21: Обучение разбито на три ключевых этапа: подготовка, адаптация на корпусе.

щие достаточным ресурсом, делают свои модели общедоступными [[jiang2023mistral](#)] [[jiang2024mixtral](#)] [[touvron2023llama](#)]

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

- способности к пересказу и переводу;
- пониманию животного и растительного мира;
- социальных правил и законов;
- решение аналитических задач.

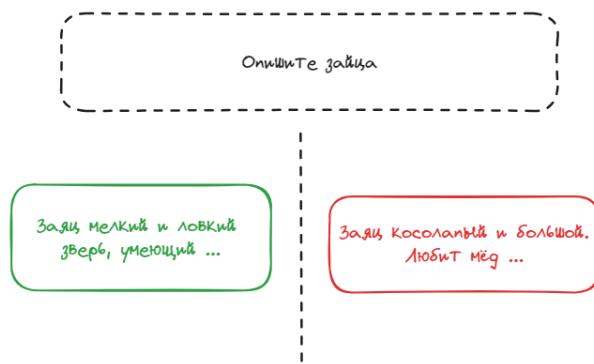


Рис. 2.22: Адаптация к задачам выполняется путем взаимодействия с пользователем и работой с экспертом.

Для сравнения моделей также используется техника попарного сравнения моделей. Таким образом, исходя из предпочтений пользователей, выносится оценка модели.

Большие языковые модели не гарантируют корректное исполнение даже базовых арифметических операций. В обзоре [[zhao2023survey](#)] показано, что подобные проблемы возникают во многих строгих постановках, где соблюдение формы требует интеллектуального участия

- написание исполняемого языком программирования кода;
- переход в математических выражениях.

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

- обращения к информационным системам (RAG - retrieval augmented generation)[lewis2020];
- работа с средой исполнения программного кода [**parisi2022talm**];
- генерация сопровождающей иллюстрации [**rombach2022high**].

Глава 3

Математические методы в педагогике

Педагогика активно развивающееся научное направление, изучающее методы воспитания гармонично развитой личности . Существенно влияние на педагогику оказывает появление направление образовательных технологий,

Глава посвящена изучению педагогики и методов научной оценки качества задачи для обучающегося. Современные требования При создании педагогической задачи важно учитывать не только содержание обучения, но и индивидуальные особенности студентов, их уровень знаний и способности. Педагогическая задача должна быть четко сформулирована, чтобы студенты могли понять, что от них требуется, и чувствовать уверенность в выполнении задания.

Важным аспектом педагогической задачи является её реалистичность и актуальность. Задача должна иметь практическую ценность и быть связанной с реальными жизненными ситуациями или профессиональными задачами. Это поможет стимулировать интерес и мотивацию студентов к изучению материала.

Реализация педагогической задачи может включать использование различных методов обучения и оценки, таких как групповая работа, проектная деятельность, обсуждения, решение проблемных ситуаций и другие. Это позволит стимулировать активное участие студентов в образовательном процессе и способствовать их полноценному развитию.

3.1 Значение образования

Введение изменений в методику преподавания требует обоснования, выраженного в положительном изменении общественного блага.

В литературе популярен аналитический подход к расчету социального блага через функции утилитарности. В основе направления лежит идея, что общественное благо может быть определено путем суммирования утилитарных предпочтений индивидов в обществе. Этот подход связан с концепцией утилитаризма, которая утверждает, что действия должны быть оценены исходя из их способности максимизировать общую полезность или счастье всех членов общества.

Известны чисто аналитические подходы к оценки общественного блага при планировании выборов, проектировании транспортных систем и сфере медицины.

3.1.1 Функция блага

Утилитарность может быть ординальна и абсолютна.

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

Происхождение функции утилитарности связано с развитием философской и экономической мыслью. Одним из основоположников утилитаризма был Джереми Бентам. В XVIII веке он разработал теорию, согласно которой целью действий является максимизация утилиты - суммарной пользы или счастья для всех членов общества.

Определение Равновесие по Нэшу Состояние, в котором ни один игрок не может получить дополнительную выгоду от своих измененных действий, если другие игроки продолжают свои стратегии.

Классическим примером является дилемма заключенного.



Рис. 3.1: Дилемма заключенного

Ординальная и абсолютная функции утилитарности представляют собой два различных подхода к измерению удовлетворения или полезности. В ординальном подходе утилитарное значение альтернатив оценивается лишь относительно других альтернатив, без установления конкретных числовых значений. Это позволяет определить порядок предпочтений между различными альтернативами, но не предоставляет количественной информации о различии в уровне удовлетворения. С другой стороны, в абсолютном подходе утилитарное значение измеряется в конкретных численных единицах, что позволяет проводить количественные сравнения между альтернативами и оценивать уровень удовлетворения более точно.

3.1.2 Механизм

Оптимальное распределение ресурсов в аналитических постановках изучается с помощью дизайна механизмов. Современный подход был предложен и развит Леонидом Гурвичем [hurwicz1960optimality]. В его работе изучается проблема доказательной эффективности организации рабочих процессов.

Механизм позволяет ограничить свободу действий игроков в системе таким образом, что действия агентов в доминантных стратегиях будут соответствовать ожиданиям принципала.

Строгое определение вводится через понятие байесовых игр с неполной информацией, предполагающих наличие нескольких многошаговых игр, направленных на формирование оптимальной стратегии.



Рис. 3.2: Дилемма заключенного

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

Изучение механизма заключается в 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 не позволяет наивно перебрать возможные исходы.

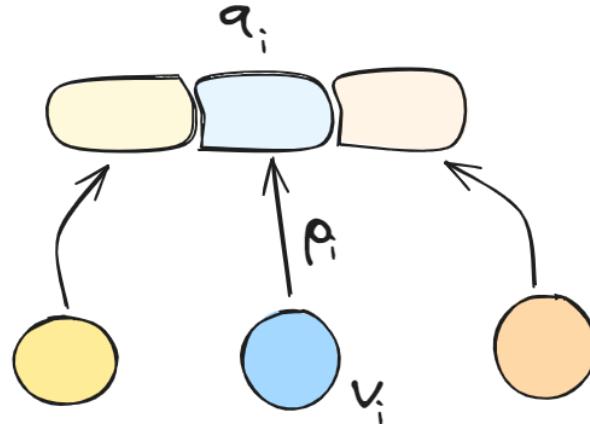


Рис. 3.3: Механизм определяет распределение ресурса x и платежа p в зависимости от ставок b

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

- случайную величину v_i с функцией распределения F . Обозначим полученный квантиль распределения $q(v_i) = 1 - F(v_i)$.
- ставку на лот b_i .
- долю полученного ресурса x_i
- p_i итоговый платеж

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

$$u_i(\mathbf{b}; q_i) = v(q_i) \cdot x_i(\mathbf{b}) \quad (3.1)$$

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

$$\max_{\mathbf{q}, \mathbf{b}} E_{\mathbf{q}_{-i}} [u_i(\mathbf{b}(\mathbf{q}); q_i)] \quad (3.2)$$

В теории [bulow1989simple] аукционов различают платеж . Формально опишем условия:

- $\tilde{x}_i(\mathbf{q}) = x_i(\mathbf{b}(\mathbf{q}))$ фактическое (от англ. ex-post) полученная доля ресурса;

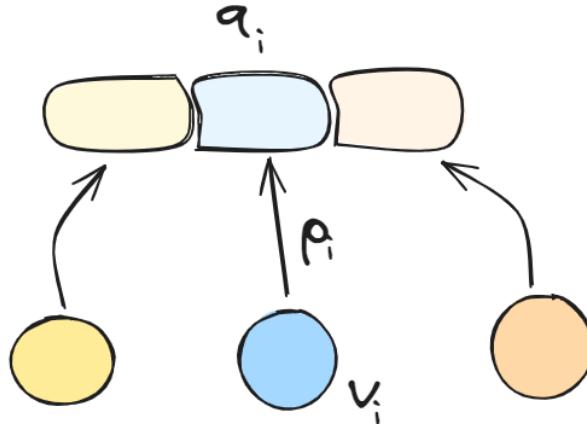


Рис. 3.4: Ресурсное распределение

- $\hat{x}_i(q_i) = \mathbb{E}_{\mathbf{q}} [x_i(\tilde{\mathbf{q}}, q_i)]$ текущее(от англ. interim) ресурсное ожидание;
- \hat{p} ожидаемый платеж.

Лемма Набор имеет Байес-Нэшево равновесие при условии

- $\hat{x}_i(q_i)$ монотонно убывает по q_i
- $\hat{p}_i(q_i) = v(q_i) \cdot \hat{x}_i(q_i) + \int_{q_i}^1 \hat{x}_i \cdot v'(z) dz$

Заданные условия достаточны для достижения уникально равновесия \mathbf{b} . Доказательство приведено в [[myerson1981optimal](#)].

Одним из примером механизма являются рейтинг системы, широко распространенные в спортивных интеллектуальных соревнованиях.

Рейтинг-система – это модель, которая ранжирует n участников в единый линейный порядок $i_1 \succ i_2 \succ \dots \succ i_n$ поданным сравнений небольших подмножеств этих игроков.

В этом случае рейтинг можно задать как оператор ζ принимающий на вход потенциалы, задающие силу объекта.

$$p(x_i \succ x_j) = \zeta(\phi(x_i), \phi(x_j)) \quad (3.3)$$

Согласно модели рейтинга Эло сила игрока задается случайной величиной ξ . Экспоненциальный вид графика связан с предположением о том, что в стратегических играх существенное различие в навыке почти наверняка гарантирует победу. Заметим, что такое предположение *Рейтинг* задается матожиданием силы $\mathcal{E}\xi$. Согласно модели Эло сила игрока задается нормально, причем дисперсия σ фиксирована для всех игроков. Тогда сила игры согласно предложению определяется как:

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp^{-\frac{1}{2\sigma^2}(x-s)^2} \quad (3.4)$$

Таким образом, рейтинг является латентной переменной. В литературе также популярна модель Брэдли-Терри, задающая вероятность победы зависит как:

$$P(\theta) \quad (3.5)$$

Заметим, что подмена $\gamma \exp(-\theta/\beta^2)$ позволяет отождествить подходы. В шахматной практике волатильность считается определенной и имеет стандартное отклонение равное 20:

$$\frac{1}{1 + 10^{\frac{R_B - R_A}{400}}} \quad (3.6)$$

3.1.3 Систем наград

Награды могут иметь разное происхождение, предназначение и формы. Например, в российском школьном образовании отличная учеба награждается медалями, а физкультурно-спортивные достижения сопровождаются знаками отличия. Награды поощряют вовлеченную учебу и совокупно позволяют выделять талантливых учащихся.

Известные компании, предоставляющие инструменты разработки, как Github¹ и Google Cloud², также используют систему знаков - бейджей для поощрения пользователей ресурса 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 Overflow](<https://stackoverflow.com/>), на которой публикуются вопросы преимущественно технической тематики. Согласно ее правилам бейдж присваивается в случае наличия лучшего ответа на популярный вопрос. Такая награда поощряет положительную конкуренцию, поощряет к созданию конструктивных и этичных ответов [yanovsky2021one].



