

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий - РТФ
Базовая кафедра «Аналитика больших данных и методы видеоанализа»

ДОПУСТИТЬ К ЗАЩИТЕ ПЕРЕД ГЭК

/ Зав. кафедрой «АБД и МВ»

(подпись)

М.А. Медведева
(Ф.И.О.)

02.06.2023 г.

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ)**

«Поддержка принятия решений на различных этапах жизненного цикла ИТ-
инновации с использованием ChatGPT»

Научный руководитель: Берг Д.Б.
профессор, Доктор физико-математических наук

Студент группы РИМ-210981 Ерицын Г.А.

Екатеринбург
2023

РЕФЕРАТ

Тема магистерской диссертации: Поддержка принятия решений на разных этапах ЖЦ ИТ-инновации с использованием ChatGPT.

Магистерская диссертация выполнена на 95 страницах, содержит 9 рисунков, 66 использованных источников.

Цель работы:

Исследование и анализ использования модели ChatGPT для помощи в принятии решений при работе над научными проектами.

Исследование и анализ использования модели ChatGPT для помощи в принятии решений на этапах ЖЦ разработки программного обеспечения.

Объект исследования - модель ChatGPT и ее применение для поддержки принятия решения в научных работах и IT-компаниях.

Предмет исследования - алгоритм применения модели ChatGPT для поддержки принятия решения в ЖЦ ИТ-инновации.

Задачи :

1. Обоснование актуальности темы для научного исследования.
2. Исследование влияния ChatGPT на процессы ЖЦ научного исследования.
3. Исследование влияния ChatGPT на процессы ЖЦ разработки ПО.
4. Формирование алгоритма взаимодействия с ChatGPT.
5. Оценка экономической эффективности.

СОДЕРЖАНИЕ

РЕФЕРАТ	1
ВВЕДЕНИЕ.....	4
1. Принятие решений при управлении жизненным циклом инновации	8
1.1 Анализ библиометрических показателей области исследования.....	8
1.2 Обоснование актуальности выбранной темы	11
1.3 Жизненный цикл инновации	15
1.4 Принятие решений на этапах научно-исследовательской работы	20
1.5 Принятие решений на этапах жизненного цикла разработки.....	24
1.6 Обоснование значимости использования искусственного интеллекта и на примере ChatGPT	27
1.7 Особенности языковых моделей.....	31
1.8 Результаты и выводы.....	40
2 Методы и алгоритмы взаимодействия с языковыми моделями.....	41
2.1 Методы взаимодействия с языковыми моделями	41
2.2 Методики создания подсказок	43
2.3 Алгоритм взаимодействия с ChatGPT	46
3 Особенности использования большой языковой модели chatgpt на различных этапах жизненного цикла.....	48
3.1 Применение ChatGPT на этапе научного исследования	48
3.1.1 Подготовка литературного обзора для научного исследования	48
3.1.2 Анализ литературы	53
3.1.3 Обоснование научной новизны научного исследования	58
3.1.4 Генерация идей.....	60
3.1.5 Исследование алгоритмов постановки задач управления	63

3.2	Применение чата ChatGPT на этапах разработки ПО	67
3.2.1	Применение ChatGPT при тестировании программного обеспечения	67
3.2.2	Применение ChatGPT в технической поддержке программного обеспечения	80
3.3	Экономическое обоснование.....	88
	Заключение	95
	Список использованных источников	100

ВВЕДЕНИЕ

В современном информационном обществе, где технологические инновации развиваются с невероятной скоростью, эффективное принятие решений становится важным фактором успеха для организаций. Особенно в области информационных технологий (ИТ), где новые идеи, продукты и сервисы регулярно возникают и проходят через жизненный цикл разработки и внедрения, необходимо иметь инструменты и методы для поддержки принятия решений на разных этапах этого процесса.

В рамках данного исследования предлагается рассмотреть использование ChatGPT - мощной модели генерации естественного языка, разработанной компанией OpenAI, для поддержки принятия решений на различных этапах жизненного цикла ИТ-инновации. ChatGPT является примером передовой технологии искусственного интеллекта, которая обладает способностью генерировать тексты, имитируя человеческий стиль и интеллект.

Полученные результаты могут быть полезными для исследователей и специалистов в области искусственного интеллекта, а также для компаний, занимающихся разработкой программного обеспечения.

Целями работы являются:

- Исследование и анализ использования модели ChatGPT для помощи в принятии решений при работе над научными проектами
- Исследование и анализ использования модели ChatGPT для помощи в принятии решений на этапах ЖЦ разработки программного обеспечения.

Задачи:

1. Обоснование актуальности темы.

2. Исследование влияния ChatGPT на процессы ЖЦ научного исследования.
3. Исследование влияния ChatGPT на процессы ЖЦ разработки ПО.
4. Формирование алгоритма взаимодействия с ChatGPT.
5. Оценка экономической эффективности.

Объект исследования - модель ChatGPT и ее применение для поддержки принятия решения в научных работах и IT-компаниях.

Предмет исследования - алгоритм применения модели ChatGPT для поддержки принятия решения в ЖЦ ИТ-инновации.

В процессе работы были использованы следующие методики исследования:

- Экспериментальные исследования: Этот метод включает манипуляцию и контроль переменных в контролируемой среде с целью изучения причинно-следственных связей и получения объективных данных. Он широко используется в научных исследованиях, особенно в физике, химии, биологии и психологии.
- Кейс-исследования: Этот метод предполагает детальное изучение конкретного случая или явления с помощью различных источников информации, таких как документы, архивы, интервью и наблюдения. Кейс-исследования позволяют получить глубокое понимание отдельных случаев и используются в социальных науках, медицине, бизнесе и других областях.
- Мета-анализ: Этот метод объединяет и анализирует результаты нескольких независимых исследований с целью выявления общих закономерностей или эффектов. Мета-анализ используется в медицине,

психологии и других областях для суммирования и оценки существующих исследований.

Научная новизна работы заключается в предложенном алгоритме работы с таким инструментом как ChatGPT, исследовании и анализе применения модели ChatGPT для помощи в работе над научными проектами и оптимизации процессов в IT-компаниях.

Практическим результатом работы является алгоритм применения ChatGPT для наиболее эффективного решения поставленных задач, с уточнением конкретных входных данных для решения рассматриваемых задач.

Экономической эффективностью предлагаемых в диссертации мер является получение прибыли компании за счет сокращения времени затрачиваемого в процессах жизненного цикла тестирования программного обеспечения и технической поддержки пользователей.

Эмпирическая база исследования: федеральный государственный стандарт высшего образования, ГОСТы, электронные источники, официальный сайт компании НЧОУ ДПО «УЭЦ «Строитель».

Работа состоит из трех глав: Обзор и анализ представленных на рынке информационных систем оптимизации бизнес-процессов компании; Модель процесса внедрения выбранной информационной системы в работу организаций, завершающих этап роста и переходящих в этап зрелость; Оптимизация бизнес-процессов компании путем внедрения информационной системы. Первая глава состоит из двух основных подпунктов: Формирование требований к подбору и сравнению информационных систем; Анализ представленных на рынке информационных систем.

Вторая глава включает в себя: Применение информационной системы для оптимизации бизнес-процессов предприятия малого и среднего бизнеса;

Разработка модели процесса внедрения информационной системы в работу организации.

Третья глава имеет следующую структуру: Составление модели архитектуры предприятия НЧОУ ДПО «УЭЦ «Строитель»; Планирование проекта; Экономическое обоснование внедрения проекта; Внедрение разработанного проекта в работу организации.

1. ПРИНЯТИЕ РЕШЕНИЙ ПРИ УПРАВЛЕНИИ ЖИЗНЕННЫМ ЦИКЛОМ ИННОВАЦИИ

1.1 Анализ библиометрических показателей области исследования

В процессе выполнения дипломной работы был проведен анализ литературы с использованием инструмента SciVal. Для определения сферы поиска работ были использованы следующие фразы: "Методы принятия решений" И ("Проблема принятия решений" ИЛИ "жизненный цикл инновации" ИЛИ "инновация" ИЛИ "принятие решений"). Общее количество работ составило 853, при этом с каждым годом количество работ по данной тематике только увеличивалось, что можно заметить, посмотрев на график изменчивости данного показателя на рисунке 1.

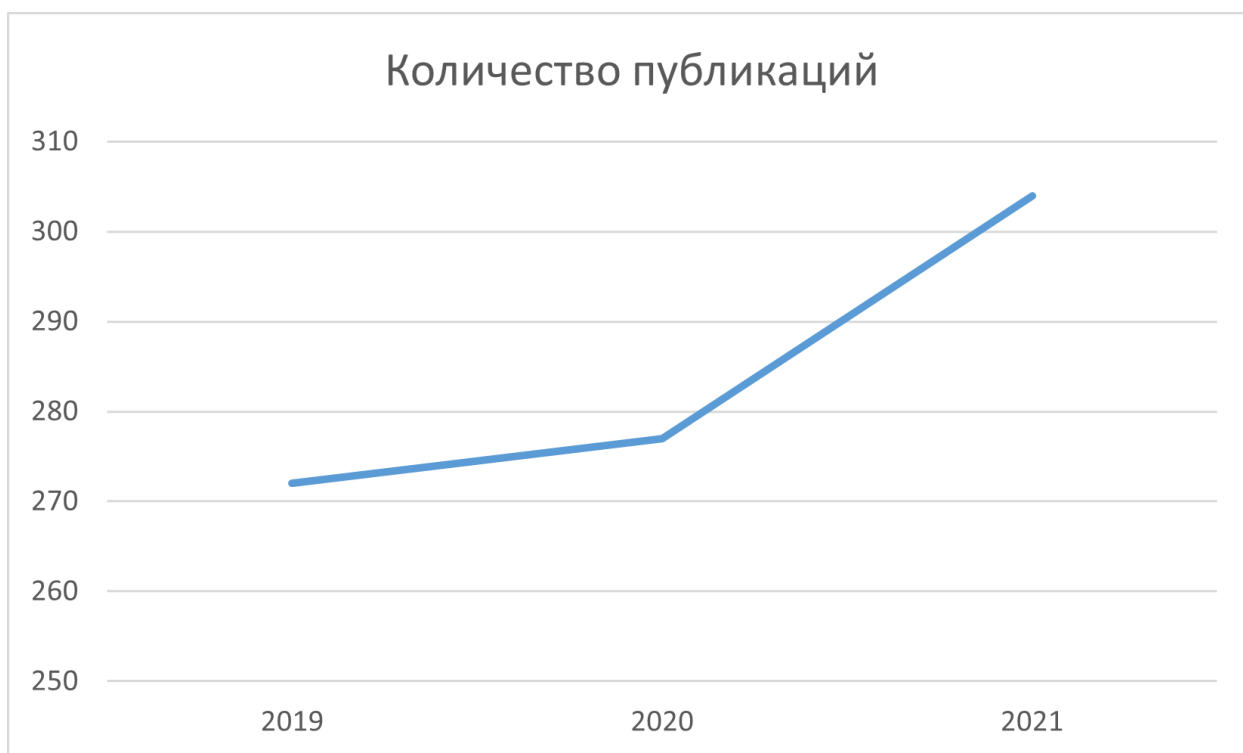


Рисунок 1 – Количество публикаций

В процессе анализа литературы были выделены наиболее часто встречающиеся слова в исследованиях, связанных с принятием решений, облако ключевых слов представлено на рисунке 2. Наиболее популярные из них были следующие:

1. Многокритериальное принятие решений.
2. Принятие решений.
3. Групповое принятие решений.
4. Многокритериальное принятие решений.



Рисунок 2 – Облако ключевых слов

Эти ключевые слова указывают на актуальность и широкое использование методов принятия решений в исследованиях, связанных с жизненным циклом инновации. Анализ литературы по данной теме позволил выявить существующие подходы, тенденции и направления исследований в области принятия решений на различных этапах жизненного цикла инноваций.

Влияние цитирования, взвешенное по полю исследований, график изменчивости которого представлен на рисунке 3, равен 1,38, что говорит о том, что данный показатель выше среднего для сравнимых публикаций на 38%.



Рисунок 3 – Влияние цитирования, взвешенное по полю исследований

Также следует рассмотреть другой аспект, характеризующий актуальность изучения выбранной темы – показатель международного сотрудничества. График изменчивости данного показателя представлен на рисунке 4. В данной области 143 научные публикации имеют международное авторство, что составляет 16,7% от всех работ.



Рисунок 4 – Показатель международного авторства

Количество просмотров публикаций в выбранной тематической области составляет более 27 тысяч. В то же время показатель цитирования показывает, как количество цитирований работ организации сравнивается со средним количеством цитирований других подобных работ в общей базе данных. В выбранной области этот показатель превышает 1,00 (10,532), что указывает на то, что наши работы получают больше цитирований, чем средние работы во всем мире.

1.2 Обоснование актуальности выбранной темы

Исследование научных статей и исследований привело к выделению следующих основных концепций: теория рационального выбора, теория экологической рациональности и поведенческая экономика.

Теория рационального выбора рассматривает достижение оптимальных результатов путем использования доступных ресурсов и принятия практических решений [1]. Она определяет "рациональность" как стремление к максимальной полезности. Рациональный выбор включает вычисление и максимизацию ожидаемой ценности [2], что обеспечивает объективность, учет входящих данных и достижение оптимального результата. Процесс выбора должен основываться на систематическом обосновании [3], мотивированных утверждений [4] и теории вероятности [5].

Рациональное принятие решений в ситуациях, связанных с возможностью риска, основывается на оценках вероятности различных сценариев развития событий. Эти вероятности могут быть определены на основе статистических данных или субъективных суждений, которые могут снизить вероятность выбора наилучшего варианта.

Принимая во внимание трудность оценки разумного и правильного выбора, а также влияние субъективных предпочтений и ценностей на принятие решения, исследователи подчеркивают важность применения принципов теории вероятности при принятии решений. Оценка вероятности

развития ситуации позволяет определить разумность выбора и учесть альтернативные варианты, чтобы достичь максимальной полезности.

В 1970-е годы исследователи Д. Канеман, А. Тверски и Р. Талер разработали поведенческую экономику. Их работа показала, что люди часто делают ошибки при принятии финансовых решений и эти ошибки распространены [6]. Исследование позволило ученым систематизировать и описать когнитивные искажения, а также дать рекомендации по их обнаружению и преодолению. Одной из ключевых особенностей поведенческой концепции является то, что большинство людей, включая экономистов, действуют подсознательно, а не рационально [7].

Согласно принципам поведенческой экономики, для понимания функционирования экономики и эффективного ее управления необходимо уделить внимание иррациональным основам, лежащим в основе идей и эмоций человека.

Исследования в области поведенческой экономики демонстрируют, что человек часто ведет себя по-другому, чем предполагает стандартная концепция разумного выбора [8]. Ученые уверены в том, что ошибки в человеческом поведении могут быть предсказуемыми - в определенных условиях люди действуют иррационально, противореча своим долгосрочным интересам, из-за когнитивных искажений, свойственных человеку. Следующие искажения были выявлены и описаны: "эффект фрейминга", когда выбор рассматривается с точки зрения выигрыша или проигрыша; "избегание потерь", когда "потери" от определенного решения воспринимаются более значимыми, чем "объективно равные" приобретения; "эффект обладания", который показывает, что вещи, которыми мы уже обладаем, имеют большую ценность для нас, чем потенциальные приобретения, и другие [9].

Следовательно, исследования в области поведенческой экономики показывают, что изменение условий, понимание когнитивных искажений, осознанность при оценке ситуации, соблюдение логических законов и

основных принципов теории вероятности могут влиять на реакции индивидуума при принятии решений.

Теория экологической рациональности, сформулированная Д. Голдстейном и Г. Гигеренцером в конце XX века, описывает рациональность как неотъемлемую часть практического поведения людей, когда они взаимодействуют с тремя основными факторами: мотивацией, информацией о способах достижения целей [10]. В контексте этой теории рациональность рассматривается в связи с окружающей средой, доступными ресурсами, а также знаниями и навыками, которыми обладает индивидуум и которые определены культурой и особенностями общества на определенном этапе развития [11].

Поведение, которое приводит к успеху в определенной организационной среде, рассматривается как рациональное. Решения принимаются на основе простых правил, которые соответствуют условиям окружающей среды [12]. Экологическая рациональность применяется в разных областях, таких как экономика, политика [13] и управление ресурсами. Коллективное управление является гибким, изменчивым и развивающимся, и зависит от различных факторов [14]. Применение адаптивных инструментов позволяет принимать решения на основе ограниченной информации или вычислений, что подтверждает эффективность простых стратегий, основанных на устоявшихся закономерностях.

Изучение различных теорий, таких как рациональный выбор [15] и экологическая рациональность [16], позволяет нам определить важные элементы, необходимые для принятия наилучших решений. Можно выделить определенные элементы, которые встречаются в этих теориях и должны быть учтены при принятии решений. Некоторые из таких элементов включают применение принципов вероятности для определения возможных сценариев, понимание когнитивных искажений для тщательной оценки ситуации и осознанного принятия решений, а также использование эвристик в сочетании с институциональной средой для опоры на интеллект и практический опыт.

Теории принятия решений, основанные на различных представлениях о том, что считается разумным и рациональным поведением человека, исследуют сознание индивида и коллективный разум с целью предоставить людям знания и инструменты, которые помогут им принимать выбор в ситуациях, связанных с риском и неопределенностью. В данной работе были изучены основные иностранные теории, чтобы выявить их общие элементы. Основываясь на значимых положениях, обобщениях и пересечениях этих теорий, можно выделить несколько важных элементов, которые следует учитывать при принятии решений, такие как принципы теории вероятности, понимание когнитивных искажений, присущих каждому человеку, и знание институциональной среды, которая определяется культурой и особенностями общества.