Рис. 3.5: Упорядоченная система наград

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

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

$$S(t_i) = \frac{b_i}{v_i} \quad (3.7)$$

¹<https://cloud.google.com/>

²<https://cloud.google.com/>

Задача механизма обеспечить максимизацию суммарного вклада участников:

$$\max_{P(m)} \sum_{i=1}^n b_i \quad (3.8)$$

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 легок в использовании и понимании для агентов. Предварительное задание уровня награды позволяет планировать путь достижения деятельности, сосредотачиваясь на индивидуальном развитии. Тем не менее такой подход не поддерживает соревновательный дух, потому подвержен не всегда желаемой кооперации среди агентов. Альтернативой является задание рейтинга согласно порядку вклада

Определение Механизм таблицы лидеров называется механизм распределения бейджей $j \in \{0, \dots, m\}$, в порядке убывания по вкладу b_i . Число бейджей и 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 недостаточно для задач, требующих постоянной отдачи от участников. Также слишком большое число бейджей неудобно для разработки на практике.

Работа [Easley2013] предлагает объединение двух подходов для разрешения перечисленных проблем.

Определение Механизм 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 аналитического изучения постановки.

Теорема 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 Бернштейн 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 $\sum b_i$.

3.2 Структура методического материала

В секции рассматриваются современные теории образования. Рассматриваются подходы Различают дидактический и основанный на опыте методы преподавания.

Также в секции будут разобраны подходы к аналитической оценке системы рейтинга

Объект изменения	Название практики	Резу
Учитель	Курсы повышения специализации	2
Учитель	Обеспечение	1
Ученик	Пороговый балл прохождения курса	1
Учитель	Предоставление частичного решения, подсказки	1
Учитель, Ученик	Вовлеченность в занятие	1
Ученик	Задание строго времени выполнения задания	1
Ученик	Упор на навыки письма и чтения	1
Домашнее окружение	Обучение с родителями	0
Учитель	Балловая оценка домашнего задания	0
Учитель	Поддержание духа класса	0
Ученик	Отбор учащихся на основании интеллектуальных способностей	0
Домашнее окружение	Изменение домашней обстановки	0
hline		

Таблица 3.1: Влияние образовательных практик согласно [bloom1984]

3.2.1 Методы образования

Дидактический педагогический метод наиболее распространенный метод обучения. Такие методы задают практики организации учебно-познавательной деятельности учащихся в рамках педагогической системы. Ключевым инструментом такого подхода является количественная оценка успеваемости выполняется на основании контрольных работ. Решение об оценки может выноситься как учителем, так и автоматически с использованием приложения. Оценка выполняет множество задач:

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

Известный психолог педагог Бенжамин Блум в работе [bloom1984] изучает проблемы массового образования. Согласно его исследованию обучающиеся по программам индивидуального образования имели результаты статистически лучше посещающих общеобразовательные учреждения. Автор, опираясь на статистический анализ практики своих ассистентов предлагает практики для общего образования, демонстрировавшие наибольшую эффективность 3.1

Автор предлагает дидактические практики, ставя акцент при обучении на контроле за успеваемостью учащихся. В таком подходе регулярно проводятся контрольные занятия, представляющие количественную оценку знаний и навыков. Для этого занятия сопровождаются подготовленной средой: учебником, рабочей тетрадью и лабораторной установкой.

Противопоставленным дидактическому методу обучения является проблемно-ориентированный подход. Эмпирический подход ставит акцентируется на активном участии обучаю-

щегося в исследовании и поиске знаний через решение проблем и практическую деятельность. 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. [РэСГСТРУРЧР «Рў2006РўРхРъР«РўСКРх] 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 и акцент на командной работе.

3.2.2 Психологические подходы к описанию интеллектуального развития

Одной из наиболее разработанных научных психологических теорий интеллектуального развития является теория когнитивного развития, разработанная Жаном Пиажетом [piaget1952origins]. Согласно доктору когнитивное развитие происходит в ходе стадий, соответствующих развитию интеллектуальных способностей. Таковыми являются все виды мыслительных процессов, таких как восприятие, память, формирование понятий, решение задач, воображение и логика.

1. сенсомоторного интеллекта
2. дооперациональных представлений
3. период конкретных операций
4. конкретное и абстрактное логическое мышление.

Этапы развития детской психики по Жану Пиаже, предложившего разделение процесса когнитивного развития ребенка на стадии, в зависимости от его возраста

Доктор обосновывает свою теорию этическими экспериментами, легко доступными к воспроизведению педагогам.

Лев Семёнович Выготский оспаривает этот подход, приводя в пример детей, которые в ходе занятий со взрослыми продемонстрировали высокий уровень освоения шахмат, плавания и прочих дисциплин 3.7. Выполненные профессором экспериментами стали основой направления культурно-исторической психологии, получившее развитие в мировой психологии. Согласно Выготскому образование имеет социальный и целенаправленный характер, направленное на развитие *высших психических функций* - сложных самоорганизующиеся процессы в деятельности человека, опосредованных знаками по своему строению и произвольные по способу своего осуществления. Таковыми навыками Выготский определил мышление, речь, произвольное внимание, воля, воображение и др.

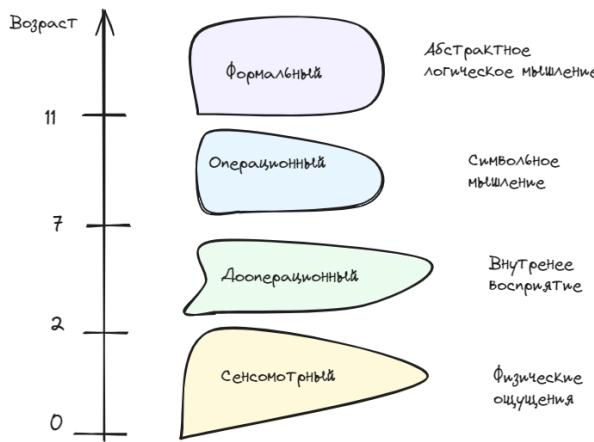


Рис. 3.6: Этапы развития детской психики по Жану Пиаже [piaget1952origins]

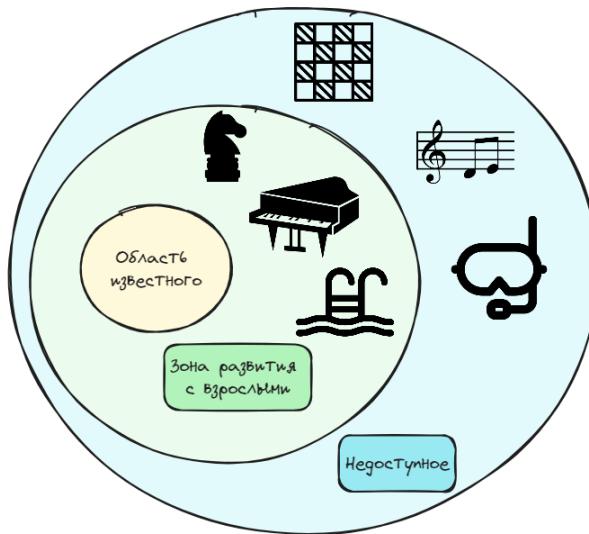


Рис. 3.7: Иллюстрация зоны ближайшего развития

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

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. Согласно профессору механизмом рефлексии является *интериоризация*. Процесс заключается в переходе внешней речи во внутренний способ мышления. Отметим, что процесс возможен когда ребёнок способен уже в уме планировать свою деятельность.

Рефлексия способствует развитию метакогнитивных навыков

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

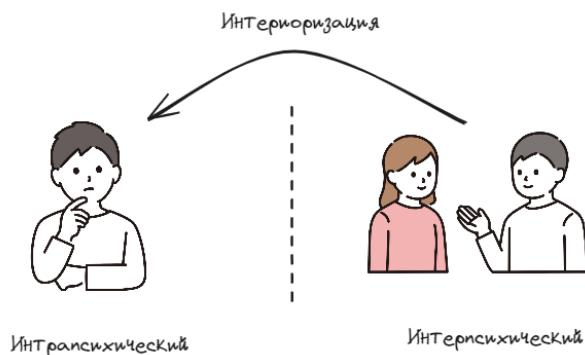


Рис. 3.8: Рефлексия по Выготскому
[РУСКРУП «СТБСФРЭРЧР€2014РёСКСЇРЊРхР,,РЧРХ]

Таким навыки позволяют справляться с практическими заданиями, обучать и осваивать неизвестные областях знаний, выявить собственные пробелы в знаниях и развить критическое мышление.

Определение Поток — это состояние сознания, характеризующееся полной вовлечённостью и концентрацией на текущей деятельности, сопровождающееся ощущением контроля, потерей чувства времени и высокой внутренней мотивацией.

Теория потока активно используется в играх [chen2007flow] и программах непрерывного обучения [jarvis2009routledge] для задания оптимальной сложности образовательной программы.

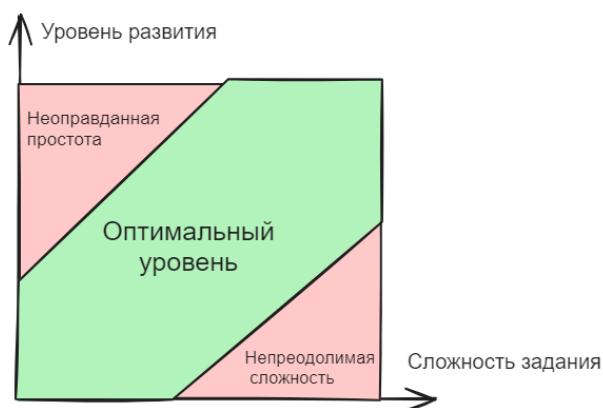


Рис. 3.9: Рефлексия по Выготскому
[РУСКРУП «СТБСФРЭРЧР€2014РёСКСЇРЊРхР,,РЧРХ]

Характеристиками потока согласно [csikszentmihalyi2005flow] 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 сложно, чтобы бросать вызов, но при этом соответствовать уровню навыков человека, чтобы избегать 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(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;
- универсальная грамматика, 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 Ричарда Номски, использовавшего статистический аппарат синтаксических деревьев для анализа массивного корпуса естественного языка для [everaert2015structures][montague1970uni]

3.3 Методическая задача

Задача является ключевым инструментом в обучении. Её цель состоит в том, чтобы обеспечить студентам определённые образовательные возможности и помочь им развить необходимые знания, умения и навыки.

Для этого педагогическая задача должна:

- предоставлять возможность для развития критического мышления и применения знаний на практике.
- быть структурированной и обеспечивать возможность оценки выполнения студентами поставленной задачи.
- иметь ясные и объективные критерии оценки, чтобы обеспечить справедливую оценку достижения учебных целей.

3.3.1 Тестовые системы

В разделе описаны основные направление электронного образования крайне популярно и имеет важную экономическую роль.

- классическое тестирование
- классификационное тестирование [CCT]
- адаптивное тестирование [CAT] В этом случае задания подбираются динамически во время тестирования

Платформы обучения языкам(ИСТ)

- Revita [[katinskaia2018revita](#)]

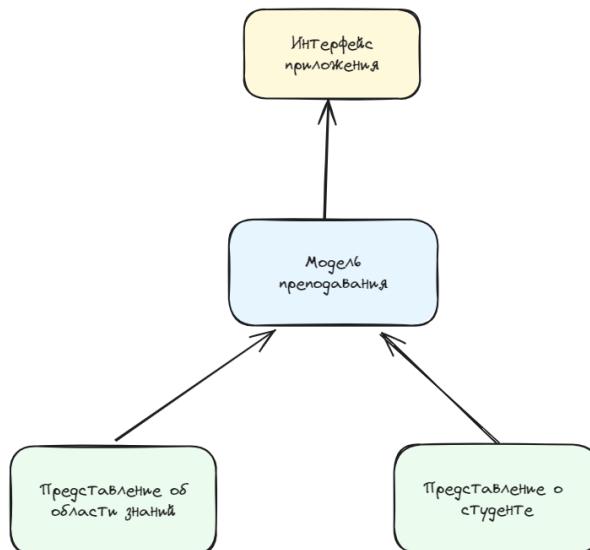


Рис. 3.10: Схема организации обучения

Определение Психометрия- дисциплина, изучающая количественные способы оценки знания.

Для этого используются статистические методы оценки знания, учитывающий случайность в измерениях. Предполагается, что знание не подлежит явному измерению, а лишь косвенному путем проведения тестирования или анализа деятельности обучающегося.

Для построения теории вводятся скрытые от наблюдения переменные называемыми латентными θ . В области образования психометрические исследования наиболее активно выполняются для тестовых заданий. Основными подходами к оценке знаний, исходя из результатов, на текущий момент является классическая и (item response) теория.

Классическая теория предполагает, что результат тестирования задан случайной величиной. Её вид вид как правило предполагается нормальным:

$$s \sim N(\theta, \sigma^2) \quad (3.9)$$

,где s задает экзаменационный результат обучающегося, параметр θ - истинный уровень знания, σ^2 - задает волатильность измерений. распределения с дисперсией, определяемой

Как правило тесты подбирают таким образом, чтобы ошибка метода на всем промежутке результатов была минимальна

$$\int_0^1 \sigma^2(\theta) \rightarrow \min$$

Существенным недостатком такой системы является предположение о равной сложности задач в контрольной работе.

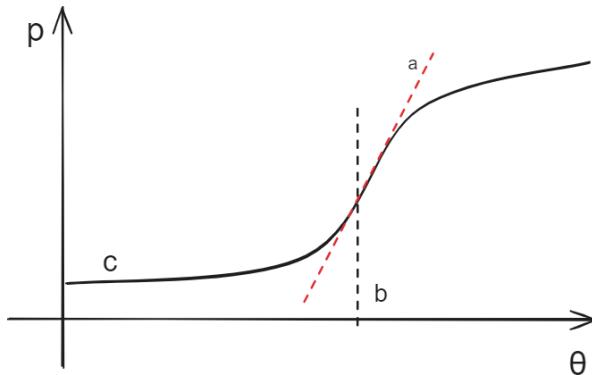


Рис. 3.11: Матрица исходов модели Байесовской оценки на шаге t

Система тестирования IRT была предложена институтом в 1950 году. Она активно используется в международных экзаменах языка и делового знания GMAT и TOEFL. В отличие от классической теории также учитывают текущий уровень знаний, что позволяет составлять набор заданий индивидуально. Наиболее известным результатом системы является 3-х параметрическая логистическая модель 3.11, учитывающая сложность задачи, вероятность угадать и волатильность оценки Одним из ключевых [lord1956measurement]:

$$p_i(\theta) = c_i + \frac{1 - c_i}{1 + e^{-a_i(\theta - b_i)}}, \quad (3.10)$$

где

- b_i - сложность задания
- a_i - характерный масштаб
- c вероятность угадать решение

Альтернативным путем является подход байесовской оценки знания, описанный в работе [corbett1994knowledge]. Модель учитывает вероятность ошибки и вероятность ошибиться при наличии знания 3.12:

- $P(L_0)$ начальные знания в предмете
- $P(S) = P(x = 0|L_t = 1)$ вероятность просчета при наличии знаний



Рис. 3.12: Эволюция представлений о знаниях учащегося

- $P(G) = P(x = 1|L_t = 1)$ вероятность угадать при отсутствии знаний

Обновление представлений выполняется через Байесов подход согласно правилам:

$$P(L_t|obs_t = 1) = \frac{P(L_t)(1 - P(S))}{P(L_t)(1 - P(S)) + (1 - P(L_t))P(G)} \quad (3.11)$$

$$P(L_t|obs_t = 0) = \frac{P(L_t)P(S)}{P(L_t)P(S) + (1 - P(L_t))(1 - P(G))}$$

Отметим, что полученный вывод предполагает, что

- вероятность забыть знание равна нулю $P(L_{t+1} = 0|L_t = 1) = 0$
- $P(L_{t+1}) = P(L_t|obs_t) + (1 - P(L_t|obs_t)) P(T)$

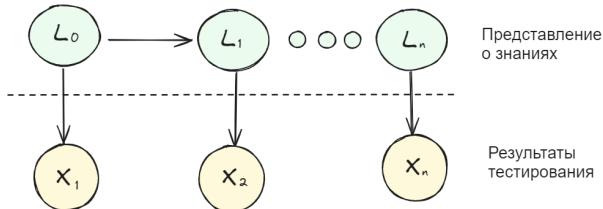


Рис. 3.13: Матрица исходов модели Байесовской оценки на шаге t

Таким образом тест можно представить в виде марковской цепи обновления представлений о знаниях учащегося 3.13.

Адаптация для случая IRT ?? позволяет учесть влияние сложности задания [bulut2023intro]

$$P(Y_{ij} = 1|\theta_J, a_i, b_i, c_i) = \quad (3.12)$$

Глава 4

Описание работы

В главе описан ход исполнения задач, выполненных в ходе диссертационной работы. Ключевым достижением работы является адаптация алгоритма Роббинса-Монро под отклик, заданный логистической функции. В финальной секции главы описана теорема, задающая в аналитическом виде коэффициенты численной схемы для оптимального спуска к заданному уровню попыток.

Практическим итогом работы стало веб-приложение, организующее персональное обучение шахматам с поддержкой интеллектуальным ассистентом. В разработанной системе большая языковая модель отвечает за общение 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 сложности согласно текущим пожеланиям. Приложение также имеет многоуровневую систему достижений, поощряющих нестандартную игру и настойчивость в обучении.

Также приведено описание сбора данных образовательной направленности из открытых источников с использованием современных систем оптического распознавания.

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 [**tinn2023fine**]. Тем не менее открытые корпусы русского языка [**hung2022multi2woz**] [**dmitrieva2023automatic**] [**ivanov2023new**] почти не содержат образовательной тематики. Представленный в секции корпус призван разрешить проблему недостатки данных в направлении образования. Результат моделирования представлены на открытых ресурсах¹ с обязательным указанием источником дан-

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

ных. Полный список источников приведен в апPENDиксе работы ??.

4.1.1 Разметка

Сбор данных для научных проектов из потребности воспроизводимости эксперимента должен проходить из открытых источников. Таковыми, например, могут быть книги из открытых источников, задачи из олимпиад и предметные вебсайты. Также сбор можно проводить из уже подготовленных научным сообществом корпусов, задавая экспертные критерии отбора и выполняя перевод с иностранного языка.

Полученные данные имеют мощность порядка миллиона слов 4.1



Рис. 4.1: Распределение данных по источникам

Сбор данных, представленных в виде документов, не имеющих подготовленного текстового слоя осуществлялся при помощи технологий оптического распознавания символов. В описании секции также частично включен методы, разработанные в рамках данного исследования.

Также в состав открытого тренировочного набора входит более тысячи аннотированных изображений

Международные сайты исследователей содержат большое число подготовленных структурированных данных. К сожалению, большинство из них представлены на китайском и английском языках. Для их адаптации предложен подход, использующий большие языковые модели как средство перевода на русский язык. В качестве открытого решения был выбран языковой ассистент llama3, показывающий

Перевод 7500 задач выполнялся в течении 12 часов. Полученные результаты приведены в предметном репозитории ²

Полный список использованной литературы приведен в апPENDиксе работы ??

Для адаптации корпуса задач также были подготовлены

4.1.2 Распознание текста

Системы оптического распознавания позволяют включать в корпусы текста для обработки цифровые документы с неподготовленным текстовым слоем. Особенностью

²https://huggingface.co/datasets/NMashalov/olympiad_task_translation

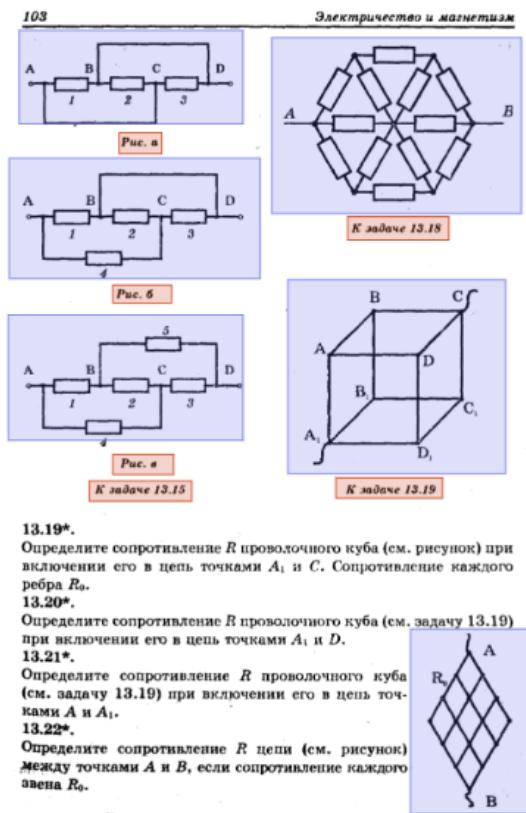


Рис. 4.2: Пример аннотированной иллюстрации из книги Генденштейн, Кирик, Гельфгат: 1001 задача по физике

данных в образовании является их представленность преимущественно в цифровых документах, различающихся по верстке и организации текста. Для восприятия материала также существенны обильно встречающиеся в естественно научной литературе иллюстрации, графики и формулы.

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

Первоначально изображение документа подвергается предварительной обработке, такой как удаление шума или коррекция искажений. Затем происходит сегментация изображения, то есть разделение его 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. Этот этап включает в себя распознавание формы символов, их контекста и других характеристик, что позволяет определить, какие символы были изображены на сканированном документе.

В завершение, распознанные символы объединяются в слова, предложения и абзацы, формируя полноценный текстовый документ. Точность и эффективность процесса OCR зависит от качества изображения, используемых алгоритмов распознавания, а также от языка и структуры текста.