1.3 Жизненный цикл инновации

С развитием инновационных направлений деятельности возникло понятие "жизненный цикл инноваций". Однако до сих пор не существует единого определения этого понятия. В связи с этим актуальным представляется провести сравнительный анализ современных подходов к инновационным циклам. В научной литературе можно выделить три основных подхода к определению этого понятия.

1. Подход, основанный на продукте. В данном случае инновация рассматривается как новый продукт, поэтому жизненный цикл инноваций включает те же основные стадии, что и у нового продукта [17]. Они включают:

Стадия 1. Разработка нового продукта. Здесь не указывается, кем и как будет разработан продукт, а также в какие сроки и на какие средства.

Стадия 2. Выход на рынок. Не конкретизируется, каким образом продукт будет представлен на рынке, на каком рынке и в каком объеме. Однако здесь требуется использование традиционного маркетинга, так как продукт является новым и неизвестным на рынке.

Стадия 3. Развитие рынка. Здесь подразумевается, что новый продукт постепенно достигнет максимального или значительного уровня продаж, хотя не совсем ясно, почему это гарантируется. Возможно, продукт не будет успешным на рынке и не получит должного признания, что будет свидетельствовать о неудаче или

Стадия 4. Устойчивость рынка. Обычно эта стадия наступает практически всегда после предыдущей, так как рынок становится насыщенным в результате цикличности.

Стадия 5. Сокращение рынка. Это также логичная стадия, обусловленная научно-техническим прогрессом и появлением новых инноваций, таких как улучшенные товары-аналоги или заменители.

Стадия 6. Снижение рынка. Неизбежная стадия для физически или морально устаревших продуктов, которые перестали быть инновационными и были вытеснены с рынка более успешными конкурирующими продуктами.

Критикуя данный подход, стоит отметить, что начальные стадии жизненного цикла являются наиболее слабыми. Перед разработкой нового продукта не предусмотрены этапы предварительного исследования, которые оценивают позитивность самой инновации и готовность потенциального рынка принять ее. Можно привести множество примеров продуктовых новшеств, которые так и не были внедрены. Например, проект E-мобиля был приостановлен после нескольких лет неудачных попыток (возможно, навсегда).

2. Процессный подход представляет собой разделение жизненного цикла инноваций на отдельные этапы, связанные с процессами инновационной деятельности. В отличие от продуктового подхода, здесь наблюдаются значимые улучшения, связанные с более детальным рассмотрением ранних стадий цикла. Обычно выделяются следующие этапы.

Этап 1: Генерация идей. Идеи, применимые в инновационных процессах, возникают не случайно, а как результат фундаментальных или прикладных исследований. Необходимо учитывать, что на самом раннем этапе юридической чистоты инновационной идеи важно учитывать существующие институты авторского права и связанные с ними права.

Этап 2: Фильтрация идей. После генерации инновационной идеи она должна пройти этап отбора, где она должна показать свою перспективность в организационно-технологическом и маркетингово-рыночном смыслах. Некоторые авторы отрицают необходимость этого этапа и сразу переходят к разработке, однако он является важным. Критерии отбора (фильтры) могут включать различные экономические показатели

Этап 3. Защита авторских прав [18] является неотъемлемой частью функционирования и развития инноваций в глобализированной экономике.

Пренебрежение этим этапом может привести к потере инновационной деятельности на последующих этапах.

Этап 4. Оценка эффективности будущего проекта [19] обычно проводится, хотя достоверность таких оценок может быть сложна, поскольку методики оценки не всегда учитывают особенности инноваций.

Эти этапы могут быть объединены в одну стадию, известную как «предпроектные работы». Отметим, что в продуктовом подходе такая стадия просто отсутствует.

Этап 5. Разработка нового продукта. Обычно этот этап совпадает с аналогичным этапом при применении продуктового подхода. Однако здесь могут быть добавлены некоторые дополнительные этапы, такие как испытания нового продукта, оформление его как товара (включая товарный знак, торговую марку) [20] и другие.

Этап 6. Организация серийного производства. Хотя этот этап не применим ко всем инновациям, он играет важную роль на насыщенных рынках инновационной продукции.

Этап 7. Коммерциализация инноваций или вывод инновационной продукции на рынок. Следует отметить, что успех всего жизненного цикла во многом зависит от успешной коммерциализации инноваций. Рынок является основным судьей положительности инновации.

Обычно сторонники процессного подхода ограничиваются рассмотрением жизненного цикла инновации только до ее выхода на рынок. Однако сторонники продуктового подхода имеют преимущество, так как они включают в жизненный цикл инновации и последующую судьбу инновационного продукта. Кажется разумным сочетание указанных подходов, особенно в случаях, когда необходимо усилить или выделить маркетинговые этапы.

3. Маркетинговый подход является наиболее современным методом и учитывает глобализацию экономических процессов, что приводит к усилению конкуренции между инновационными решениями. Однако данный подход

применим в основном к жизненному циклу коммерческих инноваций, ориентированных на достижение успеха на рынке. В российской научной среде сторонниками маркетингового подхода являются такие исследователи, как В.Д. Маркова, С.А. Кузнецова, Н.П. Кетова и Т.Л. Короткова [21].

1. На первом этапе, который называется предварительным маркетингом, проводится исследование маркетинговых возможностей для реализации будущей инновации. Это исследование охватывает пять основных аспектов: кадровый, технологический, информационный, организационный и финансовый, необходимых для создания инновационного продукта. Иными словами, в начале процесса необходимо провести анализ этих пяти основных составляющих успешной инновации.



Рисунок 5 – Основные направления маркетингового анализа [21]

Как показывает иллюстрация, на данном этапе необходимо провести анализ всех аспектов будущего инновационного процесса, включая анализ рынка. Некоторые авторы рекомендуют на данной стадии формировать фокус-группы потребителей инновационного продукта и оценивать рыночный потенциал.

Этот этап также отличается необходимостью финансирования маркетинговых исследований еще до начала разработки инновационного проекта. Финансирование должно быть предварительно рассчитано и обеспечено, а также должны быть созданы организационные условия для проведения данной фазы маркетинга внутри организации, планирующей воплощение данной идеи.

Важно отметить, что негативная оценка будущей инновации не должна рассматриваться как отрицательный результат, поскольку отказ от "плохой" инновации помогает избежать неоправданных затрат на более поздних этапах. Коммерческим инновациям этот этап жизненного цикла просто необходим.

2. "Маркетинг возможностей инновационного решения" – при разработке и создании инноваций, ориентированных на рынок, необходимо заранее оценить их реализуемость, окупаемость и экономическую эффективность. Чтобы достичь этой цели, важно на этапе разработки более четко определить целевую аудиторию инновации, размер рынка потребителей, их платежеспособность, возможности конкурентов (существующих и потенциальных), а также решить возможные юридические проблемы. Для этой задачи подходят пробный маркетинг [22], экспертные методы и инструменты системного анализа, такие как математическое или имитационное моделирование. Выбор маркетинговых средств и методов зависит от специфики инновационного проекта. Важно также заранее оценить социальные последствия введения будущей инновации на рынок [23].

3. "Маркетинг рынка" – разработанный инновационный продукт нуждается в продвижении на рынок. На этом этапе маркетинг выполняет свои традиционные функции: поиск клиентов и/или посредников, анализ ценообразования и разработка оптимальной ценовой стратегии, реклама и PR, включая мероприятия для формирования имиджа и другие. Целью этого этапа в случае успешной инновации является полноценная продажа инновационного продукта.

4. "Оценка результатов, корректировка рыночной стратегии и поддержка сбыта" - предусматривает всесторонний анализ эффективности реализации инновационного проекта в целом. На основе такого анализа принимаются обоснованные решения по корректировке маркетинговой стратегии, улучшению или даже изменению самого инновационного продукта.

Этот этап должен продолжаться до тех пор, пока инновация перестает быть инновацией (частично совпадая с подходом, основанным на продукте).

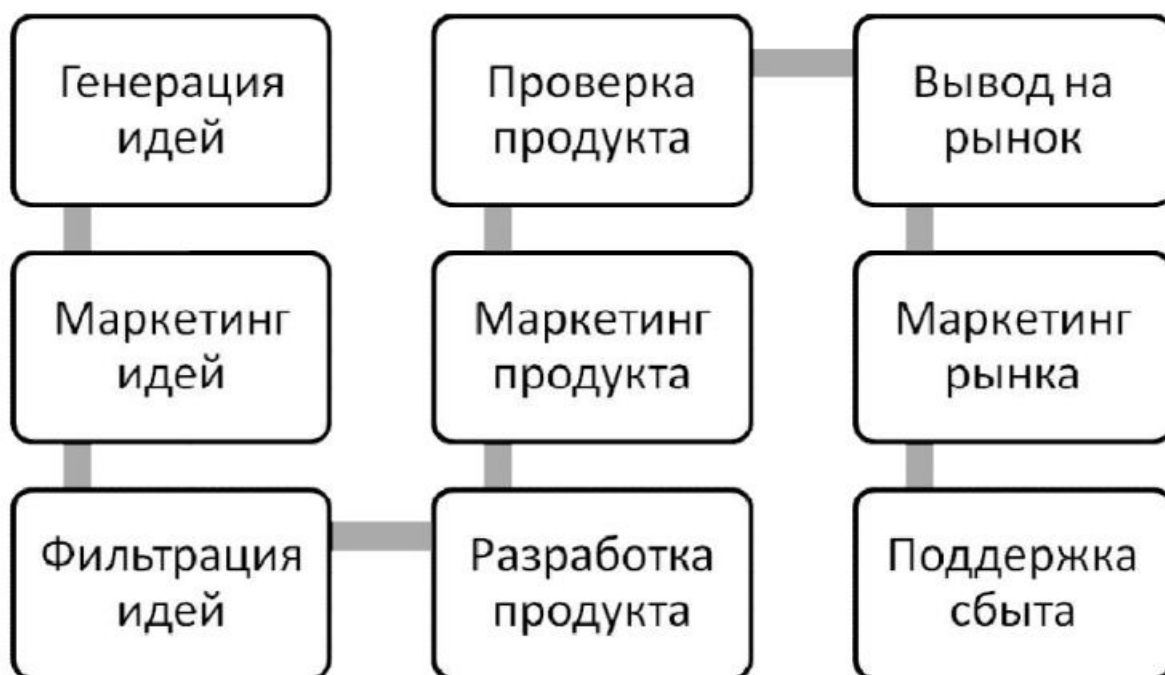


Рисунок 6 – ЖЦ инновации [23]

1.4 Принятие решений на этапах научно-исследовательской работы

Жизненный цикл научно-исследовательского проекта (ЖЦ НИР) представляет собой последовательность этапов, которые проходит проект, начиная от формулирования проблемы и постановки задачи и заканчивая реализацией решений и контролем. Опишу каждый из этих этапов и основные цели и задачи, связанные с ними:

1. Формулирование проблемы и постановка задачи:

- а) цель: четкое определение проблемы и установление целей и задач проекта;

б) задачи:

- 1) определение основной проблемы, которую необходимо исследовать или решить;
- 2) формулирование целей и подзадач, которые должны быть достигнуты для решения проблемы;
- 3) определение критериев успеха, которые помогут оценить достижение поставленных целей.

2. Планирование и проектирование проекта:

а) цель: разработка детального плана исследования и определение ресурсов, необходимых для его выполнения;

б) задачи:

- 1) определение методологии исследования и выбор соответствующих инструментов и техник;
- 2) распределение ресурсов, таких как бюджет, время и персонал, для обеспечения успешной реализации проекта;
- 3) определение ролей и ответственностей участников команды исследования.

3. Сбор и анализ данных:

а) цель: получение и анализирование данных, необходимых для ответа на исследовательские вопросы и достижения поставленных целей;

б) задачи:

- 1) разработка методов исследования и выбор подходящих инструментов для сбора данных;
- 2) сбор и систематизация данных с помощью определенных методов и техник;
- 3) анализ полученных данных с использованием соответствующих статистических и аналитических методов.

4. Интерпретация результатов и выводы:

а) цель: интерпретация и анализ полученных результатов с целью выявления закономерностей и деловых выводов;

б) задачи:

- 1) анализ полученных данных и результатов исследования;
- 2) интерпретация результатов с учетом поставленных целей и задач;
- 3) формулирование выводов и рекомендаций, основанных на полученных результатах.

5. Реализация решений и контроль:

а) цель: реализация предложенных решений на основе результатов исследования и контроль их эффективности;

б) задачи:

- 1) разработка плана реализации решений и определение необходимых действий;
- 2) осуществление реализации решений с учетом поставленных целей и задач;
- 3) контроль и оценка результатов реализации и эффективности принятых решений.

Цели и задачи на каждом этапе ЖЦ НИР направлены на обеспечение успешной реализации проекта, получение релевантных данных, анализ результатов и принятие обоснованных решений.

Жизненный цикл научно-исследовательского проекта (ЖЦ НИР) включает в себя несколько этапов, на каждом из которых могут возникать проблемы в принятии решений. Рассмотрим основные этапы ЖЦ НИР и связанные с ними потенциальные проблемы принятия решений:

1. Формулирование проблемы и постановка задачи:

- недостаточное понимание или нечеткость поставленной задачи может затруднить определение конкретных проблем, которые требуют решения;
- недостаточный анализ существующих данных и литературы может привести к неполному определению проблемы и, следовательно, неправильному выбору решений.

2. Планирование и проектирование исследования:

- недостаточная оценка ресурсов (бюджета, времени, персонала и т.д.) может привести к неправильному распределению ресурсов для принятия решений;
- неправильная организация исследовательской команды или нечеткое определение ролей и обязанностей может создать проблемы с координацией и принятием решений.

3. Сбор и анализ данных:

- некачественные или неправильно собранные данные могут привести к неверным выводам и, соответственно, к неправильному принятию решений;
- недостаточная обработка и анализ данных может привести к упущению важной информации, которая является основой для принятия решений.

4. Интерпретация результатов и выводы:

- неправильная интерпретация данных и результатов исследования может привести к ошибочным выводам и неправильному принятию решений;
- недостаточное обоснование и объяснение результатов может затруднить принятие решений на основе этих результатов.

5. Реализация решений и контроль:

- неправильное планирование и организация реализации решений может привести к их неэффективной реализации;
- недостаточный контроль и оценка результатов реализации могут привести к неправильным выводам о эффективности принятых решений.

В целом, проблемы в принятии решений в ЖЦ НИР могут быть связаны с нечетким определением проблемы, недостаточным анализом данных, неправильной интерпретацией результатов и неправильной организацией

реализации решений. Для успешного преодоления этих проблем необходимо уделить должное внимание каждому этапу ЖЦ НИР, проводить тщательный анализ и обеспечивать четкую коммуникацию и координацию между участниками проекта.

1.5 Принятие решений на этапах жизненного цикла разработки

Жизненный цикл разработки программного обеспечения (ЖЦ РПО) представляет собой последовательность этапов, которые проходит проект разработки ПО, начиная от идеи и заканчивая выпуском и поддержкой готового продукта. Опишу каждый из этих этапов и основные цели и задачи, связанные с ними:

1. Анализ и сбор требований:

- а) цель: понимание потребностей и ожиданий заказчика, а также определение функциональных и нефункциональных требований к ПО;
- б) задачи:
 - 1) взаимодействие с заказчиком для выявления и анализа его требований;
 - 2) формализация требований в виде спецификаций и документации.

2. Проектирование:

- а) цель: разработка архитектуры и дизайна ПО, определение структуры и компонентов системы;
- б) задачи:
 - 1) определение структуры и модулей ПО, включая взаимосвязи и интерфейсы;
 - 2) проектирование пользовательского интерфейса;
 - 3) выбор технологий и инструментов разработки.

3. Реализация:

- а) цель: фактическая разработка и программирование ПО на основе требований и дизайна;

б) задачи:

- 1) написание и отладка программного кода;
- 2) создание модулей и компонентов, реализация функциональности;
- 3) тестирование и отладка разработанного кода.

4. Тестирование:

а) цель: проверка качества и соответствия разработанного ПО требованиям и ожиданиям;

б) задачи:

- 1) разработка тестовых сценариев и планов;
- 2) выполнение функционального, интеграционного, системного и других видов тестирования;
- 3) выявление и устранение ошибок и дефектов.

5. Внедрение и поддержка:

а) цель: установка, внедрение и поддержка готового ПО в рабочей среде;

б) задачи:

- 1) установка и настройка ПО на целевых системах;
- 2) обучение пользователей работе с новым ПО;
- 3) поддержка и обновление ПО в соответствии с потребностями и требованиями заказчика.

Цели и задачи на каждом этапе ЖЦ РПО направлены на обеспечение успешной разработки, тестирования, внедрения и поддержки программного обеспечения, соответствующего требованиям заказчика и ожиданиям пользователей.

Рассмотрим подробнее ЖЦ тестирования ПО. При анализе ЖЦ тестирования ПО можно выявить следующие потенциальные проблемы, связанные с принятием решений:

1. Недостаточное планирование тестирования: Неправильное или недостаточное планирование тестовых активностей может привести к

пропуску важных тестовых сценариев или недостаточному покрытию функциональности, что затрудняет принятие решений о готовности ПО к выпуску.