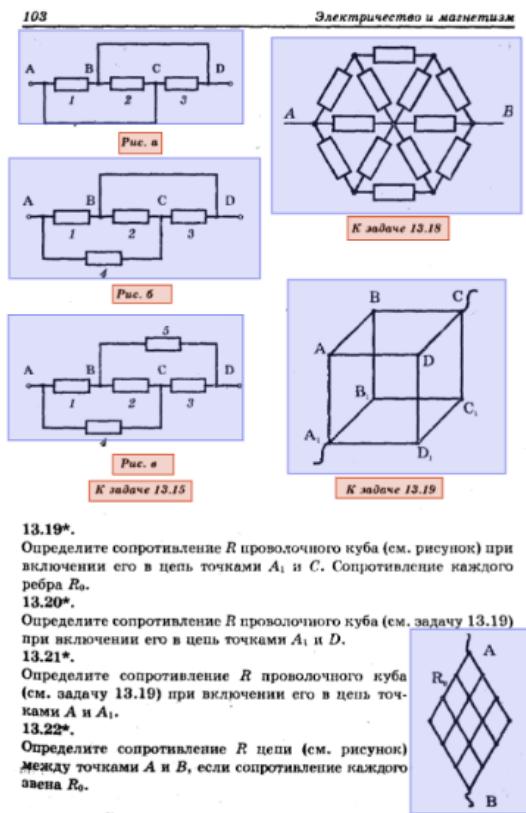


Рис. 4.3: Пример аннотированной иллюстрации из книги Генденштейн, Кирик, Гельфгат: 1001 задача по физике

Существует множество открытых пакетов для выполнения OCR

- Nougat [[blecher2023nougat](#)]
- Tesseract [[smith2007overview](#)]
- LayoutParser [[shen2021layoutparser](#)]

К сожалению, доступные открытые решения либо не поддерживают русский язык, либо не предназначены для работы с формулами. Для решения автора был разработан открытый программный пакет для Python ShuemacherOCR , предназначенный для масштабного анализа русской естественно-научной методической литературы. 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. Ключевой 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 репозитория ³.

Данные задачи были собраны из открытых педагогических источников [[libmipt](#)][[mathedu](#)] с обязательным указанием при публикации ссылок на источники.

³<https://github.com/NMashalov/SchumacherOCR>

Распознание текста по изображению выполняется нейросетью архитектуры Nougat [blecher2023nougat]. Особенностью данной архитектуры является быстрая адаптация под новые виды данных и работа с целым изображением, без необходимости промежуточного поиска регионов с текстом.

Обучение сети проводилось на корпусе препринтов статей [clement2019use], переведенных на русский язык с помощью интеллектуального ассистента ChatGPT [ouyang2022training]. Выбор был связан с возможностью сохранять оригинальную разметку TeX-документов.

Для валидации результатов был разработан открытый датасет, позволяющий измерить качество распознавания⁴.

Разметка для обучения проводилась с помощью обращения к сервису компании MathPix. Метрики качества приведены в таблице 1 и сопоставимы с результатами оригинальной модели Nougat.

Параметр	Тренировочная выборка	Отложенная выборка
BLEU	83.2	80.4
Edit distance	0.15	0.17

Для обучения на полученных данных была использована нейронная сеть YOLO [redmon2016you]. Эта архитектура нейронной сети имеет способность эффективно дообучаться на небольших выборках данных, что позволяет достигать удовлетворительных результатов. Для ситуаций, где 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 датасета. Каждое изображение включает в себя текстовую информацию, а также различные чертежи и формулы, характерные для данной области знаний.

Процесс разметки включал создание аннотаций для каждого изображения, а именно выделение границ объектов, таких как текстовые блоки, формулы и чертежи. Этот процесс требовал точности и внимательности для корректного определения границ объектов на изображении и их соответствия с аннотациями.

Для расширения датасета и обеспечения его разнообразия была применена аугментация данных. Применялись повороты, масштабирование, изменение освещения и отражение, позволили создать дополнительные вариации входных данных. Это способствовало увеличению разнообразия обучающей выборки и повышению устойчивости модели к различным вариациям данных, что важно для обеспечения ее эффективности в реальных условиях различной разметки страницы.

Метрическая оценка результатов выделения иллюстрации и аннотации

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

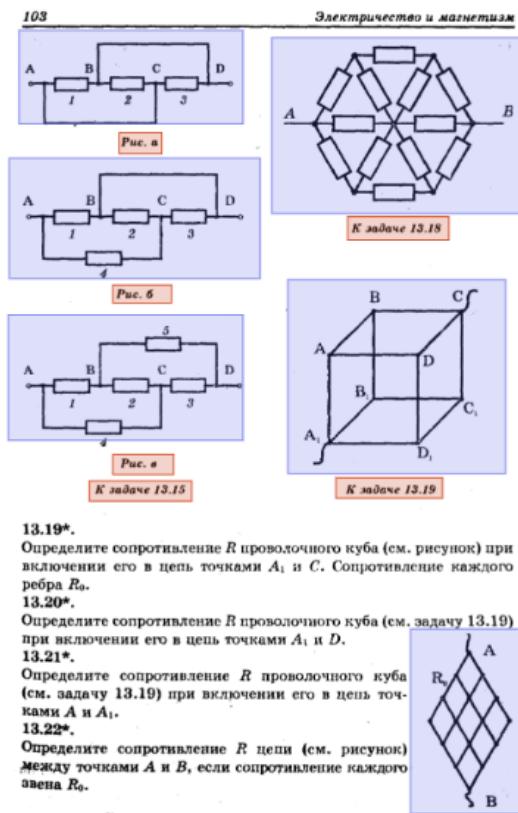


Рис. 4.4: Пример аннотированной иллюстрации из книги Генденштейн, Кирик, Гельфгат: 1001 задача по физике

Параметр	Тренировочная выборка	Отложен
mAp	78.4	
Точность распознавания ребер “изображение-аннотация”	75.2	

Автор продолжает развитие пакета для среды распределенных вычислений, использующих акторную модель для представления данных.

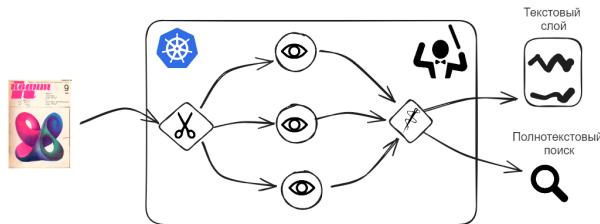


Рис. 4.5: Итоговая разметка выполняется посредством распределенных вычислений

4.2 Приложение

Современные интеллектуальные ассистенты являются многокомпонентными системами, организующими среду взаимодействия, целевой сценарий диалога, персонализацию и приватность переписки. Конкретные компоненты выбираются из целей

и возможности 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 эмпатические внимание к проблемам пользователя [jiang2023mistral] [llamatouvron2023].

Языковая модель также помогает в решении повседневных и деловых задач, сокращая академические статьи и помогая составлять планы проектов. Последние достижения также позволяют задавать вопросы по изображениям, что существенно облегчает выбор одежды и мебели [bai2023qwen]. Тем не менее современные ассистенты не способны к выполнению формальных логических операций: арифметического сложения [bubeck2023sparks], решения абстрактных логических задач [bordt2023chatgpt], соблюдение правил стратегических игр [Adam2024].



Рис. 4.6: Навыки современного ассистента ограничены коммуникацией и решением базовых задач обработки текста

В образование интеллектуальные ассистенты применяются для обучения русскому языку [РөРъСÆ2019РЧР,,СÆРхРъРъРхРæСÆСÆР,,СКР€] и рисования поясняющих графиков [bulusuautomated]. Примерами коммерческого использования ассистентов в образовании являются компании Merlin Mind и OpenAI Education. Ключевым преимуществом решений является адаптация к общеобразовательным программам стран, взаимодействие с интерактивной доской и проприетарно подготовленная база знания регулярно обновляющая предметными экспертами.

4.2.1 Проект интеллектуального ассистента

Разработка интеллектуального ассистента, как правило, долгосрочная и трудоемкая деятельность, требующая внимания к планированию и организации. В секции описан

Определение **Интеллектуальные ассистенты** прикладное программное обеспечение, выполняющее задачи пользователя, согласно командам на естественном языке. Как правило ассистенты разделяют по их применению:

- **разговорный** ассистент не имеет заранее заданной задачи и общается с пользователями для поддержания беседы
- **деловой** ассистент направлен на решение конкретной задачи

Современные системы ассистента организованы модульно. Такой подход позволяет эффективно организовать совместную работу. Модули разделяются по формальному исполнению.

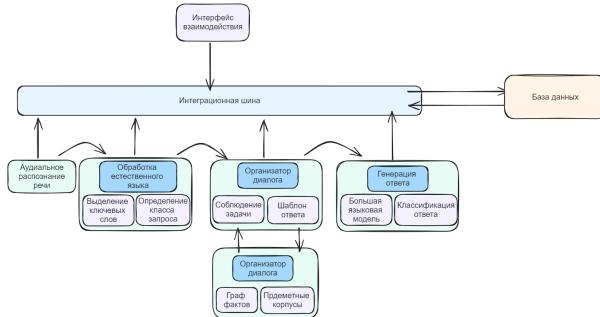


Рис. 4.7: Модульная система позволяет эффективно организовывать командную работу

выполнен в парадигме клиент-серверной разработки. Такая архитектура позволяет заменять интерфейс и программное обеспечение без необходимости в

Таким образом, для успешной адаптации ассистента необходимы

Итоговым решением, стало рекомендательную систему как систему долгосрочного планирования и ассистента как эмпатического посредника.

Для адаптации открытой реализации для работы с детьми необходимо создание формата и предмета коммуникации.

Бранные слова были исключены с помощью библиотеки PyMorphy [Korobov2015morph], выделяющую нормальную форму слова для сравнения с опорным корпусом неэтических слов.



Рис. 4.8: Модули PyMorphy [Korobov2015morph] и CLIP позволяют исключить неэтическое общение и изображение

Байесовы рейтинговые модели задают правила обновления апостериорных представлений рейтинга. Наиболее известным примером байесовой модели рейтинга яв-

ляется модель Брэдли-Терри и эквивалентная ей модель Эло. Ключевым преимуществом байесовых систем является возможность пересчета рейтинга сразу после матча. Таким образом система приобретает адаптивность, важную в коммуникациях в настоящем времени.

В такой постановке ассистент следит за эмоциональным откликом обучающегося на материал, узнает ее причину и в случае несоразмерной уровню нагрузке сообщает управляющей системе о потребности изменения сложности.

Плюсами такого подхода является

- интерпретируемость
- сужение ассистента до строгой постановки поиска наилучших навыков общения исходя из А/В тестирования.

4.2.2 Подготовка веб-приложения

Интерфейс приложения организует работу с ассистентом. Для реализации доступного в использовании приложения был использован гештальт-дизайн [wertheimer1938laws], задающий правила композиции элементов интерфейса. Также были учтены стандарты WAI-ARIA [craig2009accessible], задающие правила высокого контраста и выраженности контуров, позволяющих выполнять навигацию слабовидящим людям. Также функционал приложения ограничено доступен и незрячими людям, использующим специальные приложения для аудио отображения содержания сайта.

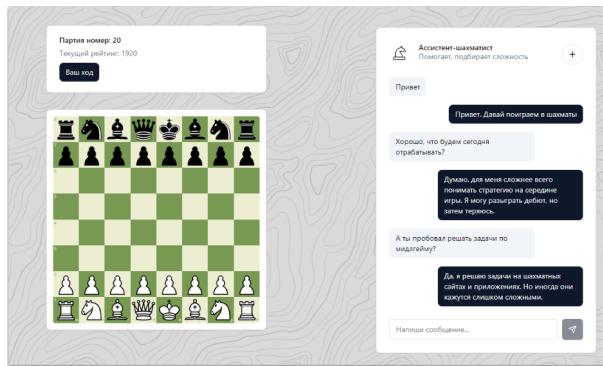


Рис. 4.9: Интерфейс имеет четкие и контрастные элементы взаимодействия. Доска для игры и диалоговое окно размещены совместно

Веб-приложение доступно при подключение через браузер по адресу доменного имени www.mathema-online.xyz. Технологии криптографии обеспечивают безопасность соединения, выпущенные сертификаты доменного имени исключают возможность подмены имени.

Интерфейс реализован с помощью популярной библиотеки React для языка программирования JavaScript [ackenheimer2015introduction]. Такой подход позволяет дескриптивно описывать элементы вебсайта, программно реагируя на взаимодействие.

ствие пользователя. Ключевой особенностью подхода является возможность использовать открытые профессионально подготовленные интерфейсы

При использовании данных из открытых источников используются ссылки согласно требования Гражданского кодекса Российской Федерации [law1274] [law1260].

4.2.3 Описание разработанных игр

В секции представлены разработанные игры, использующие современные технологии машинного обучения для персонального обучения. Выбор направления был выполнен с учетом текущих возможностей больших языковых моделей. Таким образом, в качестве предмета обучения были выбраны стратегические игры, развивающие навыки долгосрочного планирования и критического мышления. Также было исследовано применение диффузионных моделей для растрового рисования, учащее передавать структуру и природу объектов мира.

Современные большие языковые модели пока не способны к полноценному ведению игры. В работе [Adam2024] проведен анализ силы игры ассистента ChatGPT путем сравнения с шахматным движком Stockfish. Статистические исследования показывают, что текущий уровень игры модели соответствует рейтингу Эло 1600 [elo1967proposed]. Такой уровень соответствует начальному уровню игрока в шахматы. Исследователи также отмечают неспособность ассистента к строгому исполнению правил игры и наблюдают противоречия в стратегии даже в краткосрочной перспективе.

Исходя из текущих возможностей был предложен гибридный подход, заключающийся в совмещение языковой модели с открытым шахматным движком StockFish [acher2016large]. В такой постановке интеллектуальный ассистент отвечает на вопросы пользователя по ходу игры и рассказывает о возможных стратегических решениях, исходя из ситуации на доске. Движок задает уровень оппонента и помогает в

Обновление рейтинга выполняется согласно модели Эло, в зависимости от сложности оппонента по правилу:

$$x_{i+1} = x_i + K(\beta) [X_i - P(X_i = 1)] \quad (4.1)$$

, 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 по необходимости изменяет сложность
3. ассистент оглашает результаты игры и обновляет рейтинг. Пользователь может попросить совет по исходу

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 по порогу отсечки согласно алгоритму, описанному в 3.5. Такая система поощряет дух соревнования для достижений выдающихся результатов.



Рис. 4.10: Достижение на примере награды за победу в сеансе одновременной игры

Разработки игры по рисованию была выполнена с использованием открытой диффузионной модели 2.2.1, 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 из наличия фона, декорации и сложной композиции.

Алгоритм выполнения рисунка состоит из трех последовательных этапов

1. пользователь вводит предмет интереса 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, позволяющей избегать неэтичного рисунка [radford2021learning]. Уровень сложности регулируется с путем изменения композиции и наличия фона.

4.3 Рейтинговая система

В секции описана разработка, содержащая предложенного алгоритма и численного эксперимент, изучающий поведение разностной схемы в краевых постановках.

Интерактивное рисование

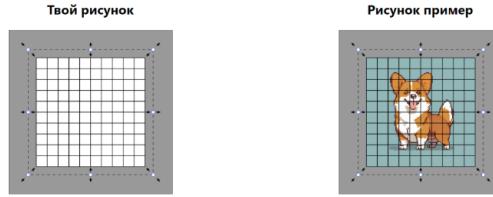


Рис. 4.11: Рисунок выполняется путем сопоставления результата с опорным изображением

4.3.1 Адаптация рейтинговой системы

Постановка представляет тест как стохастический ряд вида $\{x\}_{t=0}$ каждый элемент, которого является случайной бернулевской величиной с параметром s . Для ввода управляющей переменной задается сложность задачи d , параметризующий в совокупности с функцией отклика учащегося f , переменную $s_t = f(d)$.

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

- 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

Алгоритм, отвечающий заданным требованиям был предложен в работе [yazidi2020balanced]. Авторы предложили правило обновления сложности:

$$d_{t+1} = \Pi(d_t + \lambda(x(t) - s^*)), \quad (4.2)$$

где функция Π является ограничивающим оператором вида

$$\Pi_H(d) = \begin{cases} d, & \text{прт } 1 < d < 0 \\ 1, & d \geq 1 \\ 0, & \text{при } d \leq 0 \end{cases} \quad (4.3)$$

Алгоритм по форме соответствует алгоритму Роббинса-Монро[robbins1951stochastic], известному в теории стохастической аппроксимации. Отметим, что поскольку коэффициент λ является постоянной, то сходимость по алгоритму Роббинса-Монро не гарантируется.

$$x_{n+1} = x_n - a_n(x - b_n). \quad (4.4)$$

Схема доказательства

1. используем модифицированный алгоритм Роббинса-Монро, используя
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 $\mathbf{D}(x_n) \rightarrow \min$
[hu1997strong] [hu1998sequential]
3. определим явное выражение a_n и b_n для априорного представления о функции отклика в виде параметрической модели Эло $s(d, \alpha, \beta) = 1 - \frac{1}{1 + \exp(-\beta \cdot (d - \alpha))}$

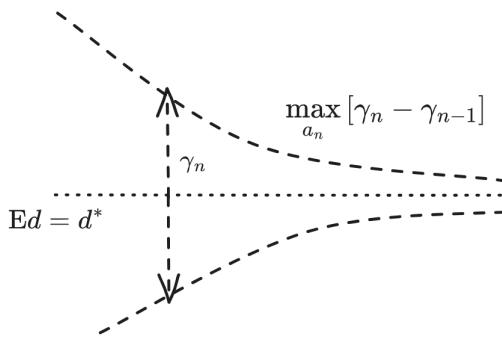


Рис. 4.12: Использование несмещенной оценки с постепенной редукцией дисперсии

Теорема. Адаптированный алгоритм Роббинса-Монро для случая наблюдений, имеющих бернуллевское 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

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

Доказательство Сходимость метода приведена в апPENDиксе работы ???. 1) Используем рекуррентное представление для поиска оптимальных параметров a_n, b_n

$$d_{n+1} = d_n - a_n(x_n - b_n). \quad (4.5)$$

Рассчитаем матожидание $E_{d_n \sim p(d_n)}$ как:

$$Ed_{n+1} = Ed_n - a_n(Ex_n - b_n). \quad (4.6)$$

Запишем условия $\text{Ed}_{n+1} = d^*$ с учетом $\text{Ed}_n = \dots \text{Ed}_1 = d^*$:

$$a_n(\text{Es}(d_n) - b_n) = 0. \quad (4.7)$$

Следовательно,

$$b_n = \text{Es}(d_n). \quad (4.8)$$

Коэффициент a_n найдем из минимизации дисперсии $\mathbf{D}d_{n+1}$. Запишем $\mathbf{D}_{x_n \sim \text{Bern}(x|s(d_n))}$ для выражения 4.5:

$$\mathbf{D}d_{n+1} = \mathbf{D}d_n + a_n^2 \mathbf{D}(x_n - b_n) - 2a_n \mathbf{E}[(x_n - b_n)d_n]. \quad (4.9)$$

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

$$\mathbf{D}(x_n - b_n) = \text{Es}(d_n)\mathbf{E}(1 - s(d_n)) = \text{Es}(d_n)(1 - \text{Es}(d_n)). \quad (4.10)$$

С учетом $b_n = \text{Es}_d$ и $\text{Ed}_n = d^*$:

$$\mathbf{E}[(x_n - b_n)d_n] = \mathbf{E}[x_nd_n] - b_n\mathbf{E}d_n = \mathbf{E}[s(d_n)d_n] - \text{Es}(d)d^*. \quad (4.11)$$

Тогда:

$$\mathbf{D}d_{n+1} = \mathbf{D}d_n + a_n^2 \text{Es}(d_n)\mathbf{E}(1 - s(d_n)) - 2a_n \mathbf{E}[s(d_n)(d_n - d^*)]. \quad (4.12)$$

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

$$a_n = \frac{\mathbf{E}[(d_n - d^*)s(d_n)]}{\text{Es}(d_n)(1 - \text{Es}(d_n))}. \quad (4.13)$$

Тогда связь между дисперсиями на каждом шаге запишется как:

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

2) 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^* :

$$d^* = \alpha - \beta \log \left(\frac{s^*}{1 - s^*} \right). \quad (4.15)$$

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 4.13:

$$\gamma_n = \gamma_{n-1} - \frac{(\mathbf{E}[(d_n - d^*)s(d_n)])^2}{\text{Es}(d_n)(1 - \text{Es}(d_n))}. \quad (4.16)$$

Найдем b_n из 4.8:

$$b_n = \int_{-\infty}^{\infty} s(x) \mathcal{N}_{d^*, \gamma_n}(x) dx = \int_{-\infty}^{\infty} \sigma(x, \alpha, \beta) \mathcal{N}_{d^*, \gamma_n} dx. \quad (4.17)$$

Для этого используем аппроксимацию логнормального интеграла через функцию ошибки $\text{erf}(x) = \int_{-\infty}^x \exp(-t^2)$:

$$\sigma(x, \alpha, \beta) \approx \frac{1}{2} + \frac{1}{2} \text{erf} \left(\frac{(x - \alpha)\sqrt{\pi}}{4\beta} \right). \quad (4.18)$$

Свертка $\text{erf}(x)$ с плотностью вероятности гауссового распределения $\mathcal{N}(d^*, \gamma_n)$ является табличным интегралом [**ng1969table**]:

$$b_n \approx \int_{-\infty}^{\infty} \left[\frac{1}{2} + \frac{1}{2} \text{erf} \left(\frac{\sqrt{\pi}}{4\beta}(x - \alpha) \right) \right] \mathcal{N}_{d^*, \gamma_n}(x) dx = \frac{1}{2} + \frac{1}{2} \text{erf} \left(\frac{\sqrt{\pi}(\alpha - d^*)}{4\beta \sqrt{1 + \frac{\pi \gamma_n^2}{8}}} \right). \quad (4.19)$$

Найдем a_n из 4.13:

$$\begin{aligned} a_n &= \frac{1}{b_n(1 - b_n)} \int_{-\infty}^{\infty} d(\sigma(x\alpha, \beta) - d^*) \mathcal{N}_{d^*, \tau}(x) dx \\ &\approx \int_{-\infty}^{\infty} x \left[\frac{1}{2} - d^* + \frac{1}{2} \text{erf} \left(\frac{(x - \alpha)\sqrt{\pi}}{4\beta} \right) \right] \mathcal{N}_{d^*, \gamma_n}(x) dx \\ &= \frac{d^*}{2\gamma_n^2} \text{erf} \left(\frac{\sqrt{\pi}(\alpha - d^*)}{4\sqrt{1 + \frac{\pi \gamma_n^2}{8}}} \right) + \frac{1}{4\gamma_n^2} \frac{1}{b_n(1 - b_n)} \int_{-\infty}^{\infty} x \text{erf} \left(\frac{(x - \alpha)\sqrt{\pi}}{4\beta} \right) \exp \left(\frac{(x - d^*)^2}{2\gamma_{\frac{1}{b_n(1-b_n)}}^2 n^2} \right) dx \\ &= \frac{d^*}{2\gamma_n^2} \text{erf} \left(\frac{\sqrt{\pi}(\alpha - d^*)}{4\sqrt{1 + \frac{\pi \gamma_n^2}{8}}} \right) + \frac{1}{4\gamma_n^2} \frac{1}{b_n(1 - b_n)} I(\alpha, \beta, d^*, \gamma_n). \end{aligned} \quad (4.20)$$