2. **Неправильный выбор тестовых критериев:** Неправильное определение критериев приемочного тестирования или недостаточное внимание к ключевым аспектам функциональности или производительности ПО может привести к неполной оценке качества и затруднить принятие решений о дальнейшем развитии или выпуске продукта.
3. **Недостаточная автоматизация тестирования:** Недостаточное использование автоматизации тестирования может привести к неполному покрытию тестами и затруднить принятие решений о стабильности и качестве ПО, особенно при частых релизах и больших объемах кода.
4. **Недостаточное взаимодействие с разработчиками:** Отсутствие эффективного обмена информацией и обратной связи между тестировщиками и разработчиками может привести к затягиванию решений по исправлению дефектов или неправильному пониманию тестовых результатов.
5. **Отсутствие рационального подхода к приоритезации:** Неправильное определение приоритетов тестовых активностей и дефектов может привести к затягиванию выпуска ПО или пропуску критических проблем, затрудняющих принятие решений о дальнейшем развитии и выпуске продукта.

Для успешного преодоления этих проблем необходимо уделять должное внимание планированию и организации тестирования, улучшать коммуникацию между тестировщиками и разработчиками, активно использовать автоматизацию тестирования и осуществлять рациональную приоритезацию задач и дефектов. Кроме того, важно применять тестирование на ранних этапах разработки и следовать agileвским методологиям, чтобы обеспечить гибкость и своевременность принятия решений.

1.6 Обоснование значимости использования искусственного интеллекта и на примере ChatGPT

Системы искусственного интеллекта (ИИ) разработаны для выполнения разнообразных "умных" задач. В конечном счете, их применение в реальном мире имеет две основные цели: помощь людям в принятии решений или в некоторых случаях замещение их функций. Искусственный интеллект не является ни сверхъестественным, ни невероятным. В соответствии с классическим определением, это просто способность компьютера к обучению, которое дает ему знания, чтобы принимать решения, аналогичные решениям, принимаемым человеческим интеллектом. Вкратце, ИИ — это способность компьютера "думать" и "размышлять". Какие задачи можно делегировать машинному интеллекту и в каких областях эта сила будет расти в следующем десятилетии?

Искусственный интеллект представляет собой не только технологию в области информационных технологий, но и совокупность различных научных дисциплин [24]. Его основная задача заключается в моделировании устройства человеческого интеллекта. Методы искусственного интеллекта (также известные как алгоритмы) представляют собой "пути обучения", которые искусственный интеллект должен пройти для достижения своих целей [25,26].

Искусственный интеллект может быть классифицирован следующим образом [27]:

1. Слабый искусственный интеллект: это означает, что искусственный интеллект способен выполнять ограниченные задачи и представлен в виде мощной программы.
2. Сильный искусственный интеллект: в настоящее время этот тип искусственного интеллекта находится в разработке. Он основан на так называемой "мыслительной способности" и способен принимать решения по "необученным" задачам. Например, некоторые роботы-операторы колл-

центров приближаются к концепции "сильного" искусственного интеллекта.

3. Суперинтеллект: Создание суперинтеллекта в настоящее время не имеет определенных предпосылок, и у нас есть только приблизительное представление о том, как он должен быть. Например, одна из точек зрения предполагает, что суперинтеллект должен быть максимально близким к человеческому интеллекту, а может быть, даже превосходить его.

По мере увеличения компьютерной автоматизации и уменьшения человеческого вмешательства, мы будем передавать автоматизированным системам множество областей принятия решений, что может привести к потере определенных навыков.

Примером использования искусственного интеллекта в медицине [28,29] может служить система под названием "Онкопомощь", разработанная компаниями "ТехЛАБ" и "Нетрика". В начале 2022 года она была успешно внедрена в Ленинградской области и предназначена для эффективной организации лечения онкологических пациентов, а также контроля состояния пациентов, объемов и результатов лечения.

Также стоит упомянуть ИИ MakerSights [30, 31], который применяется в розничной торговле и помогает в принятии обоснованных решений на протяжении всего процесса создания и внедрения продукта на рынок.

Существует множество технологий и методов искусственного интеллекта и автоматизации, которые применяются в управлении жизненным циклом продукции. Некоторые из этих технологий включают:

- 1) использование искусственного интеллекта и машинного обучения.
- 2) применение интернета вещей, позволяющего собирать данные на этапе производства и обмениваться информацией с другими компонентами системы;
- 3) применение аналитики больших данных для обработки и анализа больших объемов информации.

Анализ данных играет важную роль в управлении жизненным циклом продукции, так как позволяет определить потребности потребителей, прогнозировать спрос на продукцию и повышать ее качество [32]. С использованием алгоритмов машинного обучения и искусственного интеллекта компании могут автоматически обрабатывать большие объемы данных и извлекать ценную информацию из них. Технология интернета вещей (IoT) [33] позволяет оперативно реагировать на изменения в законодательстве, потребительском спросе и других факторах, которые могут повлиять на производство.

Большие объемы данных, собранные на различных этапах жизненного цикла продукции, могут быть использованы для прогнозирования спроса [34], определения потребностей пользователей и улучшения качества продукции. Аналитические инструменты, такие как машинное обучение, способны проанализировать эти данные и выявить тенденции. Анализ данных и применение искусственного интеллекта помогают прогнозировать продажи [35], контролировать качество продукции, управлять складскими запасами, определять потребности пользователей [36] и многое другое. Благодаря этому компании могут оптимизировать производственные процессы, снизить издержки и повысить эффективность.

Примером применения искусственного интеллекта в решениях является автоматическое прогнозирование спроса на продукцию. Это осуществляется на основе анализа данных о предыдущих продажах, погодных условиях, ценовой политике и других факторах. Такой подход помогает компаниям оптимизировать производство и избежать излишней или недостаточной производственной мощности [37].

Еще одним примером применения искусственного интеллекта является оптимизация процессов контроля качества продукции. Благодаря использованию компьютерного зрения и машинного обучения, компании могут автоматически сканировать продукцию и определять ее качество. Это

позволяет более быстро обнаруживать дефекты и улучшать производственные процессы.

В современном информационном обществе принятие эффективных решений становится все более сложной задачей. Быстрое развитие технологий и обилие доступной информации создают необходимость в инструментах, способных помочь людям принимать обоснованные и информированные решения. В рамках этой работы исследуется применение ChatGPT - языковой модели, основанной на искусственном интеллекте, в качестве инструмента поддержки принятия решений.

ChatGPT - это высокоэффективная и многофункциональная языковая модель, разработанная с использованием глубокого обучения. Она основана на архитектуре GPT (Generative Pre-trained Transformer), которая позволяет модели генерировать тексты, отвечая на заданные вопросы и участвуя в диалоге с пользователями. ChatGPT обладает уникальными особенностями, которые делают его привлекательным инструментом для поддержки принятия решений.

Во-первых, ChatGPT обладает способностью быстро анализировать большие объемы информации и выделять важные аспекты, что позволяет предоставлять пользователю комплексные и информативные ответы. Благодаря своей способности к обучению на больших корпусах текстов, модель обладает широким кругом знаний и способна предоставлять детальные исследования и объяснения, необходимые для принятия обоснованных решений.

Во-вторых, ChatGPT обладает способностью к контекстному пониманию, что позволяет ему участвовать в диалоге с пользователями и задавать уточняющие вопросы для получения более точной информации. Это особенно полезно при поддержке принятия решений, поскольку модель может адаптироваться к конкретным потребностям пользователя и предоставлять индивидуальные рекомендации и советы.

Использование ChatGPT в качестве инструмента поддержки принятия решений обладает несколькими значимыми преимуществами. Во-первых, модель способна обрабатывать информацию значительно быстрее, чем человек, и проводить анализ большого объема данных. Это позволяет существенно сократить время, затрачиваемое на поиск и обработку информации, и сосредоточиться на сути проблемы.

Во-вторых, ChatGPT является гибким инструментом, который может быть адаптирован для различных сфер применения. Он может быть использован в бизнесе для оптимизации процессов, в медицине для поддержки врачей в принятии диагнозов, в образовании для предоставления обучающих материалов и рекомендаций студентам, и во многих других областях. Это делает его универсальным инструментом для поддержки принятия решений в различных контекстах.

В свете сложности современных задач принятия решений, применение ChatGPT как инструмента поддержки принятия решений обладает значительным потенциалом. Его уникальные особенности, такие как способность анализировать большие объемы информации, контекстное понимание и гибкость в применении, делают его эффективным и перспективным инструментом для помощи людям в принятии обоснованных и информированных решений. Исследование применения ChatGPT в этой области имеет значимость для дальнейшего развития и применения искусственного интеллекта в поддержке принятия решений.

1.7 Особенности языковых моделей

В области искусственного интеллекта существует сложная проблема — разработка машины, способной понимать человеческий язык. Например, при поиске "близлежащих итальянских ресторанов" в поисковой системе необходимо, чтобы алгоритм анализировал каждое слово в запросе и выводил соответствующие результаты. Приложение для перевода должно учитывать

контекст определенного слова на английском языке и различия в грамматике между языками.

Все эти задачи относятся к области информатики, известной как Обработка естественного языка или NLP. Достижения в NLP привели к широкому спектру практических приложений, включая виртуальных помощников, например, спам-фильтры, обнаруживающие вредоносную электронную почту.

Недавно в NLP был сделан значительный прорыв с появлением больших языковых моделей (LLM). Модели LLM, такие как GPT-3, стали невероятно мощными, и кажется, что они успешно справляются с любыми задачами в области NLP.

Обработка естественного языка (NLP) представляет собой технологию машинного обучения, которая позволяет компьютерам интерпретировать, манипулировать и понимать человеческий язык [38]. Современные организации сталкиваются с огромными объемами голосовых и текстовых данных, поступающих из различных коммуникационных каналов, таких как электронная почта, текстовые сообщения, новостные ленты социальных сетей, видео и аудиозаписи. Они применяют технологию NLP для автоматической обработки этих данных, анализа намерений или настроений в сообщениях и реагирования на человеческую коммуникацию в режиме реального времени.

Обработка естественного языка играет решающую роль в эффективном анализе текстовых и речевых данных. Этот подход позволяет преодолеть различия в диалектах, сленге и грамматических нарушениях, которые часто встречаются в повседневных разговорах. Компании активно применяют этот метод для автоматизации нескольких задач, включая:

- обработку, анализ и архивирование больших объемов документов;
- анализ отзывов клиентов или записей колл-центра [39];
- запуск чат-ботов для автоматического обслуживания клиентов;
- классификацию и извлечение текста.

Также возможно интегрирование NLP в клиентско-ориентированные приложения для более эффективного взаимодействия с клиентами. Например, чат-бот может анализировать и сортировать запросы клиентов, автоматически отвечать на распространенные вопросы и перенаправлять сложные запросы в службу поддержки. Такая автоматизация помогает снизить затраты, освобождает сотрудников от необходимости тратить время на рутинные запросы и повышает уровень удовлетворенности клиентов.

Множество компаний применяют программное обеспечение и инструменты обработки естественного языка (NLP) для повышения эффективности, точности и оптимизации своих операций. Ниже приведены несколько примеров использования таких инструментов.

Защита конфиденциальных данных. Компании в страховом, юридическом и медицинском секторах обрабатывают большие объемы конфиденциальных документов, таких как медицинские карты, финансовые данные и личная информация. Вместо того чтобы выполнять ручную проверку, компании применяют технологию NLP для автоматического редактирования личных данных и обеспечения защиты конфиденциальности.

Улучшение взаимодействия с клиентами. С использованием технологий NLP чат-боты и голосовые боты становятся более "человеческими" в общении с клиентами [40]. Компании используют чат-ботов для расширения возможностей обслуживания клиентов и повышения его качества при минимальных затратах на эксплуатацию.

Анализ бизнес-процессов. Маркетологи применяют инструменты NLP для получения информированного представления о том, как клиенты относятся к продуктам или услугам компании. Путем анализа определенных фраз они могут оценить настроение и эмоции клиента, выраженные в письменных отзывах.

Обработка естественного языка объединяет компьютерную лингвистику, машинное обучение и модели глубокого обучения для работы с человеческим языком.

Компьютерная лингвистика – это наука, которая исследует и строит модели человеческого языка с использованием компьютеров и программных инструментов [41]. С помощью методов компьютерной лингвистики, таких как синтаксический и семантический анализ, создаются платформы, которые помогают компьютерам понимать разговорный человеческий язык. Например, переводчики языков, синтезаторы речи и программное обеспечение для распознавания речи основаны на компьютерной лингвистике.

Машинное обучение – это технология, которая обучает компьютеры на основе выборочных данных, чтобы повысить их эффективность [42]. Человеческий язык обладает особенностями, такими как сарказм, метафоры, вариации в структуре предложений и некоторые грамматические исключения, на изучение которых люди тратят много времени. С использованием методов машинного обучения разработчики обучают NLP-приложения распознавать и точно понимать эти особенности уже с самого начала.

Глубокое обучение – это специальная область машинного обучения, которая учит компьютеры учиться и мыслить подобно людям [43]. Оно включает нейронную сеть, состоящую из узлов обработки данных, которые напоминают операции, выполняемые человеческим мозгом. С помощью глубокого обучения компьютеры могут распознавать, классифицировать и находить сложные закономерности во входных данных.

Как правило, внедрение NLP начинается с сбором и подготовкой неструктурированных текстовых или речевых данных из различных источников, таких как облачные хранилища данных, опросы, электронные письма или внутренние приложения бизнес-процессов.

Первый этап - предварительная обработка [44]. Для подготовки данных для различных приложений в программном обеспечении NLP используются методы предварительной обработки, такие как токенизация, стемминг, лемматизация и удаление стоп-слов. Токенизация разбивает предложение на отдельные слова или фразы. Стемминг и лемматизация упрощают слова до их основной формы. Например, они преобразуют "начало" в "старт". Удаление

стоп-слов гарантирует, что слова, не несущие существенного значения для предложения, такие как "для" и "с", будут исключены.

Второй этап - обучение. После предварительной обработки исследователи используют эти данные для обучения моделей NLP с помощью методов машинного обучения [45,46,47, 48, 49].

Такие модели способны выполнять конкретные задачи на основе предоставленной текстовой информации. Для обучения алгоритмов NLP требуется большой объем данных, чтобы повысить их точность [50].

Третий этап - развертывание и вывод. После обучения модели специалисты по машинному обучению развертывают ее или интегрируют в существующую рабочую среду. Модель NLP принимает входные данные и предсказывает результат для конкретного сценария использования, для которого она предназначена. Приложение NLP может быть запущено на реальных данных, и ожидаемый результат будет получен.

Методы обработки естественного языка (NLP) разбивают человеческий текст или речь на более мелкие компоненты, которые компьютерные программы могут легко интерпретировать [51]. Ниже приведены общие возможности обработки и анализа текста в NLP.

Одна из задач NLP — это тегирование речи [52]. Это процесс, при котором программное обеспечение NLP отмечает отдельные слова в предложении в соответствии с их контекстуальными функциями, такими как существительные, глаголы, прилагательные или наречия [53]. Такая разметка помогает компьютеру понять, как слова взаимодействуют друг с другом и образуют значимые отношения.

Смысл слов может меняться в различных сценариях. Например, слово "замок" может иметь разные значения в разных предложениях. Для устранения неоднозначности программное обеспечение NLP определяет предполагаемое значение слова, используя языковую модель или словарные определения.

Распознавание речи преобразует голосовые данные в текст. Этот процесс включает разделение слов на более мелкие части и решение проблем,

связанных с акцентами, интонацией и неправильным использованием грамматики в разговорной речи. Важным применением распознавания речи является транскрипция, которую можно выполнить с помощью сервисов преобразования речи в текст.

Машинный перевод использует обработку естественного языка для преобразования текста или речи с одного языка на другой, сохраняя контекстуальную точность.

Распознавание именованных сущностей позволяет определить уникальные имена людей, мест, событий, компаний и прочего. Программное обеспечение NLP использует распознавание именованных сущностей для определения связей между различными сущностями в предложении.

Анализ эмоций [53], также известный как анализ тональности, является подходом искусственного интеллекта к интерпретации эмоций, выраженных в текстовых данных. Программа NLP анализирует текст с целью обнаружения слов или фраз, которые указывают на эмоции, такие как недовольство, счастье, сомнения, сожаление и другие скрытые эмоции.

Вот несколько общих подходов к обработке естественного языка:

1. Контролируемая обработка естественного языка (NLP): В этом подходе программное обеспечение обучается на основе маркированных или известных входных и выходных данных. С помощью больших объемов известных данных программа изучает, как правильно обрабатывать любой неизвестный ввод. Например, компании используют NLP-инструменты для категоризации документов по определенным этикеткам.
2. Неконтролируемая обработка естественного языка (NLP): В этом подходе используется статистическая языковая модель для прогнозирования закономерностей в немаркированных входных данных. Например, функция автозаполнения в текстовых сообщениях предлагает соответствующие слова, исходя из контекста и ответов пользователя.
3. Понимание естественных языков (NLU): это подмножество NLP, которое фокусируется на анализе значения предложений. NLU позволяет

программе находить сходные значения в разных предложениях или обрабатывать слова, имеющие разные значения [53].

4. Генерация естественного языка (NLG): Этот подход направлен на создание естественного текста, подобного тому, что создают люди, на основе ключевых слов или тем [53]. Например, интеллектуальный чат-бот с возможностями NLG может общаться с клиентами, как сотрудник службы поддержки, и генерировать соответствующие ответы.

Языковая модель, по сути, представляет собой простой алгоритм, который определяет вероятность правильности последовательности слов в предложении [51]. Когда мы заменяем небольшой объем данных на огромный массив данных из интернета, мы приближаемся к идее большой языковой модели.

Исследователи могут обучать нейронные сети на большом объеме текстовых данных, чтобы улучшить языковую модель. Благодаря большому объему текстов, которые модель учила, она может точно предсказывать следующее слово в последовательности. Модель становится настолько сложной, что способна выполнять различные задачи в области обработки естественного языка. Эти задачи включают обобщение текста, создание нового контента и даже имитацию человеческого разговора.