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^* :

$$T'_{d^*} = -\frac{1}{2\gamma_n^2} I(\alpha, \beta, d^*, \gamma_n) + \frac{d^*}{2\gamma_n^2} \text{erf} \left(\frac{\sqrt{\pi}(\alpha - d^*\beta)}{4\beta \sqrt{1 + \frac{\pi \gamma_n^2}{8}}} \right). \quad (4.21)$$

С другой стороны:

$$\begin{aligned} T'_{d^*} &= \frac{1}{2} \text{erf} \left(\frac{\sqrt{\pi}(\alpha - d^*)}{4\beta \sqrt{1 + \frac{\pi \gamma_n^2}{8}}} \right)' = -\frac{2}{\sqrt{\pi}\beta} \frac{\sqrt{\pi}}{4\sqrt{1 + \frac{\pi \gamma_n^2}{8}}} \frac{1}{2} \exp \left(\frac{-\pi(\alpha - d^*)^2}{16\beta^2(1 + \frac{\pi \gamma_n^2}{8})} \right) = \\ &= -\frac{1}{4\beta \sqrt{1 + \frac{\pi \gamma_n^2}{8}}} \exp \left(\frac{-\pi(\alpha - d^*)^2}{16\beta^2(1 + \frac{\pi \gamma_n^2}{8})} \right). \end{aligned} \quad (4.22)$$

Приравнивая 4.21 и 4.22 получаем:

$$I = \frac{1}{4\sqrt{1 + \frac{\pi\gamma_n^2}{8}}} \exp\left(\frac{-\pi(\alpha - d^*)^2}{16\beta^2(1 + \frac{\pi\gamma_n^2}{8})}\right) - \frac{d^*}{2\gamma_n^2} \operatorname{erf}\left(\frac{\sqrt{\pi}(\alpha - d^*)}{4\beta\sqrt{1 + \frac{\pi\gamma_n^2}{8}}}\right). \quad (4.23)$$

Подставляем I в 4.20 получаем аппроксимацию a_n :

$$a_n \approx \frac{1}{b_n(1 - b_n)} \frac{1}{4\beta\sqrt{1 + \frac{\pi\gamma_n^2}{8}}} \exp\left(\frac{-\pi(\alpha - d^*)^2}{16\beta^2(1 + \frac{\pi\gamma_n^2}{8})}\right). \quad (4.24)$$

Также получим γ_n из 4.23 и 4.16:

$$\gamma_n = \gamma_{n-1} - \frac{\gamma_{n-1}^2}{1 + \gamma_{n-1}} \frac{1}{b_n(1 - b_n)} \exp\left(\frac{-\pi(\alpha - d^*)^2}{16\beta^2(1 + \frac{\pi\gamma_n^2}{8})}\right)^2. \quad (4.25)$$

■

Явно приведем ключевые выражения для численного выражения:

1. Приближение оптимального корня:

$$d^* = \alpha - \beta \log\left(\frac{s^*}{1 - s^*}\right). \quad (4.26)$$

Коэффициенты a_n , b_n , γ_n рассчитываются рекурсивно

$$\begin{aligned} \gamma_n &= \gamma_{n-1} - \frac{\gamma_{n-1}^2}{1 + \gamma_{n-1}} \frac{1}{b_n(1 - b_n)} \exp\left(\frac{-\pi(\alpha - d^*)^2}{16\beta^2(1 + \frac{\pi\gamma_n^2}{8})}\right)^2 \\ a_n &= \frac{1}{b_n(1 - b_n)} \frac{1}{4\beta\sqrt{1 + \frac{\pi\gamma_n^2}{8}}} \exp\left(\frac{-\pi(\alpha - d^*)^2}{16\beta^2(1 + \frac{\pi\gamma_n^2}{8})}\right) \\ b_n &= \frac{1}{2} + \frac{1}{2} \operatorname{erf}\left(\frac{\sqrt{\pi}(\alpha - d^*)}{4\beta\sqrt{1 + \frac{\pi\gamma_n^2}{8}}}\right). \end{aligned} \quad (4.27)$$

Процедуру пересчета коэффициента можно выполнить до проведения испытания, тем самым ускорив исполнение программы.

4.3.2 Численные эксперименты

Численное моделирование исследует поведение предложенного алгоритма для функции логистической регрессии в сравнение с классическими подходами. Исходный код на языке Python доступен в открытом репозитории диссертации⁵.

Ключевым параметром для анализа является соотношение изменения параметра сложности задачи Δd к параметру роста β . Таким образом, были исследованы две

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

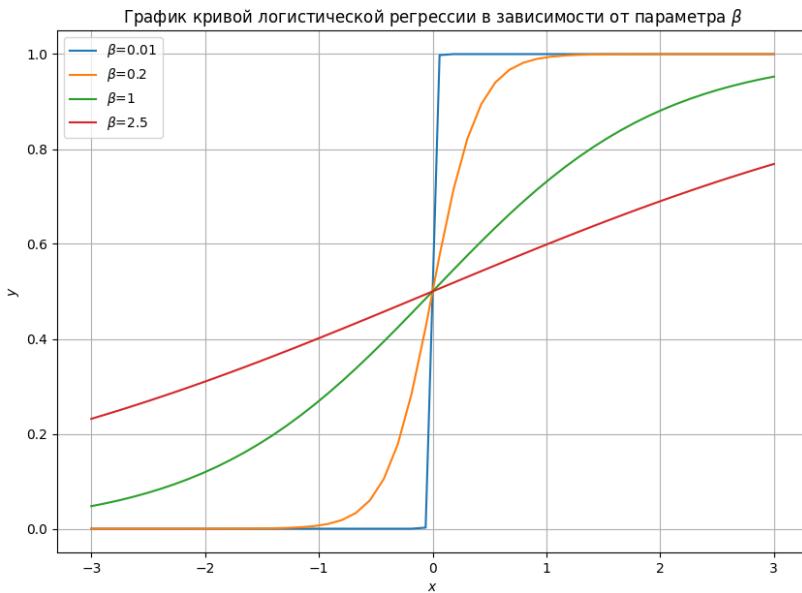


Рис. 4.13: Демонстрация изменения графика логистической регрессии от параметра β

ключевые краевые постановки:

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

В каждом эксперименте сравнивалась эффективность метода с классическим алгоритмом Роббинса-Монро и его аналогом с фиксированным коэффициентом λ как в работе [yazidi2020balanced]. Дополнительно изучено влияние модификации по методу Поляка скользящим средним.

Гладкие графики траекторий получается путем визуализации среднего и перцентилей распределения. Для их численного расчета используется метод бутстрэп. Эксперимент проводится $B >> 1$ раз, после чего статистики считаются путем расчета распределения.

В разделе приведены ключевые графики, полный набор доступен в приложении статьи. 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 . 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 $s^* = 0.4$ ⁶. Отметим также чувствительность алгоритм

⁶Для наглядности в таблице классический алгоритм Роббинса-Монро сокращается до сокращения "Р.-М." с указанием параметра шага.

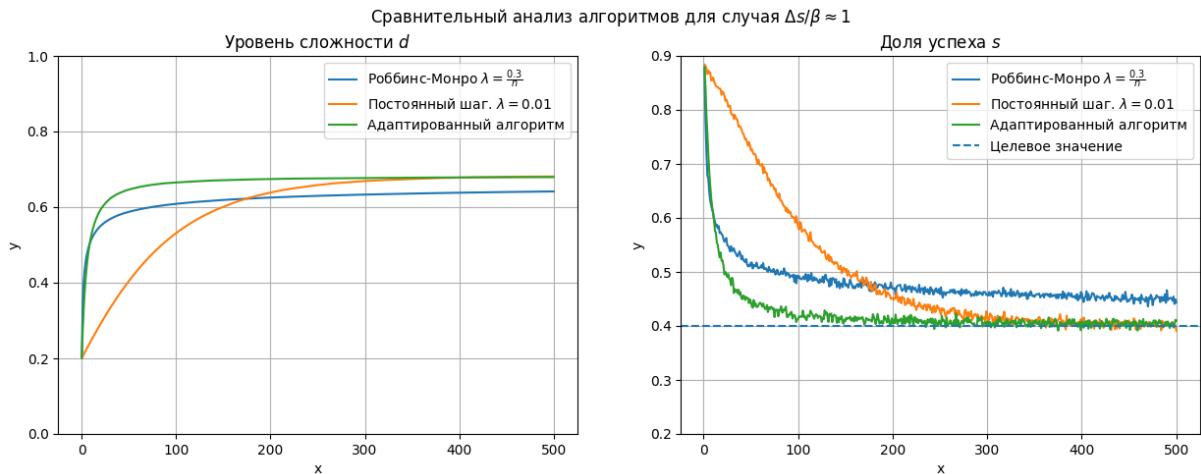
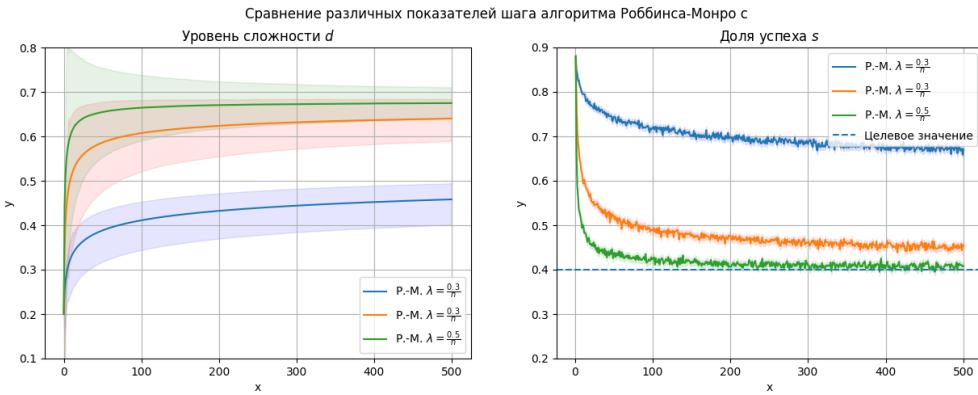


Рис. 4.14: Предложенный алгоритм

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

Роббинса-Монро к параметру шага. 4.15


 Рис. 4.15: Классический алгоритм Роббинса-Монро чувствителен к параметру λ

Случай $\Delta d/\beta \approx 0$

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

⁷ Для наглядности в таблице классический алгоритм Роббинса-Монро сокращается до сокращения "Р-М" с указанием параметра шага.

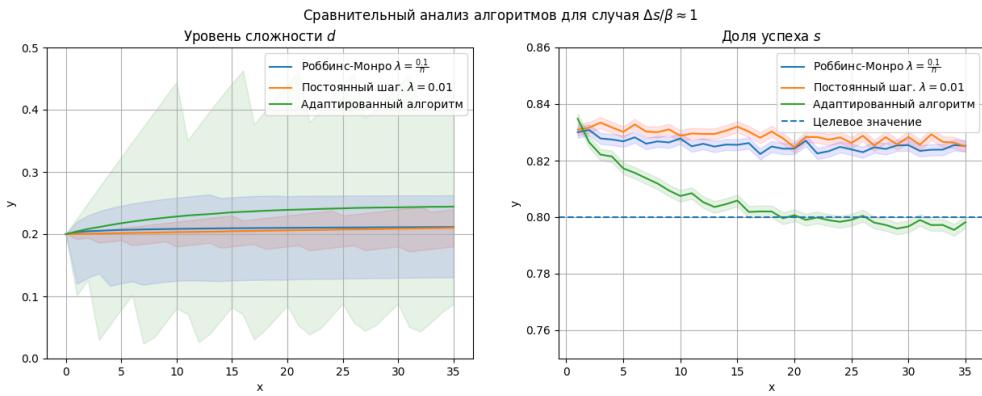


Рис. 4.16: Предложенный алгоритм имеет высокую скорость реакции d

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 4.2: Сравнение числа шагов сходимости в постановке $\Delta d/\beta \approx 0$

Случай значительного отличия априорных представлений о наклоне кри- вой от действительного

Рассмотрен случай, в котором β априорная значительно отличается β^* действи-
тельного.

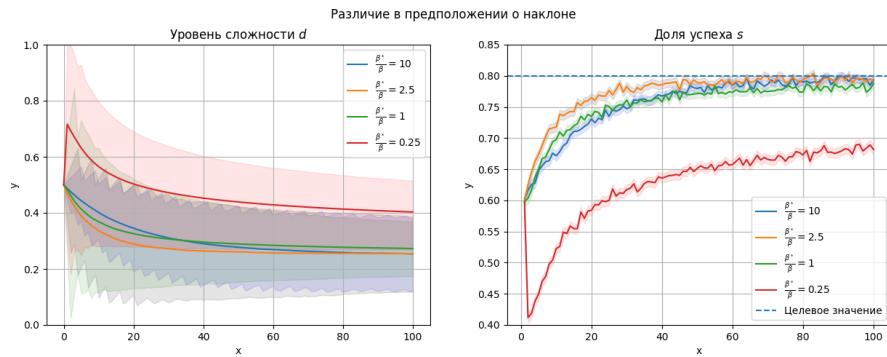


Рис. 4.17: Значительные различия в предположениях о наклоне логистической
регрессии приводят к снижению эффективности алгоритма

Модификация скользящим средним

Численно исследуем применимость метода скользящего среднего к предложенно-
му алгоритму и алгоритму с постоянным шагом.

Расходимость предложенного метода связана с нарушениями условий 1.47.

Таким образом, метод скользящего среднего

1. позволяет использовать больший шаг для обеспечения большей скорости схо-
димости

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 4.3: Сравнение числа шагов сходимости в постановке различающихся априорных представлений о наклоне кривой от действительного

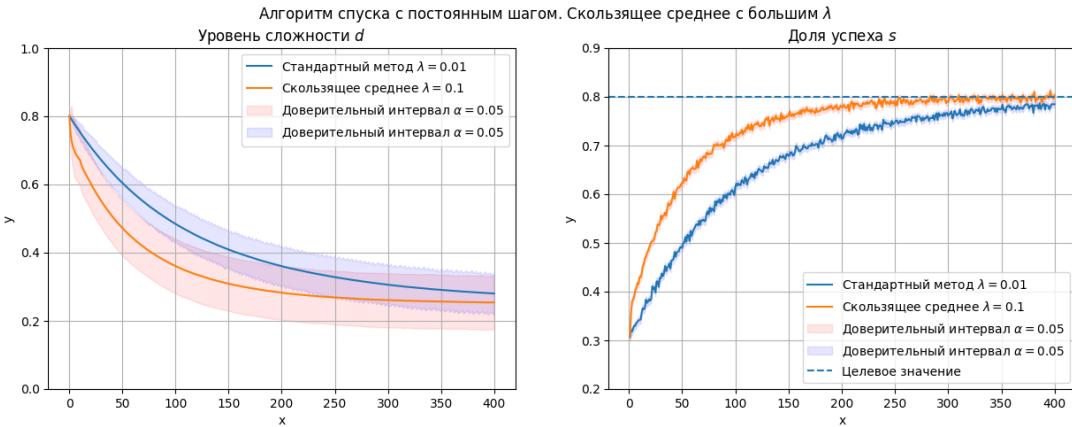


Рис. 4.18: Метод скользящего среднего позволяет использовать больший параметр шага, не теряя устойчивость метода

2. не применим к предложенному алгоритму

Полный набор задач моделирования приведен в аппендиксе работы 6.1.

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 4.4: Сравнение числа шагов с применением метода скользящего среднего

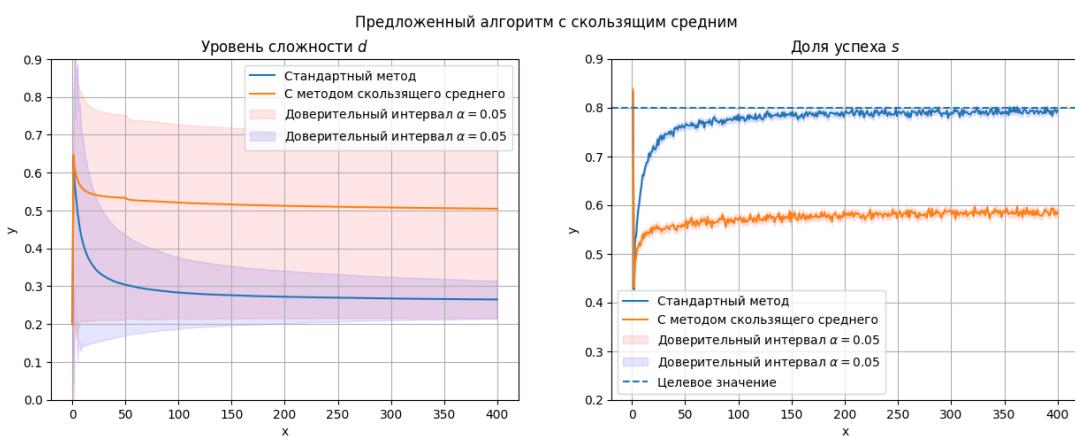


Рис. 4.19: Метод скользящего среднего не применим к предложенному алгоритму

Глава 5

Заключение

5.1 Итоги работы

Поставленные цели по адаптации большой языковой цели к задачам разработки интеллектуального ассистента были успешно достигнуты.

Ассистент сопровождает процесс обучения шахматам и рисования, задает оптимальный уровень сложности и общается на тему.

Исследована постановка алгоритма Роббинса-Монро в условиях отклика представленной случайной бернуллевской величины. Для случая ответа в виде логистической функции получен адаптивный численный алгоритм. К его ключевые преимуществам можно отнести:

- оптимальную скорость сходимости при выполнение условий теоремы 4.3.1;
- стабильную дисперсию s и d на всех шагах оптимизации;
- работу с естественными параметрами логистического распределения на всем интервале оптимизации. Это обстоятельство выгодно отличает метод от классических методов, требующих подбора шага оптимизации.

Тем не менее алгоритм требует выбора априорных представлений о наклоне функции логистического распределения 4.17. Выполнить данный расчет возможно на экспериментальных данных, используя в качестве бинарного классификатора логистическую регрессию.

5.2 Дальнейшая работа

Секция содержит постановки научных работ на текущий момент находящиеся в исследовании.

Совместное обучение подразумевает одновременное достижение высоких уровней компетенций в предмете изучения. Такое обучения, как правило, сопровождается подготовленным методическими материалами общими для всех обучающихся.

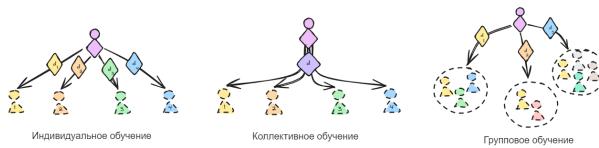


Рис. 5.1: Групповое обучение позволяет задать баланс между объемом проверки и специфичностью задания

Задача преподавателя в наиболее успешном прохождение учащихся методической программы. При внесении изменений в курс оценивается изменение среднего показателя учащихся. Усредненные аналитические показатели позволяют принимать решения с статически заданными порогами риска и приобретений. Это обеспечивает устойчивые рост образования в среднем. Для развития практики полезно учитывать индивидуальные потребности учащихся, заключающиеся в разном уровне освоения материала и задачах его использования. Разрешить проблему можно путем индивидуального образования, но такой подход осложнен увеличением преподавательской нагрузки. Одним из компромиссных решений является групповое образование, обеспечивающее баланс между специфичностью задания для учащегося и временем для проверки для преподавателя.

Обучение в группах требует аналитического описания в трех направлениях:

1. определение правил объединения в группы
2. подбор сложности задания для группы, включающий модель оценки эффективности совместной работы
3. оценка распределения нагрузки в случае группового задания

Разработанный в работе алгоритм адаптивного подбор сложности применим только в постановках индивидуального обучения, практическое применение которого актуально для цифровых систем. В постановке коллективного обучения предложенный алгоритм сможет обеспечить оптимальный уровень сложности для всего коллектива в общем. Такая модель слишком груба, поскольку не учитывается важные для образовательного коллектива процессы соревнования и кооперации учащихся. Учет коллективных эффектов выполняется в постановках распределенной оптимизации, теории марковских полей и теоретической физике. Для анализа малых групп используются ковариационные матрицы и супераддитивные функции.

Супераддитивной называется функция f для которой

$$\forall x, y, x + y \in \text{dom}(f) \rightarrow f(x + y) \geq f(x) + f(y) \quad (5.1)$$

Такие функции позволяют оценить по индивидуальным качествам учащихся их совместный вклад в дело. Примерами супераддитивных функций являются

- min-sum $\sum_i \min([\vec{x}]_i, s^*)$, где s^* - порог отсечки

- max-mean $N \cdot \bar{x} + \max_i(\vec{x} - \bar{x})$
- квадратичная форма $\vec{x}^T A \vec{x}$

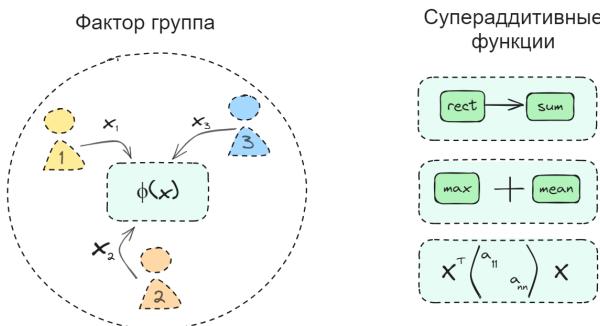


Рис. 5.2: Урпповые задания

Практически функцию улучшение среднего результата в классе.

Общий результат лучше исключает случайности и может быть оценкой эффективности как отдельного педагога, так и всего преподавательского коллектива. Тем не менее такой подход не всегда может учесть индивидуальные образовательные потребности учащихся. Одним из возможных разрешений такой проблемы является объединение учащихся в группы для выполнения задач.