На этапе предварительного обучения модели языкового моделирования LLM ознакамливается с имеющимися текстовыми данными для изучения общей структуры и языковых правил.

За последние несколько лет LLM прошли предварительную подготовку на больших наборах данных, охватывающих значительную часть общедоступного Интернета. Например, модель языковой модели GPT-3 [63] обучалась на данных, включающих веб-страницы, веб-сообщения и оцифрованные книги, извлеченные из более чем 50 миллионов доменов.

Затем эти обширные наборы данных подаются на вход модели, известной как "трансформер". Трансформеры представляют собой тип

глубоких нейронных сетей, которые особенно хорошо работают с последовательными данными.

Трансформеры используют архитектуру кодировщик-декодировщик для обработки входных и выходных данных [64]. Фактически, трансформер содержит две нейронные сети: кодировщик и декодировщик. Кодировщик извлекает смысл из входного текста и сохраняет его в виде вектора. Затем декодировщик получает этот вектор и интерпретирует текст.

Однако ключевой концепцией, которая обеспечивает отличную работу архитектуры трансформера [65], является использование механизма самообратной связи. Идея внутреннего внимания позволяет модели обращать внимание на наиболее значимые слова в предложении. Этот механизм учитывает даже взаимосвязи между словами, находящимися в дальних позициях друг от друга.

Еще одним преимуществом самообратной связи является возможность параллельной обработки. Вместо последовательной обработки данных, модели трансформера могут обрабатывать все входные данные одновременно. Это позволяет моделям трансформера обучаться на огромных объемах данных относительно быстро по сравнению с другими методами.

После завершения предварительного этапа обучения вы можете вводить новый текст для базовой модели LLM с целью дальнейшего улучшения ее результатов. Этот процесс называется "тонкая настройка" [54] и часто применяется для достижения лучших результатов LLM в конкретной задаче.

Например, можно использовать LLM для создания контента для различных целей [55].

Существует несколько видов тонкой настройки. Быстрая настройка предполагает предоставление модели небольшого количества примеров с надеждой, что языковая модель сама выяснит, как получить аналогичные результаты. Одноразовая настройка подразумевает аналогичный процесс, но с предоставлением только одного примера.

LLM, включая GPT-3, обладают способностью выполнять множество задач, даже без сложной настройки. Однако, у этих моделей все еще есть свои ограничения.

Отсутствие полного понимания контекста.

На первый взгляд LLM могут показаться интеллектуальными. Однако, их функционирование отличается от работы человеческого мозга. LLM полагаются только на статистические расчеты для достижения результатов и не способны самостоятельно рассуждать о идеях и концепциях.

Из-за этого LLM могут давать бессмысленные ответы просто потому, что слова кажутся "правильными" или "статистически вероятными", даже если они расположены в неправильном порядке.

Галлюцинации.

Такие модели, как GPT-3, также страдают от неточности ответов. LLM могут порой создавать впечатление, будто они знают ответ, когда на самом деле они генерируют неверную информацию без осознания отсутствия реальных оснований для этого ответа [53].

Например, пользователь может попросить модель объяснить мысли Стива Джобса о последнем iPhone. Модель может процитировать что-то выдуманное на основе имеющихся у нее обучающих данных.

Предубеждения и ограниченные знания.

Как и многие другие алгоритмы, большие языковые модели могут усваивать предвзятость, присутствующую в обучающих данных. Поскольку мы все больше полагаемся на LLM для получения информации, разработчикам этих моделей необходимо найти способы снизить потенциальные негативные последствия предвзятых ответов.

Также стоит отметить, что слабые места в данных для обучения модели также могут ограничить ее знания. В настоящее время требуется продолжительное время для обучения больших языковых моделей, и эти модели полагаются на ограниченные объемы данных. Вот почему ChatGPT имеет ограниченные знания о событиях, произошедших после 2021 года [53].

1.8 Результаты и выводы

В этом разделе был рассмотрен библиографический анализ, аргументирована актуальность, новизна и уникальность помощи в принятии решений на различных этапах жизненного цикла ит-инновации с использованием ChatGPT, а также изложены общие понятия больших языковых моделей, описаны их функции и разновидности. Описаны преимущества использования искусственного интеллекта при принятии решений, а также приведены примеры успешного применения данной технологии.

2 МЕТОДЫ И АЛГОРИТМЫ ВЗАИМОДЕЙСТВИЯ С ЯЗЫКОВЫМИ МОДЕЛЯМИ

2.1 Методы взаимодействия с языковыми моделями

Подготовка подсказок (Prompt engineering) — это концепция искусственного интеллекта, особенно обработки естественного языка. В оперативной инженерии описание задачи, которую должен выполнить ИИ, встроено во входные данные, например, как вопрос, вместо того, чтобы быть явно заданным. Быстрая разработка обычно работает путем преобразования одной или нескольких задач в набор данных на основе подсказок и обучения языковой модели с помощью так называемого «обучения на основе подсказок» или просто «обучения подсказок» [56].

Простыми запросами можно достичь значительных результатов, однако качество этих результатов зависит от объема предоставляемой информации и ее качества. Запрос может включать инструкции, вопросы или другие данные, такие как контекст, входные данные или примеры, передаваемые модели [57]. Использование этих элементов позволяет более точно указать модели, что именно вы ищете, и, в результате, получить более высококачественные результаты.

Начнем с простого примера. Запрос: «температура воды», полученный ответ: «Температура воды — это мера теплоты или холода, которая определяет, насколько горячей или холодной является вода. Температура воды может быть измерена в градусах Цельсия (°C) или Фаренгейта (°F). Обычно вода имеет температуру около комнатной (приблизительно 20-25 °C) или тела (приблизительно 37 °C) для человека, но она может меняться в зависимости от окружающей среды и других факторов, таких как источник тепла или холода». Как можно заметить, когда языковая модель генерирует следующие фразы, они имеют смысл, связанный с темой "температура воды". Однако результаты могут быть неожиданными или далекими от того, что вы намерены достичь с помощью данной задачи.

Немного видоизменим задачу для ChatGPT. Запрос: «Закончи предложение: температура воды», полученный ответ: «Температура воды может иметь значительное влияние на различные процессы и явления. Она определяет комфортность окружающей среды для людей и животных, а также может влиять на физические и химические свойства воды. Кроме того, температура воды является важным фактором при определении возможности развития и выживания различных видов растений и животных в водных экосистемах.» Такой подход, при котором создаются оптимальные указания для модели, чтобы она могла выполнить задачу, называется подсказкой.

Подсказка форматированием

Языковой модели можно задавать задачу в виде вопроса, и в таком случае безусловно он выдаст вам некий результат, который является ответом. Но можно персонализировать ожидаемые ответы под себя задавая контекст.

Пример: Вопрос: есть ли хорошие фильмы, снятые до 1930 г? Ответ: Нет

Вопрос: существуют ли хорошие фильмы, снятые в период с 1930 по 1960 г?

Ответ: Нет Вопрос: существуют ли хорошие фильмы, снятые в период с 1960

по 2000? Ответ: да Вопрос: Фильмы какого периода стоит посмотреть?

Ответ, полученный от языковой модели: «Рекомендуется посмотреть фильмы, снятые в период с 1960 по 2000 год.»

Данный подход называется подсказкой формат типа: <Question>? <Answer>
<Question>? <Answer> <Question>? <Answer> <Question>?

Элементы подсказки.

Подсказка включает в себя разные вещи: инструкцию, контекст, входные данные и индикатор вывода. Инструкция — это задача или указание, которое модель должна выполнить. Контекст — это дополнительная информация, которая поможет модели дать лучший ответ. Входные данные — это вопрос или данные, на которые будет происходить поиск ответа. Индикатор вывода — это формат или тип ответа. При этом не обязательно использовать все четыре элемента подсказки, и выбор зависит от задачи.

2.2 Методики создания подсказок

2.2.1 Общие рекомендации по созданию подсказок

Некоторые советы, которые стоит принять во внимание, при создании подсказок [57]:

1. Нужно декомпозировать задачу при возможности: в процессе придумывания подсказки, важно помнить, что это процесс, который требует много экспериментов, чтобы достичь оптимальных результатов. Следует начать с простых подсказок и постепенно добавлять новые элементы и контекст, стремясь к лучшим результатам. Поэтому очень важно повторять итерации подсказки по мере продвижения. При наличии большой задачи, включающей в себя множество подзадач, можно попробовать декомпозировать её на более мелкие задачи и постепенно улучшать результаты каждой. Это поможет избежать излишней сложности в начале процесса разработки подсказок;
2. необходимо писать инструкцию: можно создавать эффективные подсказки для простых задач, используя команды, чтобы указать модели, что именно ей следует сделать, например, "Написать", "Отфильтровать", "Нарисовать", "Перевести", "Упорядочить" и т. д. Важно экспериментировать и определить, какие команды работают лучше всего. Нужно попробовать использовать разные инструкции с разными ключевыми словами, контекстами и данными, чтобы найти наиболее подходящий вариант для решаемой задачи. Чем более конкретный и релевантный контекст у задачи, тем лучше результаты. Также рекомендуется размещать инструкции в начале подсказки и использовать разделитель, например "---", чтобы четко отделить инструкцию от контекста;
3. указание особенностей запроса: нужно быть очень конкретным в инструкциях и задачах, которые чтобы модель должна выполнить. Чем подробнее будет написано объяснение задачи, тем лучше будут

результаты. Это особенно важно, если происходит поиск определенного результата или стиля. Не существует слов или ключевых фраз, которые всегда дадут лучшие результаты. Важно иметь хороший формат и понятную инструкцию. На самом деле, использование примеров в инструкции очень полезно для получения желаемого результата в определенных форматах. При разработке инструкции, не стоит забывать о длине, так как для неё есть ограничения. Необходимо подумать, насколько подробной и точной она должна быть. Не стоит включать слишком много лишних деталей - они не всегда полезны. Детали должны быть актуальными и помогать выполнить задачу. Это то, над чем придется экспериментировать. Рекомендуется проводить много экспериментов и тестов, чтобы оптимизировать инструкции под определенные задачи;

4. нужно избегать неточностей: данный пункт в целом схож с предыдущим, в попытках задать некий контекст и особенности запроса нужно максимально корректно их задавать, так как одна неточность может увести результат в другое направление;
5. не указывать об ограничениях в прямом виде, а стараться сформулировать задачу таким образом, чтобы не было необходимости указывать о них. Таким образом будет получена большая конкретность и фиксация на деталях, которые приводят к более релевантным результатам.

2.2.2 Методы формулирования подсказок

Для решения более сложных задач были разработаны различные техники создания подсказок:

1. zero-Shot Prompting: современные языковые модели обучены на больших объемах данных и поэтому они могут решать некоторые задачи с нуля. На данной методике не стоит задерживаться надолго, так как она подразумевает отправку одного запроса, без указаний инструкций, контекстов [58];

2. **few-Shot Prompting**: подсказка с несколькими выстрелами может быть использована для обучения модели, предоставляя демонстрации в подсказке, чтобы направить ее на более высокую производительность. Демонстрации служат условием для последующих примеров, где мы ожидаем, что модель будет генерировать ответы. При масштабировании моделей до достаточного размера, появляются новые свойства в использовании нескольких выстрелов [59]. Пример из работы “Language Models are Few-Shot Learners” используется для демонстрации подсказки с несколькими выстрелами, где задача состоит в правильном использовании нового слова в предложении;
3. подсказка по цепочке мыслей: по сути своей смысл данной методики заключается в том, чтобы задавать конкретные алгоритмы решения задачи и примеры решения аналогичных задач. Можно использовать несколько подсказок в сочетании, чтобы достичь лучших результатов в более сложных задачах, где требуется рассуждение перед предоставлением ответа [60].
4. **self-consistency**: вероятно является одним из продвинутых методов оперативной разработки. Идея метода заключается в выборе нескольких разнообразных путей рассуждений через CoT с несколькими попытками и использовании поколений для выбора наиболее последовательного ответа. Этот метод повышает производительность подсказок по цепочке мыслей в задачах, связанных с арифметикой и логическими рассуждениями [61].
5. подсказка сгенерированных знаний (**generated knowledge prompting**): в статье [62] была предложена идея использования модели для генерации знаний, которые затем могут быть использованы в качестве части подсказки. Суть данного метода заключается в том, что в модель следует передать некоторые формулировки, а затем закрепить их фактами или определениями. По предложению авторов статьи, если в модель передать достаточное количество знаний, то при отправке запроса в виде формулировки, модель понимая каким образом стоит вести размышление

сделает предположение на основе своих знаний из базы данных и тех, что были получены при взаимодействии с человеком.

2.3 Алгоритм взаимодействия с ChatGPT

При проведении исследования были обнаружены также новые не подтвержденные методы работы с языковыми моделями, его рассмотрим далее:

- определите цели: перед использованием ChatGPT в научной работе необходимо ясно определить цели исследования и вопросы, на которые вы хотите получить ответы. Это поможет узконаправленному взаимодействию с моделью и сфокусироваться на конкретных задачах. Например, можно ответить на вопрос: Какой результат является ожидаемым при использовании ChatGPT;
- определите входные и выходные данные: следует разобраться в том, какие данные будут предложены нейросети в качестве контекста и данных, и в каком виде нужно получить итоговый результат;
- декомпозируйте задачу на мелкие;
- при возможности указывайте подсказки в прямом виде. Пример можно посмотреть на рисунке №6;
- создайте несколько различных запросов для решения одной и той же задачи;
- проанализируйте полученные результаты соотнеся их с изначальными целями.

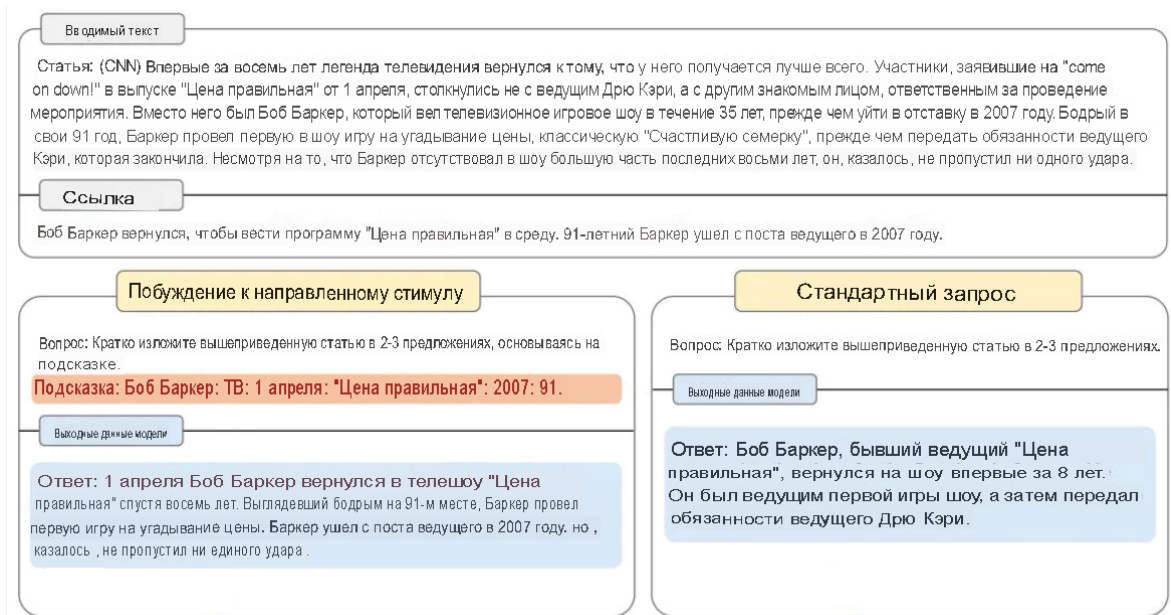


Рисунок 7 – Сравнение различных типов подсказок [57]

3 ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ БОЛЬШОЙ ЯЗЫКОВОЙ МОДЕЛИ CHATGPT НА РАЗЛИЧНЫХ ЭТАПАХ ЖИЗНЕННОГО ЦИКЛА

3.1 Применение ChatGPT на этапе научного исследования

3.1.1 Подготовка литературного обзора для научного исследования

В последних версиях ChatGPT разработчики неплохо потрудились над исправлением ошибок при работе над составлением литературного обзора, но на данный момент всё ещё данный процесс нельзя назвать эффективным. При проведении исследования на подготовку литературного обзора для НИР были опробованы несколько подходов задания подсказки в запросе к нейросети. Рассмотрим самые интересные из них.

Вначале был опробован алгоритм, которым можно получить список литературы на SciVal.

Запрос: «Какую научную литературу ты можешь предложить для научной работы с ключевыми словами: Искусственный интеллект, принятие решений при указании источника выдавай ссылку на работу».

Полученный результат: «Я могу предложить вам несколько научных работ, связанных с искусственным интеллектом и принятием решений:

1. Autor, D. (2018). Deep learning for autonomous driving: A survey. IEEE Transactions on Intelligent Transportation Systems, 1-20. doi: 10.1109/TITS.2018.2797856.
2. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mané, D. (2016). Concrete Problems in AI Safety. arXiv preprint arXiv:1606.06565.
3. Russell, S. J., & Norvig, P. (2016). Artificial Intelligence: A Modern Approach (3rd ed.). Prentice Hall.

4. Pearl, J. (1984). Heuristics: Intelligent Search Strategies for Computer Problem Solving. Addison-Wesley.
5. Tversky, A., & Kahneman, D. (1974). Judgment under Uncertainty: Heuristics and Biases. Science, 185(4157), 1124-1131. doi: 10.1126/science.185.4157.1124.
6. Zhang, S., Yao, L., Sun, A., & Tay, Y. (2020). Deep learning based recommender system: A survey and new perspectives. ACM Computing Surveys (CSUR), 53(2), 1-36. doi: 10.1145/3318267.

Пожалуйста, обратите внимание, что некоторые из этих работ могут быть платными. Чтобы получить доступ к полному тексту статьи, вам может потребоваться подписка на журнал или определенные привилегии доступа».

Стоит отметить, что при поиске литературы предложенной нейросетью в сервисе «arxiv.org» не были найдены ошибки.

Мною нейросети была предложена тема и структура работы, затем был запрошен список источников, изучив которую, можно написать работу.

Изначальный запрос: Срок не важен. Язык - русский. Тема работы: "Исследование на тему внедрения ChatGPT в процессы It компании и научные работы, структура:

- введение;
- обоснование актуальности выбранной темы;
- цель работы Задачи исследования;
- обзор литературы и источников.

Описание ChatGPT:

- а) общая информация о ChatGPT;
- б) архитектура и основные принципы работы;
- в) возможности применения в бизнесе и науке;
- г) внедрение ChatGPT в процессы IT компании:
 - 1) анализ существующих проблем в процессах компании;

- 2) оценка возможности применения ChatGPT для решения проблем;
- 3) примеры использования ChatGPT в IT компаниях;
- 4) планирование и реализация внедрения ChatGPT в компанию.

Использование ChatGPT в научных работах: Обзор научных работ, использующих ChatGPT Анализ применения ChatGPT в научных исследованиях Возможности дальнейшего использования ChatGPT в науке Экономический анализ внедрения ChatGPT в процессы IT компании: Оценка экономической эффективности внедрения ChatGPT Сравнение затрат на внедрение ChatGPT и потенциальной выгоды для компании Рекомендации по использованию ChatGPT в бизнес-процессах Заключение: Результаты исследования Выводы Рекомендации по дальнейшему применению ChatGPT в бизнесе и науке Список литературы: Перечень использованных источников " И просьба написать список источников: «Напиши список источников литературы для этой работы.»

Полученный результат: «Конечно, вот список основных источников литературы, которые использовались для написания данной работы:

1. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. arXiv preprint arXiv:2005.14165.
2. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 4171-4186.
3. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI Blog, 1(8), 9.
4. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 5998-6008.

6. Gehring, J., Auli, M., Grangier, D., Yarats, D., & Dauphin, Y. N. (2017). Convolutional sequence to sequence learning. arXiv preprint arXiv:1705.03122.
7. Holtzman, A., Buys, J., Forbes, M., Bosselut, A., Golub, D., & Choi, Y. (2020). The curious case of neural text degeneration. arXiv preprint arXiv:1904.09751.
8. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692.
9. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., & Bowman, S. R. (2019). GLUE: A multi-task benchmark and analysis platform for natural language understanding. arXiv preprint arXiv:1804.07461.
10. Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving language understanding by generative pre-training. URL <https://openai.com/blog/language-unsupervised/>.
11. Vaswani, A., Fedus, W., Narang, N., Zhao, M., Parmar, N., Uszkoreit, J., & Bengio, S. (2021). Scaling transformer models with transparent model parallelism. arXiv preprint arXiv:2104.05610.

Эти источники представляют собой научные статьи, которые были опубликованы в пиар-ревью журналах и конференциях. Они являются авторитетными источниками в области иску...», для продолжения предложения можно попросить нейросеть завершить действие, но для нашей задачи в этом нет смысла.

Проверив результат можно заметить, что предложенная литература действительно существует, и в целом подходит под исследование.

Однозначно неудачным решением было предложено запросить научную литературу на основе темы, при этом указывая некоторые ограничения, в пункте я укажу лишь конечный запрос и полученный результат. Запрос: «Напиши список из 5 научных работ на тему тестирования ПО, где авторами являются граждане России, а также приложи ссылку на работы».

Полученный результат:

1. "Автоматизация тестирования мобильных приложений с использованием Appium" - И. А. Абдуллаева, Н. А. Мухамадеева, Д. А. Бикмухаметова. <https://cyberleninka.ru/article/n/avtomatizatsiya-testirovaniya-mobilnyh-prilozheniy-s-ispolzovaniem-appium/viewer>
2. "Интерактивная система тестирования программного обеспечения" - Д. Ю. Горбунов, В. Ю. Ломакин. <https://www.elibrary.ru/item.asp?id=25722357>
3. "Создание системы автоматизированного тестирования на основе Selenium WebDriver" - С. А. Поляков, Д. В. Шатунов. <https://cyberleninka.ru/article/n/sozдание-sistemy-avtomatizirovannogo-testirovaniya-na-osnove-selenium-webdriver/viewer>
4. "Анализ возможностей использования машинного обучения для улучшения качества тестирования программного обеспечения" - М. В. Кузнецов, Д. С. Михайлов. <https://www.elibrary.ru/item.asp?id=45636487>
5. "Опыт применения методологии контроля качества программного обеспечения на примере разработки интеллектуальных систем" - А. А. Сычев, Д. А. Афонин, А. В. Бондаренко. <https://cyberleninka.ru/article/n/opyt-primeneniya-metodologii-kontrolya-kachestva-programmnogo-obespecheniya-na-primere-razrabotki-intellektualnyh-sistem/viewer>».

В результате запроса ChatGPT придумал 4 источника, при этом указав ссылки, которые не ведут никуда.

Также стоит отметить, что даже в кейсах, когда у нейросети вышло выдать качественный результат можно найти проблемы. В попытке расширить список литературы, при уже полученном, качественном результате, я попросил ChatGPT дополнить список ещё 10 работами, и так несколько раз результат не всегда был положительным, в 2 из 5 попыток нейросеть начала придумывать список литературы. Учитывая всё это можно сделать главный

вывод: запрашивая любой результат у нейросети стоит проводить проверку полученных результатов.

3.1.2 Анализ литературы

Обработка и анализ больших объемов текстовых данных являются важными задачами в научных исследованиях, особенно в области обработки естественного языка (Natural Language Processing, NLP) и информационного поиска. Применение ChatGPT в этом контексте может предоставить исследователям мощный инструмент для работы с текстовыми данными больших объемов.

Обработка текстовых данных, извлечение информации и анализ текстов: ChatGPT может использоваться для предварительной обработки текстовых данных, таких как очистка от шума, токенизация, лемматизация и извлечение ключевых слов.

Запрос: «Проведи токенизацию сообщения. Сообщение: Параллельные вычисления: В отличие от RNN, Transformer позволяет параллельно обрабатывать входные последовательности, что существенно ускоряет процесс обучения и вывода.»

Полученный результат: «Токенизация сообщения: ["Параллельные", "вычисления", ":", "В", "отличие", "от", "RNN", ",", "Transformer", "позволяет", "параллельно", "обрабатывать", "входные", "последовательности", ",", "что", "существенно", "ускоряет", "процесс", "обучения", "и", "вывода", "."]»

Это помогает исследователям подготовить данные для дальнейшего анализа и обработки. Затем ChatGPT может применяться для извлечения информации из текстовых данных, включая именованные сущности, ключевые фразы, связи и тематическое моделирование. Это позволяет исследователям получить важные факты и структурировать текстовые данные для дальнейшего анализа.

Запрос: «Выпиши ключевые слова из научной работы Статья: Attention Is All You Need»

Полученный результат: «Ключевые слова из научной работы "Attention Is All You Need" (Внимание — это все, что вам нужно) включают:

- 1) Внимание (Attention);
- 2) Модель трансформера (Transformer model);
- 3) Нейронная сеть (Neural network);
- 4) Машинное обучение (Machine learning);
- 5) Обработка естественного языка (Natural language processing);
- 6) Последовательность (Sequence);
- 7) Механизм внимания (Attention mechanism);
- 8) Энкодер-декодер (Encoder-decoder);
- 9) Входные данные (Input data);
- 10) Внимательность (Attentional);
- 11) Многоязычность (Multilingualism);
- 12) Применение (Application);
- 13) Генерация текста (Text generation);
- 14) Перевод (Translation);
- 15) Классификация (Classification).

Эти ключевые слова представляют основные темы и концепции, которые обсуждаются в работе "Attention Is All You Need".»

Таким образом, если, начиная НИР студент обладает информацией, о каких-либо работах, то он может использовать данную методику, для получения ключевых слов или фраз по ним, чтобы затем использовать их для получения более подробного списка литературы, например на сервисе SciVal.

Анализ сентимента и эмоций: ChatGPT может использоваться для анализа сентимента и эмоций в текстовых данных. Это позволяет исследователям оценивать отношение людей к определенным темам и выявлять эмоциональные состояния, выраженные в тексте.

Классификация и кластеризация текста: ChatGPT может использоваться для классификации и кластеризации текстовых данных. Это позволяет исследователям автоматически группировать тексты по схожести и определять их принадлежность к определенным категориям или классам.

Запрос: «Сгруппируй список источников по типу. Источники: Attention Is All You Need, Wikipedia, Evolution Strategies as an Alternative to Reinforcement Learning, Deep double descent, John Carmack | Lex Fridman — Doom, Quake, VR, AGI, Programming, Video Games, and Rockets».

Полученный результат: «Группировка списка источников по типу:

а) научные работы:

- 1) Attention Is All You Need;
- 2) Evolution Strategies as an Alternative to Reinforcement Learning - arxiv.org;
- 3) Deep double descent - arxiv.org;

б) веб-ресурсы:

- 1) Wikipedia - wikipedia.org;
- 2) John Carmack | Lex Fridman — Doom, Quake, VR, AGI, Programming, Video Games, and Rockets - (ссылка отсутствует, возможно, это интервью или видео, которое не имеет прямой связи с научными работами).

Пожалуйста, обратите внимание, что источник "John Carmack | Lex Fridman — Doom, Quake, VR, AGI, Programming, Video Games, and Rockets" не

является научной работой и не имеет прямого отношения к исследованиям или научным публикациям.»

Анализ связей и сетей: ChatGPT может помочь исследователям анализировать связи и взаимодействия между текстовыми элементами, такими как персоны, организации, темы или концепции. Это открывает возможности для исследования социальных сетей, анализа взаимодействия авторов и раскрытия структурных характеристик текстовых данных. Данный пункт проще исследовать на практических данных, что и было сделано в исследовании о применении ChatGPT в сфере тестирования.

Суммаризация текста: ChatGPT может быть использован для автоматической генерации краткой суммы текстового содержания. Это позволяет исследователям быстро извлечь ключевую информацию и основные идеи из больших объемов текста.

Запрос: «Напиши краткое описание научной работы Статья: Attention Is All You Need»

Полученный результат: «Статья "Attention Is All You Need" является одной из ключевых работ в области обработки естественного языка и машинного обучения. Авторы представляют новый подход к моделированию последовательностей, основанный на механизме внимания [57], без использования традиционных рекуррентных нейронных сетей.

В работе представлено новое архитектурное решение под названием "Transformer", которое полностью опирается на механизм внимания для обработки и генерации текста. Оно состоит из энкодера и декодера, каждый из которых состоит из нескольких слоев. Основная идея заключается в том, что вся информация о входных данных передается через слои энкодера с помощью операций внимания, а затем декодер генерирует соответствующие выходные последовательности.

Этот подход позволяет достичь высокой гибкости и эффективности в моделировании различных задач, таких как машинный перевод, генерация текста и классификация. В работе также обсуждаются преимущества модели

Transformer по сравнению с традиционными рекуррентными моделями, такими как LSTM и GRU.

"Attention Is All You Need" стал одной из наиболее цитируемых и влиятельных статей в области обработки естественного языка и с тех пор послужил основой для множества последующих исследований и разработок в этой области.»

Применение ChatGPT для обработки и анализа больших объемов текстовых данных облегчает исследователям работу с текстовыми корпусами, извлечение информации и анализ текстового контента.

Можно указывать дополнительную подсказку, например попросить нейросеть раскрыть в чем конкретно являются преимущества модели Transformer по сравнению с традиционными рекуррентными моделями.

Запрос: «Напиши краткое описание научной работы Статья: Attention Is All You Need Подсказка: расскажи подробно преимущества Transformer по сравнению с традиционными рекуррентными моделями»

Полученный результат: «Статья "Attention Is All You Need" представляет собой революционный подход к обработке последовательностей в нейронных сетях, представленный в виде модели под названием "Transformer". Этот подход, разработанный исследователями Google Brain, предлагает полностью отказаться от традиционных рекуррентных моделей, таких как рекуррентные нейронные сети (RNN), и полностью основаться на механизме внимания.

В статье авторы демонстрируют, что Transformer-модель способна достичь лучших результатов в ряде задач обработки естественного языка, таких как машинный перевод и генерация текста, по сравнению с традиционными рекуррентными моделями. Они предлагают новый подход к моделированию внимания, который позволяет модели сосредоточиться на важных частях входной последовательности и учесть контекст при генерации выходной последовательности.

Преимущества Transformer перед традиционными рекуррентными моделями включают:

1. **Параллельные вычисления:** В отличие от RNN, Transformer позволяет параллельно обрабатывать входные последовательности, что существенно ускоряет процесс обучения и вывода.
2. **Дальние зависимости:** благодаря механизму внимания, Transformer легко улавливает зависимости между удаленными частями последовательностей, что является сложной задачей для RNN.
3. **Долгосрочная память:** Transformer сохраняет информацию о предыдущих состояниях во внимательности, что помогает ему учитывать контекст на длинных временных интервалах.
4. **Гибкость:** Transformer легко масштабируется для обработки как коротких, так и длинных последовательностей и может быть применен в различных задачах обработки естественного языка.

Статья "Attention Is All You Need" положила основу для использования Transformer-моделей во многих современных задачах обработки естественного языка и считается одной из самых важных работ в этой области».

Таким образом можно заметить то, что при указании дополнительных задач нейросеть выдает абсолютно разные значения, пытаясь решить проблему, указанную в запросе.

3.1.3 Обоснование научной новизны научного исследования

Отсутствие предыдущих исследований: В случае, если в литературе нет существующих работ, решающих ту же проблему или исследующих ту же тему, можно обосновать научную новизну своей работы. Запрос в ChatGPT может быть следующим: "Как обосновать научную новизну исследования, если в литературе отсутствуют предыдущие работы по данной проблеме?".

Пример полученного ответа от ChatGPT:

"Одним из способов обоснования научной новизны в случае отсутствия предыдущих исследований может быть акцентирование важности и актуальности решения данной проблемы для области знаний. Также можно указать на то, что отсутствие предыдущих работ по данной теме свидетельствует о необходимости проведения исследования и заполнения этой научной ниши."

Расширение предыдущих исследований: если в литературе имеются некоторые работы, но ваше исследование вносит новые аспекты или расширяет предыдущие исследования, можно обосновать новизну своей работы. Запрос в ChatGPT может быть следующим: "Каким образом можно обосновать научную новизну исследования, расширяющего предыдущие исследования в области X?".

Пример полученного ответа от ChatGPT:

"Для обоснования научной новизны при расширении предыдущих исследований можно ссылаться на недостатки и ограничения существующих работ и предлагать новые подходы или методы, которые решают эти проблемы. Также можно обратить внимание на новые результаты, полученные в рамках исследования, и объяснить, как они дополняют или улучшают предыдущие работы."

Применение новых методов или подходов: если ваше исследование использует новые методы или подходы, которые ранее не применялись, можно обосновать научную новизну исследования. Запрос в ChatGPT может быть следующим: "Как обосновать научную новизну исследования, применяющего новые методы или подходы в области Y?".

Пример полученного ответа от ChatGPT:

"Для обоснования научной новизны при применении новых методов или подходов можно описать их преимущества и различия от существующих методов. Также можно провести сравнительный анализ результатов, полученных при использовании новых методов, с результатами предыдущих"

исследований и показать, что новые методы дают более точные или эффективные результаты."

Пример запроса в ChatGPT: "Какие варианты обоснования научной новизны при исследовании алгоритмов машинного обучения для решения проблемы X?"

Пример ответа от ChatGPT: "При исследовании алгоритмов машинного обучения для решения проблемы X можно обосновать научную новизну, указав на уникальные характеристики или модификации предлагаемых алгоритмов. Также можно сравнить их производительность с существующими алгоритмами и показать преимущества новых подходов в решении данной проблемы."

3.1.4 Генерация идей

Генерация идей и концепций является важным этапом научных исследований, который требует креативного мышления и поиска новых подходов к решению проблем. Применение ChatGPT в этом контексте может быть полезным инструментом для генерации новых идей, формулировки концепций и предложений.

1. Креативная генерация текста: ChatGPT обладает способностью генерировать тексты на основе заданных входных данных. Это позволяет исследователям получать новые идеи и концепции, основываясь на существующей информации или примерах.
2. Поиск альтернативных решений: ChatGPT может помочь исследователям генерировать альтернативные варианты решений и подходов к научным проблемам. Это может способствовать разнообразию и инновационности исследований.
3. Поддержка и расширение исследовательского процесса: ChatGPT может дополнять исследовательский процесс, предоставляя идеи, концепции и

перспективы, которые исследователь мог не рассмотреть. Это может стимулировать творческое мышление и обогатить исследовательскую работу.

Применение ChatGPT для генерации идей и концепций в научных работах может способствовать инновационности и оригинальности исследований, а также помочь исследователям в поиске новых подходов к решению проблем.

Пример того каким образом можно сгенерировать идеи для научной работы с помощью ChatGPT. Необходимо обработать список источников, которые были найдены в процессе приготовления к работе и проанализировать их на недостатки.

Пример. Запрос в ChatGPT: « Проанализируй научную статью и опиши какие недостатки в ней есть, как её можно улучшить. Статья: <https://arxiv.org/pdf/1706.03762.pdf> - Attention Is All You Need»

Результат: «Анализ научной статьи "Attention Is All You Need":

Статья "Attention Is All You Need" представляет новый подход к машинному переводу, который получил значительное влияние в области глубокого обучения. Несмотря на значимость этой работы, статья также имеет несколько недостатков, а также потенциальные области для улучшения.