5.2.1 Благодарности

Автор благодарит кафедру инновационной педагогики Московского физико-технического института за помощь в подготовке работы. Особенno я хотел бы поблагодарить научного консультанта Дмитрия Евгеньевича Щербакова и научного руководителя Ивана Валерьевича Ященко за ценные консультации. Также я признателен за поддержку в организации работы и рекомендации участия в научных конференциях секретарю кафедры Наталье Николаевне Парамоновой.

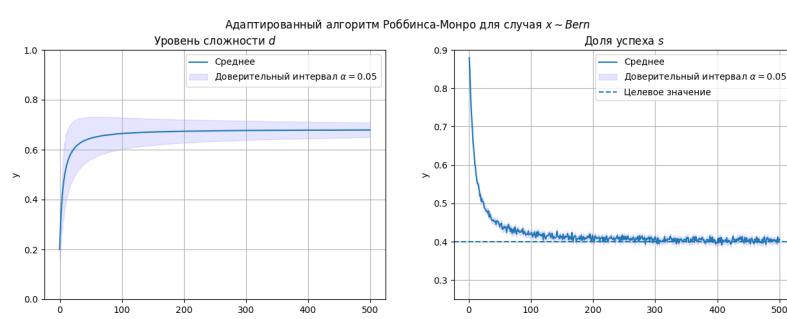
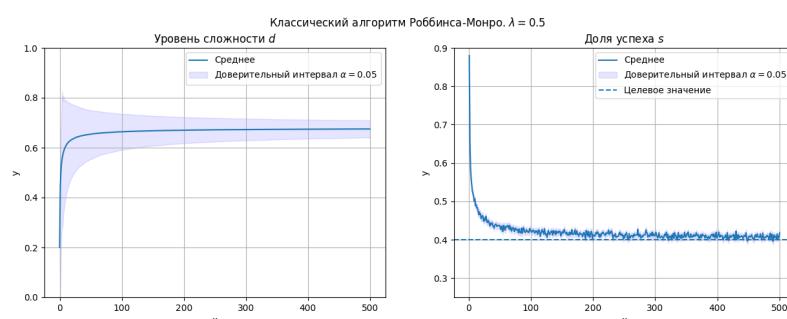
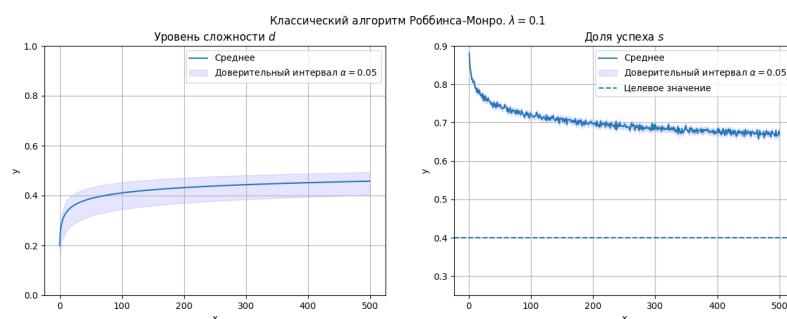
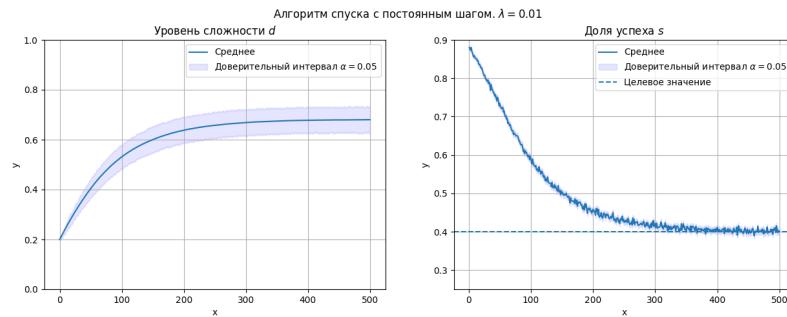
Глава 6

Дополнительные материалы

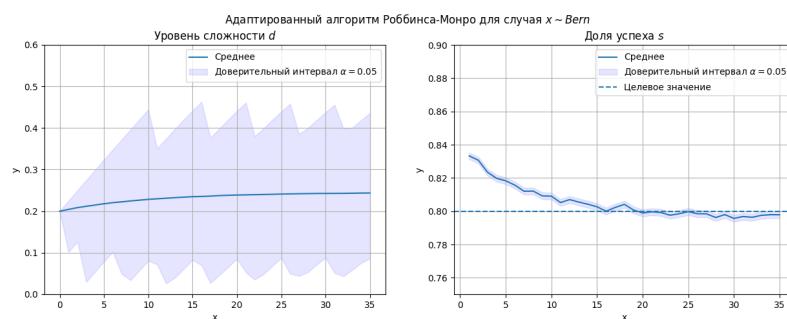
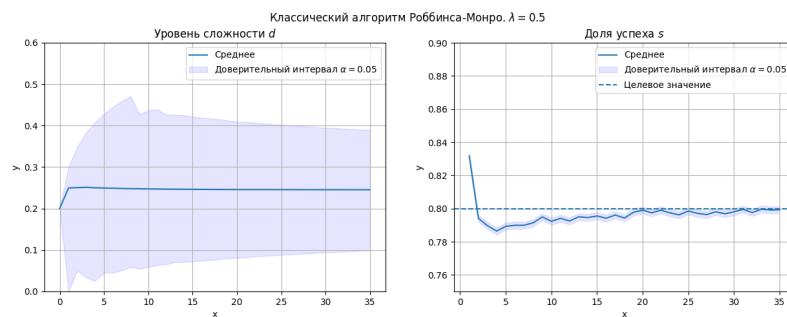
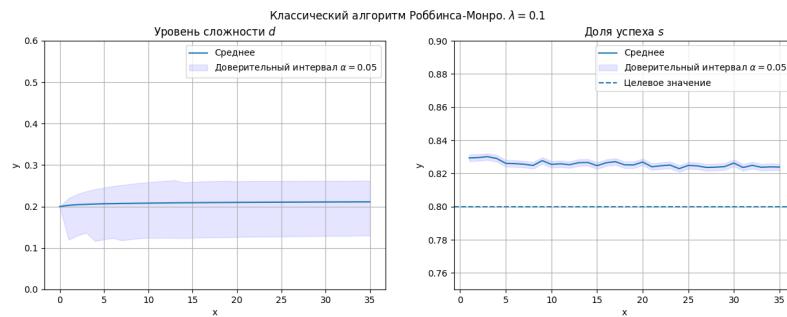
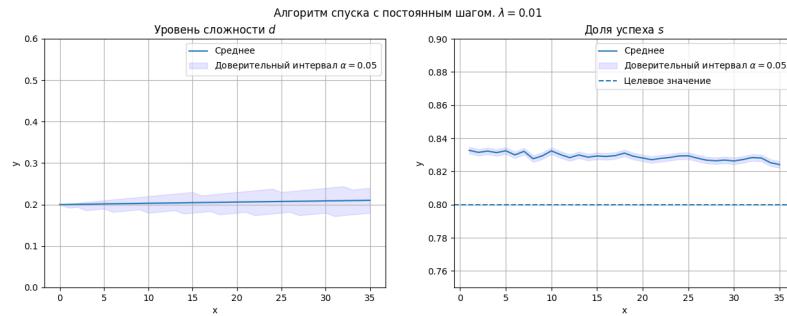
6.1 Численное моделирование

6.1.1 Приложение

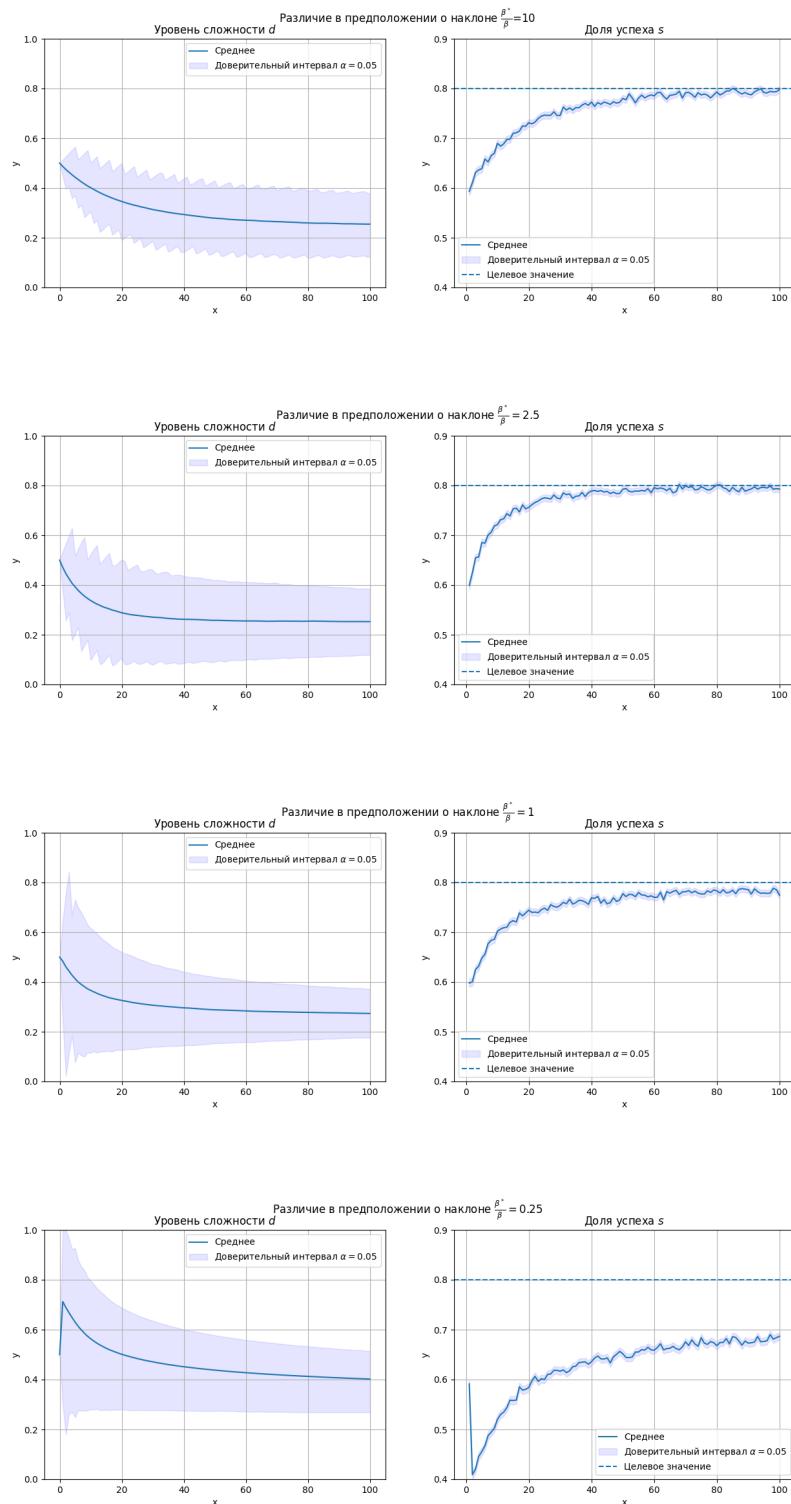
Эксперимент 1



Эксперимент 2



Эксперимент 3



Эксперимент 4