1. Отсутствие примеров и иллюстраций: Статья содержит много абстрактных концепций, которые могут быть трудными для восприятия без визуальной поддержки. Включение примеров и иллюстраций может значительно улучшить понимание материала, особенно для школьников.
2. Ограничение на одну задачу: Статья сосредоточена на задаче машинного перевода и подходе Transformer. Она не охватывает другие аспекты машинного обучения и применение модели в других областях. Дополнительное исследование и примеры, показывающие применение модели в различных задачах, могли бы сделать статью более всесторонней и полезной.

3. Отсутствие сравнения с другими моделями: В статье не приведено сравнение модели Transformer с другими существующими подходами к машинному переводу. Включение сравнительного анализа с другими моделями позволило бы лучше оценить преимущества и ограничения подхода, предложенного в статье.

Улучшения, которые можно внести в статью:

1. Упрощение и объяснение математических моделей и обозначений с использованием понятных примеров и иллюстраций.
2. Расширение области применения модели, включая другие задачи машинного обучения, чтобы обеспечить более полное представление возможностей модели Transformer.
3. Включение сравнительного анализа с другими существующими моделями машинного перевода для оценки относительных преимуществ и недостатков подхода, предложенного в статье.
4. Дополнительные ресурсы, такие как презентации или видеоматериалы, которые могут помочь визуализировать и упростить представленные концепции для более широкой аудитории, включая школьников.

В целом, статья "Attention Is All You Need" представляет важный вклад в область машинного перевода, но может быть улучшена с помощью более простого и доступного изложения материала, а также дополнительного исследования и примеров для демонстрации применения модели в различных областях и сравнения с другими подходами.»

Получив похожие результаты по другим работам у автора НИР сложится некий список недостатков текущих работ, которые можно будет попробовать решить.

3.1.5 Исследование алгоритмов постановки задач управления

В данном разделе будет рассмотрено применение ChatGPT в научных работах для исследования алгоритмов постановки задач управления. Алгоритмы постановки задач управления широко применяются в области автоматизации и управления системами различного рода, включая промышленные процессы, транспортные системы, робототехнику и многие другие. Исследование и улучшение таких алгоритмов имеет важное значение для оптимизации и эффективного управления системами.

Определение задачи управления:

Перед тем, как приступить к исследованию алгоритмов постановки задач управления, необходимо определить саму задачу управления. Задача управления обычно формулируется как оптимизационная проблема, где требуется найти оптимальное управляющее воздействие на систему с учетом определенных критериев и ограничений. Подходы к формулированию задач управления могут варьироваться в зависимости от специфики системы и требуемых результатов.

Пример запроса в ChatGPT: "Какие существуют методы формулирования задач управления в контексте промышленных процессов?"

Ответ ChatGPT: "В контексте промышленных процессов используются различные методы формулирования задач управления, включая:

1. Метод оптимального управления: основан на математическом описании системы и определении оптимального управляющего воздействия, минимизирующего выбранный критерий.
2. Метод адаптивного управления: позволяет системе самостоятельно адаптироваться к изменениям во внешней среде или внутренних условиях.
3. Метод обратной связи: основан на измерении выходов системы и корректировке управления для достижения желаемых результатов.

4. Метод стохастического управления: учитывает случайные воздействия и шумы в системе, применяя вероятностные методы для оптимального управления.

Источник: [Ссылка на источник]"

В данном примере ChatGPT предоставляет общую информацию о некоторых методах формулирования задач управления в контексте промышленных процессов.

Пример запроса в ChatGPT: "Каким образом можно сформулировать задачу управления для оптимизации процесса разработки программного обеспечения в IT-компаниии?"

Ответ ChatGPT: "Для оптимизации процесса разработки программного обеспечения в IT-компаниии можно сформулировать следующую задачу управления:

Цель: улучшить производительность и качество разработки программного обеспечения в IT-компаниии.

Задача управления: разработать и внедрить систему автоматизации процесса разработки, включающую следующие этапы:

1. Управление требованиями: Разработка и поддержка процесса управления требованиями, включая сбор, анализ, документирование и отслеживание изменений требований.
2. Управление проектами: Планирование, контроль и отслеживание выполнения проектов разработки, включая управление ресурсами, расписанием и бюджетом.
3. Управление версиями и конфигурацией: Установка и поддержка системы контроля версий, а также управление конфигурацией программного обеспечения.

4. Управление тестированием: Разработка и поддержка процесса тестирования программного обеспечения, включая планирование тестов, создание тестовых случаев и выполнение тестирования.
5. Управление качеством: Внедрение системы управления качеством, включая процессы проверки качества кода, статический анализ, тестирование на соответствие требованиям и обратную связь с пользователями.
6. Управление коммуникацией: Обеспечение эффективной коммуникации и взаимодействия между разработчиками, тестировщиками, менеджерами и заказчиками.
7. Мониторинг и анализ: Сбор и анализ данных о процессе разработки для выявления узких мест, определения проблем и принятия управленческих решений.

Результаты задачи управления: Улучшение производительности, снижение рисков и повышение качества процесса разработки программного обеспечения в IT-компаниях.

Для получения такого результата от ChatGPT, рекомендуется сформулировать запрос более конкретно и указать требуемый результат. Например, можно использовать вопросы вроде "Каким образом сформулировать задачу управления для оптимизации процесса X в Y-компаниях?" или "Какие основные шаги следует включить в задачу управления для достижения цели Z?". Такой подход поможет ChatGPT более точно понять вашу задачу и предоставить соответствующий ответ.

Проводя некоторые выводы по данному разделу, можно сделать выводы, что ChatGPT может быть полезным инструментом для поддержки принятия решений в исследовательской работе. Он обладает способностью анализировать и интерпретировать данные, проводить вычисления и предлагать варианты решений на основе заданных критериев. Это может быть

особенно полезно при обработке больших объемов данных, когда требуется провести сложные вычисления и принять обоснованные решения.

Процесс использования ChatGPT для поддержки принятия решений включает следующие шаги:

1. **Формулировка вопроса или задачи:** Первым шагом является четкое определение вопроса или задачи, для которой требуется принять решение. Необходимо сформулировать вопрос таким образом, чтобы он был ясным и понятным для ChatGPT.
2. **Задание критериев:** далее следует задать критерии, которые будут использоваться для оценки и выбора решений. Критерии могут быть различными в зависимости от конкретной задачи и контекста исследования. Например, это может быть минимизация стоимости, максимизация производительности или достижение определенных целей.
3. **Запрос в ChatGPT:** после формулировки вопроса и задания критериев, следует задать запрос в ChatGPT. Запрос может включать описание текущей ситуации, имеющихся данных, а также задачу или цель, для которой требуется принять решение. Запрос должен быть ясным и содержательным, чтобы ChatGPT мог предоставить соответствующий ответ или предложение.

Общие примеры запросов и решений с использованием ChatGPT:

Запрос: "Какую модель машинного обучения следует выбрать для классификации медицинских изображений с наилучшей точностью при минимальных вычислительных затратах?"

Решение: ChatGPT может предложить различные модели машинного обучения, сравнить их производительность на основе доступных данных и рекомендовать модель с оптимальным сочетанием точности и вычислительной эффективности.

Запрос: "Какие факторы могут влиять на эффективность солнечных панелей?"

Решение: ChatGPT может анализировать научные исследования, данные о погоде, географическое положение и другие факторы, чтобы определить влияние различных переменных на эффективность солнечных панелей. Он может предложить рекомендации по оптимальному размещению панелей и прогнозированию их производительности.

Запрос: "Какую стратегию следует применять для оптимизации использования ресурсов в облачных вычислениях?"

Решение: ChatGPT может анализировать данные о загрузке серверов, предоставить информацию о существующих стратегиях оптимизации и предложить варианты решений, учитывая требования и ограничения исследования.

3.2 Применение чата ChatGPT на этапах разработки ПО

3.2.1 Применение ChatGPT при тестировании программного обеспечения

В данном разделе будет рассмотрено применение ChatGPT в процессе тестирования программного обеспечения (ПО). ChatGPT может быть полезным инструментом для автоматизации и оптимизации тестирования, предоставляя возможность генерировать тестовые случаи, проводить диалоговые сессии с ПО и анализировать результаты. Приведу примеры запросов в ChatGPT с полученными результатами, чтобы наглядно продемонстрировать его применение в данной области.

1. Генерация тестовых случаев:

ChatGPT может использоваться для автоматической генерации тестовых случаев, основываясь на заранее определенных правилах и спецификациях ПО. Например, можно сформулировать запрос в ChatGPT следующего вида:

Запрос: "Сгенерируй тестовый случай для регистрации нового пользователя на веб-портале."

Ответ ChatGPT: "Шаги тестового случая:

- 1) открыть веб-портал;
- 2) нажать на кнопку "Регистрация";
- 3) ввести имя пользователя, адрес электронной почты и пароль;
- 4) нажать на кнопку "Зарегистрироваться";
- 5) проверить, что новый пользователь успешно зарегистрирован и может войти в систему."

Этот пример демонстрирует, как ChatGPT может помочь в генерации шагов тестового случая на основе заданных требований. Однако, важно отметить, что результаты, сгенерированные ChatGPT, требуют последующей проверки и доработки со стороны тестировщика.

К сожалению, ChatGPT не совсем подходит для решения данной задачи так как у него нет доступа к веб-сервису, а при генерации тест кейсов необходима детализированная информация.

2. Предсказывание необходимых проверок:

ChatGPT может быть задействован в проведении диалоговых сессий с ПО для проверки его функциональности и отклонения от ожидаемого поведения. Например, можно задать следующий вопрос в ChatGPT:

Запрос: "Что произойдет, если пользователь введет неверный пароль при попытке входа в систему?"

Ответ ChatGPT: "При вводе неверного пароля система должна вывести сообщение об ошибке и предложить пользователю повторить попытку входа. Если пользователь вводит неверный пароль несколько раз подряд, его аккаунт может быть заблокирован на некоторое время."

В этом примере ChatGPT предоставляет информацию о ожидаемом поведении системы при вводе неверного пароля. Однако, необходимо учесть, что ChatGPT основан на обучающих данных, и его ответы могут быть ограничены доступными знаниями и контекстом.

Для данной задачи можно улучшить результат путём подготовки предварительных данных и уточнения подсказки:

Запрос в ChatGPT: «-- Сгенерируй возможные тест-кейсы для регистрации нового пользователя на веб-портале. --- Входные данные: email, номер телефона, пароль При регистрации необходимы все входные данные Подсказка: при генерации тест-кейсов учитывай как положительные так и негативные сценарии»

Полученный результат: «Ниже приведены возможные тест-кейсы для регистрации нового пользователя на веб-портале:

а) положительные сценарии:

- 1) зарегистрировать нового пользователя с уникальными и допустимыми email, номером телефона и паролем;
- 2) проверить, что при успешной регистрации нового пользователя он получает подтверждение о создании аккаунта;
- 3) проверить, что после успешной регистрации пользователь может войти в систему с использованием своих учетных данных.

б) негативные сценарии:

- 1) проверить, что регистрация не проходит, если одно или несколько полей (email, номер телефона, пароль) не заполнены;
- 2) проверить, что регистрация не проходит, если email или номер телефона уже зарегистрированы в системе;
- 3) проверить, что регистрация не проходит, если пароль не соответствует требованиям безопасности (недостаточная длина, отсутствие специальных символов и т.д.);
- 4) проверить, что регистрация не проходит, если формат email некорректный;

- 5) проверить, что регистрация не проходит, если формат номера телефона некорректный;
 - 6) проверить, что регистрация не проходит, если пароль содержит легко угадываемые данные, такие как имя пользователя или дату рождения;
 - 7) проверить, что регистрация не проходит, если пароль и подтверждение пароля не совпадают.
- в) дополнительные сценарии:
- 1) проверить, что при регистрации нового пользователя система отправляет подтверждение на указанный email для активации аккаунта;
 - 2) проверить, что при регистрации нового пользователя система отправляет SMS-сообщение с кодом подтверждения на указанный номер телефона для активации аккаунта;
 - 3) проверить, что пользователь может выбрать дополнительные опции, такие как подписку на рассылку или выбор языка интерфейса, при регистрации;
 - 4) проверить, что система обрабатывает регистрацию нескольких пользователей одновременно без конфликтов;
 - 5) проверить, что система предоставляет возможность восстановления пароля, если пользователь забыл его после регистрации.

Это лишь некоторые примеры тест-кейсов, которые могут быть использованы для тестирования регистрации нового пользователя на веб-портале. Конкретные тест-кейсы и их шаги зависят от конкретных требований и функциональности вашего веб-портала.»

3. Анализ результатов тестирования:

Регрессионное тестирование — это тип тестирования программного обеспечения, который выполняется для проверки, сохраняется ли функциональность системы после внесения изменений или исправления ошибок. Оно направлено на обнаружение регрессий, то есть нежелательных изменений в работе системы, которые могут возникнуть в результате новых изменений или исправлений.

В процессе разработки программного обеспечения новые функции могут быть добавлены, существующие функции могут быть изменены, а ошибки могут быть исправлены. Однако эти изменения могут нежелательно повлиять на существующую функциональность, вызвав регрессию. Регрессионное тестирование помогает выявить такие случаи.

Основные цели регрессионного тестирования включают:

- подтверждение работоспособности системы после внесения изменений;
- предотвращение появления новых ошибок или регрессий в функциональности, которая ранее работала нормально;
- подтверждение, что исправления ошибок не вызывают нежелательные побочные эффекты в других частях системы.

Смоуковое тестирование (smoke testing) — это форма поверхностного тестирования программного обеспечения, которое выполняется для быстрой проверки основных функций и наличия критических ошибок после сборки или обновления системы. Целью смоукового тестирования является быстрое выявление серьезных проблем, которые могут препятствовать дальнейшему тестированию или использованию системы.

Основные цели смоукового тестирования:

- проверка работоспособности основных функций системы;

- обнаружение критических ошибок или проблем, которые могут препятствовать продолжению тестирования или использованию системы;
- экономия времени и ресурсов путем быстрой фильтрации проблемных сборок или обновлений.

Смоуковое тестирование не заменяет полного тестирования системы, но предоставляет быструю обратную связь о работоспособности системы после внесения изменений. Это помогает ускорить процесс обнаружения проблем и улучшить качество программного обеспечения, особенно в начальной стадии тестирования.

На основе результатов регрессионного тестирования, с помощью ChatGPT можно принимать решения того, какие именно тест кейсы следует добавлять в смоуковый ран. Для этого, нейросети необходимо передать следующую информацию:

- список компонентов с описанием их взаимодействия между собой;
- список багов, в котором будут описаны компоненты, в которых была найдена ошибка;
- специфичная информация по багу в случае наличия.

Пример решения задачи:

Запрос в ChatGPT:

Составь список компонентов из предложенных необходимых для повторного тестирования (то есть в которых ранее была найдена ошибка). ---
 Контекст: в системе есть компоненты: циклическое действие, коннектора, глобальные переменные, массивы, системные поля. Компоненты, связанные между собой: Коннектора - циклическое действие Коннектора - массив Коннектора - глобальные переменные Циклическое действие - массив Баги, которые были починены перед проверкой связаны с компонентом: глобальные переменные.

Подсказка: учитывай, что если в компоненте была найдена ошибка, то в список нужно добавить связанный с ним компонент. Существует ограничение: если компонент не связан с компонентом, в котором была найдена ошибка, его добавлять не следует в список»

Полученный результат:

«Исходя из предоставленной информации и учитывая ограничение, список компонентов, необходимых для повторного тестирования (в которых ранее была найдена ошибка), включает:

- глобальные переменные;
- коннектора (так как он связан с глобальными переменными).

Остальные компоненты - циклическое действие, массивы и системные поля - не добавляются в список, так как они не связаны напрямую с компонентом, в котором была найдена ошибка (глобальные переменные).»

Распределение задач на тестировщиков:

С помощью нейросети можно автоматизировать процесс распределения времени на сотрудников.

Запрос:

Составь план регрессионного тестирования на каждого из тестировщиков на основе списка компонентов с учетом временного ограничения по 10 часов на каждого тестировщика. —

Задача: равномерно распределить время тестирования на тестировщиков, при описании компонентов для каждого тестировщика сортируй их по приоритету.

Классификация входных данных: Компонент (к) Время тестирования (вт) Приоритет тестирования компонента (птк) Ответственный за компонент (озк)

Контекст: 1. к - циклическое действие, вт - 3 ч, птк - высокий, озк - тестировщик1 2. к - коннектора, вт - 5 ч, птк - критичный, озк - тестировщик1 3. к - глобальные переменные, вт - 2 ч, птк - высокий, озк - тестировщик2 4. к

- массивы, вт - 3 ч, птк - низкий, озк - тестировщик1 5 к - системные поля, вт - 4 ч, птк - средний, озк - тестировщик2 –

Полученные результаты:

«Для равномерного распределения времени между тестировщиками исходя из списка компонентов, предлагаю следующий план тестирования:

Исходя из задачи, нужно равномерно распределить время тестирования на тестировщиков. В данном случае, всего доступно 16 часов (3 часа + 5 часов + 2 часа + 3 часа + 3 часа).

Предлагаю следующую схему распределения времени:

а) Тестировщик 1:

- 1) циклическое действие (3 часа);
- 2) глобальные переменные (2 часа);
- 3) системные поля (3 часа);
- 4) массивы (0 часов).

б) Тестировщик 2:

- 1) Коннектора (5 часов);
- 2) Массивы (3 часа).

Таким образом, каждому тестировщику будет выделено одинаковое количество времени (8 часов), и время тестирования будет равномерно распределено между компонентами.»

4. Работа с базой данных:

ChatGPT может использоваться для генерации SQL-запросов или проверки корректности запросов к базе данных. Например, можно задать следующий запрос:

Запрос: "Напишите SQL-запрос для получения списка всех пользователей, зарегистрированных после определенной даты."

Ответ ChatGPT: "SELECT * FROM users WHERE registration_date > '2022-01-01';"

В данном примере ChatGPT генерирует SQL-запрос, который позволяет получить список пользователей, зарегистрированных после определенной даты.

5. Проверки API через Postman:

ChatGPT может помочь в создании запросов для тестирования API и анализе полученных ответов. Например, можно задать следующий запрос:

Запрос: "Как проверить, что API возвращает ожидаемый статус код 200 при успешном запросе?"

Ответ ChatGPT: "Для проверки статус кода 200 можно отправить GET-запрос к API и проверить, что в ответе присутствует поле 'status' со значением '200'."

В этом примере ChatGPT предоставляет рекомендацию по проверке статус кода API при успешном запросе.

6. Создание чек-листов и работы с требованиями:

ChatGPT может быть использован для создания чек-листов и проверки соответствия требованиям ПО. Например, можно задать следующий запрос:

Запрос: "Какие проверки необходимо выполнить для требования 'При вводе неверного пароля пользователю должно выводиться сообщение об ошибке'?"

Ответ ChatGPT: "Для проверки данного требования необходимо:

- а) Ввести неверный пароль и проверить, что появляется сообщение об ошибке;
- б) Ввести верный пароль и проверить, что сообщение об ошибке не появляется;
- в) Проверить, что сообщение об ошибке отображается с корректным текстом."

В этом примере ChatGPT предоставляет чек-лист для проверки требования и указывает необходимые шаги.

7. Применение техник тест-дизайна:

ChatGPT может помочь в выборе и применении различных техник тест-дизайна, таких как метод граничных значений или комбинаторное тестирование. Например, можно задать следующий запрос:

Запрос: "Как применить метод граничных значений для тестирования формы регистрации с полем 'Возраст'?"

Ответ ChatGPT: "Для применения метода граничных значений необходимо выбрать значения на границах допустимого диапазона возраста, а именно:

- минимальное значение возраста;
- значение возраста, меньше минимального допустимого;
- значение возраста, больше максимального допустимого;
- максимальное значение возраста."

В этом примере ChatGPT объясняет применение метода граничных значений для тестирования поля "Возраст" в форме регистрации.

8. Эстимация тестирования регрессии:

ChatGPT может помочь в проведении эстимации тестирования регрессии, определяя объем работы и ресурсы, необходимые для тестирования изменений в программном продукте. Например, можно задать следующий запрос:

Запрос: "Как оценить объем работы для тестирования регрессии при внесении изменений в систему авторизации?"

Ответ ChatGPT: "Для оценки объема работы можно использовать следующие шаги:

- а) идентифицировать все компоненты, связанные с системой авторизации;
- б) определить, какие изменения внесены в каждый компонент;
- в) оценить количество и сложность тест-кейсов, связанных с изменениями;
- г) определить количество ресурсов (людей, времени) для выполнения тестирования;
- д) учесть возможность взаимодействия изменений в системе авторизации с другими компонентами и провести соответствующие тесты.

В этом примере ChatGPT предоставляет рекомендации по оценке объема работы и ресурсов для тестирования изменений в системе авторизации.

9. Автоматизация тест-кейса на авторизацию пользователя:

ChatGPT может помочь в автоматизации тест-кейсов на авторизацию пользователя с использованием различных языков программирования и инструментов. Например, можно задать следующий запрос:

Запрос: "Как автоматизировать тест-кейс на авторизацию пользователя с помощью языка Python и Selenium?"

Ответ ChatGPT: "Для автоматизации тест-кейса на авторизацию пользователя с использованием языка Python и Selenium можно использовать следующий код:

```
```python
from selenium import webdriver
Инициализация веб-драйвера
driver = webdriver.Firefox()
Открытие страницы авторизации
driver.get('https://example.com/login')
Ввод логина
```

```
login_input = driver.find_element_by_id('username')
login_input.send_keys('myusername')
Ввод пароля
password_input = driver.find_element_by_id('password')
password_input.send_keys('mypassword')
Нажатие на кнопку "Войти"
submit_button = driver.find_element_by_id('submit')
submit_button.click()
Проверка успешной авторизации
assert driver.current_url == 'https://example.com/home'
Закрытие веб-драйвера
driver.quit()
```

В данном примере ChatGPT предоставляет пример кода на языке Python с использованием библиотеки Selenium для автоматизации тест-кейса на авторизацию пользователя.

Эти примеры демонстрируют возможности применения ChatGPT в различных аспектах тестирования программного обеспечения. Однако, необходимо отметить, что для каждого конкретного случая тестирования может потребоваться дополнительная настройка и адаптация ChatGPT для получения наиболее точных и полезных результатов.

## 10. Проведение код-ревью

Один из результатов, которые можно получить это оптимизация кода. Вот несколько обоснований, почему использование ChatGPT для оптимизации кода может быть выгодным:

1. Доступ к экспертным знаниям: ChatGPT обучен на огромном объеме информации и может предоставить сведения и экспертное мнение по различным аспектам программирования и оптимизации кода. Он может помочь выявить узкие места в коде и предложить оптимальные решения для повышения эффективности.

2. Предоставление альтернативных подходов: ChatGPT может предложить альтернативные методы и подходы к решению задачи, которые могут быть более эффективными и оптимальными. Он может поделиться своими знаниями о передовых методах программирования и помочь оптимизировать код для более быстрой и эффективной работы.
3. Проверка на ошибки и проблемы: ChatGPT может помочь обнаружить и исправить ошибки и проблемы в коде. Он может выявить потенциальные проблемы с производительностью, повторяющийся или избыточный код, неэффективные алгоритмы и т. д. Это позволит улучшить работу программы и сэкономить время и ресурсы.

Пример запроса в ChatGPT:

---Оптимизируй код---

Код:

```
lst = []
lst2 = []
msg = 'Введите имя или stop для завершения:'
while (name:= input(msg)):
 lst.append(name)
 phone = input('Введите номер телефона:')
 lst2.append(phone)
print(f'{lst = }')
print(f'{lst2 = }')
```

Полученный результат:

«Оптимизация кода может включать различные аспекты, такие как повышение читаемости, эффективность и сокращение повторяющегося кода. Вот оптимизированный код на основе предоставленного фрагмента:

```
lst = []
lst2 = []
while True:
 name = input('Введите имя или stop для завершения:')
```

```

if name == 'stop':
 break
lst.append(name)
phone = input('Введите номер телефона:')
lst2.append(phone)
print(f'lst = {lst}')
print(f'lst2 = {lst2}')

```

В этом оптимизированном коде:

- убрана переменная "msg", и вызов input() перенесен прямо в цикл;
- цикл while изменен на бесконечный цикл с использованием ключевого слова True, и прерывание цикла добавлено с помощью условия if name == 'stop': break;
- форматирование строк с помощью f-строк;
- код стал более лаконичным и читаемым без необходимости дополнительных переменных.

Это более компактный и эффективный код, который сохраняет функциональность и логику оригинального фрагмента.»

### 3.2.2 Применение ChatGPT в технической поддержке программного обеспечения

Дополнительно в отдельном пункте опишу процесс технической поддержки с использованием ChatGPT

Помощь команде технической поддержки:

За счет того, что ChatGPT умеет работать с задаваемым ему контекстом, сотрудники технической поддержки могут использовать его для автоматизации процесса ответа на вопросы. Так, например, для начала мы передаём контекст нейросети: «Сообщаю тебе контекст: soarCompany supports integration with the following external systems:

- MaxPatrol SIEM (assets): 15 – 23;

- MaxPatrol VM: 25;
- MaxPatrol SIEM (incidents): 23 – 25;
- IBM QRadar: 7.2.1 - 7.3.3;
- McAfee ESM (only with MSSQL DB): 10.0 - 11.3;
- HP ArcSight: 6.9 - 7.4;
- Kaspersky Security Center (only with MSSQL DB): 10 – 14;
- Symantec Endpoint Protection (only with MSSQL DB): 12.0 - 14.0;
- McAfee ePolicy Orchestrator: 5.9 - 5.10;
- InfoWatch Device Monitor: 6.11.

It is important to note that these are the supported versions for soarCompany integration, which means the system should work properly with these versions, but it could be possible that some features or functionalities may not be available in some older versions of the systems.» Указав данную вводную информацию, можно использовать ChatGPT в качестве чат-бота для помощи клиентам в не сложных вопросах, таких как следующий. Запрос «с какой версией Max-Patrol сможет интегрироваться soarCompany для получения информации по активам?»»

Полученный результат: «Согласно предоставленной информации, soarCompany может интегрироваться с MaxPatrol SIEM для получения информации по активам в диапазоне версий 15-23. Это означает, что для успешной интеграции soarCompany требуется установленная и работающая версия MaxPatrol SIEM, которая находится в этом диапазоне версий.»

При этом стоит отметить, что до указания вводной информации ChatGPT не смог ответить на данный вопрос, а лишь выдал поверхностную информацию: «На момент моего обновления в сентябре 2021 года, последней версией MaxPatrol была версия 8. Но я не имею информации о более новых версиях, выпущенных после моего обновления.»

Что касается интеграции soarCompany с MaxPatrol для получения информации по активам, то точные требования интеграции и совместимости

зависят от конкретной версии MaxPatrol и ее возможностей. Для получения более точной информации о возможности интеграции soarCompany с MaxPatrol и требованиях к версии MaxPatrol, рекомендуется обратиться к официальной документации или поставщику MaxPatrol.».

Генерация SQL- запросов.

В данном пункте не буду повторяться с частью из тестирования. Стоит лишь упомянуть, что техническая поддержка часто собирает информацию о продуктовых багах перед тем, как передать их в департамент разработки, и данная возможность должна упростить сбор данных.

Bash-скрипты.

У сотрудников второй линии поддержки задачи сложнее чем написать sql запрос или ответить на вопрос из документации. Часть их работы – помогать в устранении неполадок или сбор информации о существующих проблемах в инфраструктуре системы. С данными вводными был сделан запрос к сотрудникам технической поддержки с просьбой выдать несколько задач, которые им следует решить. Вот что из этого вышло:

1. Запрос: «Напиши bash скрипт: Содержание скрипта: выделение из исходной строки подстроки с границами, заданными порядковыми номерами символов в исходной строке. Усложнение: предусмотреть возможность не выделения, а удаления подстроки. Основные средства: команда cut, переменные оболочки.»

Полученный результат: «Конечно! Вот пример Bash-скрипта, который выполняет выделение или удаление подстроки из исходной строки с помощью команды `cut`:

```
```bash
```

```
#!/bin/bash
```

```
# Исходная строка
```

```
source_string="Hello, World!"
```

```
# Позиции начала и конца подстроки (порядковые номера символов)
```

```

start_position=2
end_position=8
# Флаг для определения операции (выделение или удаление)
operation="extract"
if [ "$operation" = "extract" ]; then
    # Выделение подстроки
    result=$(echo "$source_string" | cut -c "$start_position"-"$end_position")
else
    # Удаление подстроки
    result=$(echo "$source_string" | cut -c 1-"$start_position-1")$(echo
"$source_string" | cut -c "$end_position"-)
fi
# Вывод результата
echo "Результат: $result"
...

```

В этом скрипте мы используем переменные `source_string`, `start_position` и `end_position` для задания исходной строки и границ подстроки. Переменная `operation` позволяет выбрать операцию (выделение или удаление). Если `operation` равно "extract" (выделение), то скрипт выделит подстроку. Если `operation` имеет любое другое значение, то будет выполнено удаление подстроки.

С помощью команды `cut` и опции `-c` мы вырезаем подстроку, указывая диапазон символов с `start_position` по `end_position`. Результат сохраняется в переменную `result`, которая выводится на экран.

Вы можете изменить значения переменных `source_string`, `start_position`, `end_position` и `operation` в скрипте, чтобы адаптировать его под свои потребности.»

Запрос: «Напиши bash скрипт: Содержание скрипта: выдача сразу трех временных штампов для заданного имени файла. Основные средства:

переменные оболочки, `gawk`, командная подстановка. Усложнение: обработка произвольного числа имен, реакция на ошибки.»

Полученные результаты: «Конечно! Вот пример Bash-скрипта, который выдает три временных штампа (создания, последнего доступа и последнего изменения) для заданного имени файла. Скрипт также поддерживает обработку произвольного числа имен файлов и реагирует на ошибки:

```
```bash
#!/bin/bash
Проверяем, передано ли имя файла в аргументах командной строки
if [$# -eq 0]; then
 echo "Ошибка: не указано имя файла."
 exit 1
fi
Проходимся по каждому аргументу (имени файла)
for filename in "$@"; do
 # Проверяем, существует ли файл
 if [! -e "$filename"]; then
 echo "Ошибка: файл '$filename' не существует."
 continue
 fi
 # Получаем временные штампы файла
 creation_timestamp=$(gawk -v file="$filename" 'BEGIN {stat=file; stat_info=stat
" --printf="%W\n\n"; system(stat_info)}')
 access_timestamp=$(gawk -v file="$filename" 'BEGIN {stat=file; stat_info=stat "
--printf="%X\n\n"; system(stat_info)}')
 modification_timestamp=$(gawk -v file="$filename" 'BEGIN {stat=file;
stat_info=stat " --printf="%Y\n\n"; system(stat_info)}')
 # Выводим результаты
 echo "Имя файла: $filename"
 echo "Время создания: $creation_timestamp"
```



своими руками. Теперь рассмотрим пример того как можно решать такие задачи самой простой сложности.

Запрос: « Example request structure:

```
{
"data": {
"type": "comment",
"attributes": {
"text": "Lorem #ipsum dolor sit ..."
}
}
}
```

--- Напиши пример Python скрипта для данной структуры ---»

Полученный результат: «Certainly! Here's an example Python script that creates a JSON payload based on the given structure:

```
```python
import json
# Create the data structure
data = {
    "data": {
        "type": "comment",
        "attributes": {
            "text": "Lorem #ipsum dolor sit ..."
        }
    }
}
# Convert the data to JSON
json_payload = json.dumps(data)
# Print the JSON payload
print(json_payload)
```
```

In this script, we use the `json` module to work with JSON data. We create a Python dictionary `data` that represents the JSON structure. The dictionary is then converted to a JSON string using `json.dumps()`.

Finally, we print the JSON payload, which represents the provided structure.

You can run this script to generate the JSON payload. Feel free to modify the content within the `data` dictionary to match your specific needs.»

### 3.3 Экономическое обоснование

Для оценки экономической эффективности применения были определены процессы, в которые было принято решение внедрить ChatGPT как инструмент для помощи в принятии решении.

Так как практическое применение нейросети вместе в внедрением проводилось в отделе контроля качества, то будем рассматривать процессы в данной области.

Сначала рассмотрим то как эти процессы выглядят на предприятии до внедрения ChatGPT:

1. Регрессионное тестирование. Регрессионное тестирование направлено на подтверждение работоспособности системы и ее стабильности, чтобы убедиться, что изменения не негативно повлияли на уже существующую функциональность.

Общий процесс регрессионного тестирования выглядит следующим образом:

- Идентификация тестовых случаев: На этом этапе определяются тестовые случаи, которые проверяют основные функции и особенности системы. Они могут быть выбраны из предыдущих тестов или основываться на функциональных требованиях и ожидаемом поведении системы.
- Создание тестовых данных: Создаются необходимые данные для выполнения тестовых случаев. Это может включать подготовку тестовых сценариев, настройку окружения или подготовку тестовых баз данных.
- Выполнение тестовых случаев: Тестировщики выполняют заранее определенные тестовые случаи, используя созданные тестовые данные.

Они проверяют функциональность системы и сравнивают результаты с ожидаемыми.

- **Регистрация и анализ результатов:** Результаты выполнения тестовых случаев регистрируются и анализируются. Обнаруженные проблемы или ошибки фиксируются и передаются разработчикам для исправления.
- **Исправление проблем и повторное тестирование:** Разработчики исправляют обнаруженные проблемы, а затем выполняется повторное регрессионное тестирование, чтобы убедиться, что исправления не вызвали новые проблемы или регрессии.

2. Смоуковое тестирование. Целью смоукового тестирования является быстрое выявление серьезных проблем, которые могут препятствовать дальнейшему тестированию или использованию системы.

Общий процесс регрессионного тестирования выглядит следующим образом:

- **Определение основных функций:** На этом этапе определяются основные функции и требования к системе, которые должны быть проверены. Это может включать основные операции, функции ввода-вывода, взаимодействие с пользователем и другие важные компоненты.
- **Создание смоуковых тестов:** Тестировщики разрабатывают набор смоуковых тестов, которые позволяют проверить основные функции системы. Это могут быть короткие и простые сценарии, выполняющие основные операции, и проверки наличия ключевых компонентов.
- **Выполнение смоуковых тестов:** На этом этапе тестировщики выполняют смоуковые тесты в соответствии с предварительно разработанными сценариями. Они проверяют, что основные функции работают без существенных ошибок и что система загружается и функционирует должным образом.

- Анализ результатов: После выполнения смоуковых тестов результаты анализируются. Обнаруженные проблемы или ошибки регистрируются и передаются разработчикам для исправления.
- Принятие решения: На основе результатов смоукового тестирования принимается решение о том, можно ли продолжать более полное и глубокое тестирование системы. Если обнаружены серьезные проблемы, тестировщики и разработчики могут принять решение о приостановке тестирования до исправления этих проблем.

3. Предварительный анализ ТЗ. Основная цель предварительного анализа ТЗ - обеспечить полное понимание требований и предвидеть потенциальные проблемы, связанные с тестированием.

Вот основные шаги процесса предварительного анализа ТЗ со стороны тестирования:

- Изучение ТЗ: Тестировщики внимательно изучают ТЗ, включая все разделы, требования и описания функциональности системы. Они стремятся полностью понять цели, ожидания и ограничения проекта.
- Выделение основных требований: На этом этапе тестировщики определяют основные требования и функции системы, которые будут тестироваться. Это включает определение основных сценариев использования, функций, модулей и взаимодействия с другими системами или компонентами.
- Определение приоритетов и рисков: Тестировщики анализируют приоритеты требований и оценивают риски, связанные с тестированием каждого требования. Это помогает определить основные области для тестирования и приоритезировать тестовые задачи.
- Определение стратегии тестирования: На основе выделенных требований и приоритетов тестировщики определяют стратегию

тестирования. Это включает выбор методов, подходов и инструментов для тестирования, а также планирование различных типов тестирования, таких как функциональное, интеграционное, нагрузочное и т. д.

- **Выявление неоднозначностей и противоречий:** При анализе ТЗ тестировщики могут обнаружить неоднозначности, противоречия или недостаточно четко сформулированные требования. В таких случаях они должны согласовать эти вопросы с заказчиком или командой разработки для уточнения требований.
- **Подготовка тест кейсов:** На основе анализа ТЗ тестировщики создают тест кейсы, включающие описание тестовых сценариев, ожидаемые результаты и критерии приемки. Это помогает структурировать процесс тестирования и обеспечивает понятность и целостность тестовых активностей.

В результате предварительного анализа ТЗ со стороны тестирования формируется более четкое понимание требований к системе и плана тестирования. Это позволяет оптимизировать процесс тестирования, обеспечить более эффективное покрытие требований и предупредить возможные проблемы, связанные с неполными или неоднозначными требованиями.

4. **Код-ревью.** Код-ревью является важной частью разработки программного обеспечения, поскольку помогает повысить его надежность, читаемость и обслуживаемость.

Вот основные шаги процесса код-ревью:

- **Подготовка кода для ревью:** Разработчик, который желает получить код-ревью, готовит свой код для обзора. Это включает коммит или отправку изменений в систему контроля версий, чтобы другие разработчики могли ознакомиться с ним.

- **Запрос на ревью:** Разработчик отправляет запрос на код-ревью своему коллеге или команде разработчиков. Это может быть сделано с помощью системы управления задачами или специальных инструментов для код-ревью.
- **Обзор кода:** Разработчики, получившие запрос на код-ревью, внимательно изучают код и анализируют его. Они обращают внимание на правильность реализации требований, соответствие стандартам кодирования, понятность и эффективность алгоритмов, наличие ошибок и потенциальных проблем.
- **Предоставление обратной связи:** Разработчики, проводящие код-ревью, предоставляют обратную связь автору кода. Они могут комментировать участки кода, указывать на ошибки или недочеты, предлагать улучшения и альтернативные решения. Обратная связь обычно оформляется в виде комментариев в системе контроля версий или в инструментах для код-ревью.
- **Обсуждение и исправление:** Разработчик, получивший обратную связь, обсуждает комментарии и предложения с рецензентами кода. Он может задавать вопросы, уточнять детали и принимать решения о внесении исправлений в свой код. Исправления обычно вносятся в новых коммитах или патчах.
- **Завершение ревью:** После внесения необходимых исправлений и устранения замечаний ревью процесс считается завершенным. Код может быть принят или отправлен на повторное ревью в случае необходимости.

То каким образом применить нейросеть было описано в разделе выше, в данной главе будем рассматривать насколько экономически обосновано её применение в этих процессах. Средняя зарплата тестировщика в организации равна ста двадцати тысячам рублей. Исходя из этого были посчитаны

расходы под выделенные процессы до и после внедрения ChatGPT и построен график предоставленный на рисунке 8.

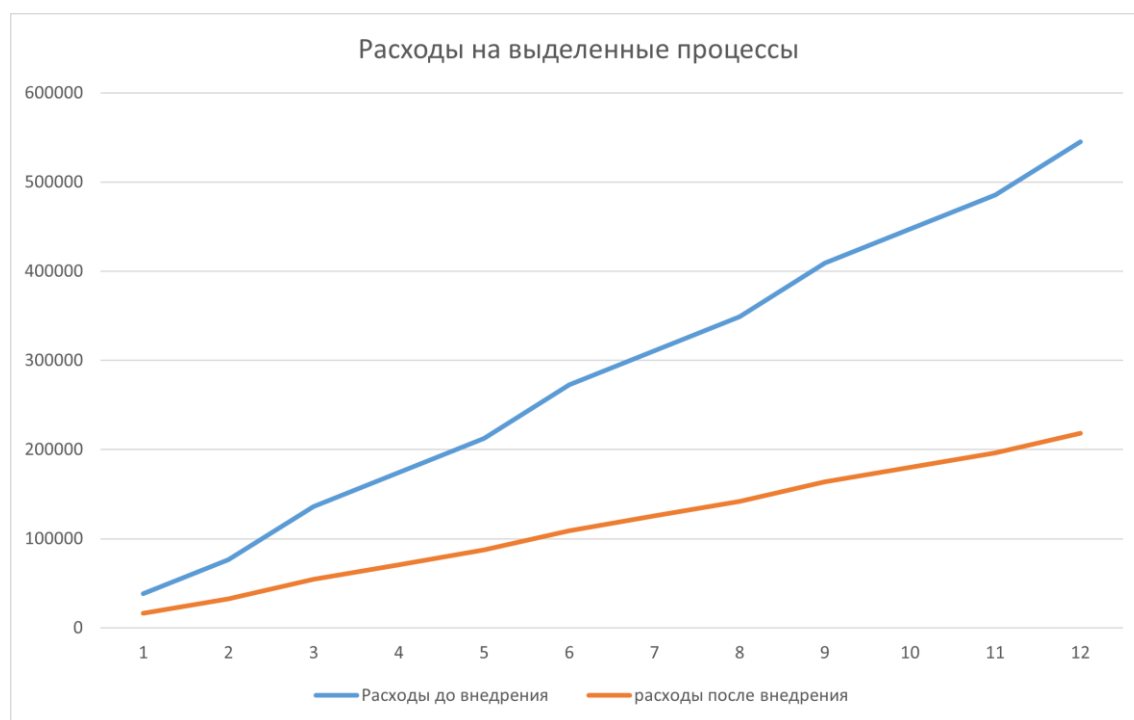


Рисунок 8 – график расходов на выделенные процессы

Прогнозируя расходы на 12 месяцев учитывая опыт за тот же период прошлого года, можно сделать вывод, что предполагаемые расходы уменьшаться на 288 тысяч рублей за год. Для наглядности был построен график экономии с нарастающим итогом, который представлен на рисунке 9. Глядя на него видно, что срок окупаемости равен примерно двум месяцам. Расходы на внедрение равны примерно 80 тысячам рублей, а временной интервал, за который можно внедрить ChatGPT не мешая другой работе достигает около двух месяцев.

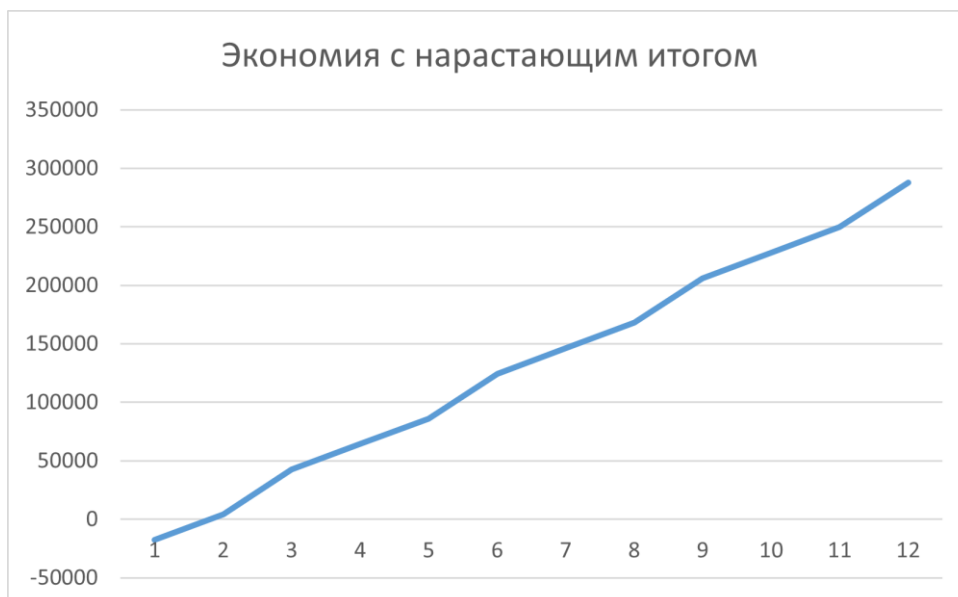


Рисунок 9 – график экономии с нарастающим итогом

Дополнительно была посчитана рентабельность инвестиций (ROI) по доступным данным. Для расчета воспользуемся формулой:

$$ROI = (\text{Прибыль от инвестиций} / \text{Инвестиции}) * 100$$

Подставляя значения рентабельность инвестиций оказалась равна 360. Это означает, что прибыль от инвестиций составляет 360% от вложенных средств.

## ЗАКЛЮЧЕНИЕ

В заключении хотелось бы оценить полученные результаты по поставленным целям. Начну с заключения по цели о поддержке принятия решений в научно-исследовательской работе.

Преимущества ChatGPT в контексте научных исследований существенны и важны для улучшения процессов и результатов научных исследований:

1. Расширение возможностей исследования: ChatGPT может служить инструментом для расширения возможностей исследования. Он может помочь в формулировании идей, предложении новых подходов и методов, а также предоставить дополнительные инсайты и перспективы.
2. Генерация новых идей: ChatGPT может служить творческим инструментом, способствующим генерации новых идей и концепций. Он может помочь исследователям искать варианты решений, формулировать новые гипотезы и разрабатывать инновационные подходы.
3. Поддержка при обработке больших объемов данных: ChatGPT может помочь исследователям в обработке и анализе больших объемов данных. Он может автоматизировать процессы обработки данных, выявление паттернов и тенденций, а также предоставить средства для визуализации и интерпретации результатов. В данном пункте всё же стоит сделать упоминание про ограничения, но о них дальше

При использовании ChatGPT в научных работах существуют ограничения, связанные с его способностью полностью понять контекст и задачу исследования. Несмотря на значительные преимущества модели, она имеет свои ограничения, которые следует учитывать:

1. Ограничения контекста: ChatGPT может иметь ограниченное понимание длинных и сложных контекстов. Если задача требует учета большого

объема информации или сложной логики, модель может испытывать трудности в полном понимании и учете всех аспектов контекста.

2. Зависимость от формулировки вопроса: ChatGPT чувствителен к формулировке вопросов и задачи. Небольшое изменение в формулировке вопроса может привести к значительному изменению ответа. Поэтому важно аккуратно и ясно формулировать вопросы, чтобы получить соответствующие и релевантные ответы.
3. Неучтенность внешних знаний: ChatGPT, хотя и обладает обширными знаниями, не всегда может учесть актуальные или специфические внешние знания, которые могут быть важны для конкретной исследовательской задачи. Исследователям может потребоваться дополнительная работа по интеграции таких знаний в процесс работы модели.
4. Неполное понимание задачи исследования: В связи с ограничениями модели в понимании широкого контекста, ChatGPT может недостаточно полно понимать задачу исследования, что может повлиять на точность и полноту генерируемых ответов.
5. Необходимость проверки и анализа полученных результатов. При использовании ChatGPT в научных работах важно осуществлять проверку и анализ полученных результатов. Хотя ChatGPT обладает значительной гибкостью и способностью генерировать текст, есть несколько аспектов, которые требуют внимания:

1. Проверка точности: важно проверять точность генерируемого текста, особенно при научных исследованиях, где требуется высокая степень достоверности и точности информации. Проверка может включать сопоставление с известными фактами, проверку на противоречия и анализ логической связности.

2. Оценка релевантности: хотя ChatGPT способен генерировать текст, он не всегда может гарантировать полную релевантность исследовательской

задаче. Важно проанализировать, насколько генерируемые ответы соответствуют поставленным вопросам и задачам исследования.

3. Анализ стиля и формы: ChatGPT может иметь тенденцию к повторениям, непоследовательности и генерации "безумных" ответов. При анализе результатов необходимо обратить внимание на стиль, грамматическую правильность и общую логику генерируемого текста.

4. Зависимость от входных данных: Результаты, полученные с использованием ChatGPT, могут зависеть от входных данных, таких как контекст и формулировка вопросов. Изменение входных данных может привести к различным ответам. Поэтому важно проводить анализ результатов с учетом входных параметров и применять систематический подход к проверке.

В целом, необходимость проверки и анализа полученных результатов является неотъемлемой частью работы с ChatGPT в научных исследованиях. Это поможет обеспечить точность, релевантность и надежность полученных данных, а также учесть особенности работы модели.

5. Ограничения по качеству генерируемого текста. Например, это могут быть ошибки и неточности: ChatGPT может генерировать текст, содержащий ошибки, неточности или неправильные утверждения. Это может быть вызвано ограничениями в данных обучения или недостаточным контролем на этапе генерации. Также в данном пункте необходимо упомянуть о тенденции к совпадению с обучающими данными: ChatGPT имеет тенденцию к повторению и воспроизведению информации из своих обучающих данных. Это может ограничивать его способность генерировать новые и оригинальные идеи.

Все эти ограничения требуют осторожного подхода к использованию ChatGPT в научных работах. Исследователи должны проводить критическую

оценку и проверку генерируемого контента, а также осуществлять дополнительные шаги для повышения качества и достоверности результатов

Хотя ChatGPT демонстрирует высокий потенциал, существуют некоторые ограничения, которые необходимо учитывать при интерпретации результатов исследований.

В целях повышения уровня достоверности и качества получаемых результатов, следует учитывать эти ограничения и принимать соответствующие меры:

- проводить дополнительную проверку и верификацию результатов, особенно в случаях, где точность и достоверность данных имеют особое значение;
- использовать дополнительные источники информации и методы для подтверждения результатов, полученных с помощью ChatGPT;
- обеспечивать качественные и разнообразные данные для работы с моделью, чтобы уменьшить возможность искажений и предвзятости.

Оценка уровня достоверности и качества результатов, полученных с использованием ChatGPT является важной задачей для исследователей, и требует внимательного анализа и проверки.

Далее можно проанализировать результаты полученные по цели о поддержке принятия решений в разработке программного обеспечения.

На основе проведенного исследования и анализа литературы можно сделать следующие выводы относительно использования ChatGPT в тестировании ПО и технической поддержке клиентов:

1. Улучшение эффективности тестирования ПО: ChatGPT может быть использован для автоматизации процесса тестирования ПО. Он может генерировать тестовые сценарии, обрабатывать результаты тестов и предоставлять быстрые обратные связи разработчикам. Это позволяет ускорить процесс тестирования и повысить его эффективность.

2. Повышение качества технической поддержки клиентов: ChatGPT может служить важным инструментом для предоставления технической поддержки клиентам. Он способен автоматически отвечать на часто задаваемые вопросы, предоставлять инструкции по устранению проблем и помогать пользователям решать технические вопросы. Это позволяет повысить качество обслуживания и снизить нагрузку на службу поддержки.
3. Ограничения: однако использование ChatGPT в тестировании ПО и технической поддержке клиентов также сопряжено с определенными вызовами и ограничениями. Например, ChatGPT может сталкиваться с проблемами понимания контекста, неправильной интерпретацией вопросов и предоставлением неточных или неправильных ответов. Кроме того, необходимо обеспечить качественное обучение и поддержку ChatGPT для достижения оптимальных результатов.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Chater N. Human rationality and the psychology of reasoning: Where do we go from here? / N. Chater, M. Oaksford // *British Journal of Psychology*. – 2001. – Vol. 92. – № 1. – P. 193–226
2. Cushman F. Rationalization is rational / F. Cushman F // *Behavioral and Brain Sciences*. – 2020. – Vol. 43
3. Jost J.T. The role of stereotyping in system-justification and the production of false consciousness / J.T. Jost, M.R. Banaji // *British journal of social psychology*. – 1994. – Vol. 33. – № 1. – P. 1–27.], анализе предвзятости суждений [Nickerson R.S. Confirmation bias: A ubiquitous phenomenon in many guises / R.S. Nickerson // *Review of General Psychology*. – 1998. – Vol. 2(2). – P. 175–220
4. Kunda Z. The case for motivated reasoning / Z. Kunda // *Psychological Bulletin*. – 1990. – Vol. 108(3). – P. 480–498
5. Stanovich K.E. The comprehensive assessment of rational thinking / K.E. Stanovich // *Educational Psychologist*. – 2016. – Vol. 51. – № 1. – P. 23–34
6. Kahneman D. A psychological point of view: Violations of rational rules as a diagnostic of mental processes / D. Kahneman // *Behavioral and Brain Sciences*. – 2000. – Vol. 23. – P. 681–683
7. Thaler R.H. From cashews to nudges: The evolution of behavioral economics / R.H. Thaler // *American Economic Review*. – 2018. – Vol. 108. – № 6. – P. 1265–87
8. Barberis N. A survey of behavioral finance / N. Barberis, R. Thaler // *Handbook of the Economics of Finance*. – 2003. – Vol. 1. – P. 1053–1128
9. Kahneman D. Choices, values and frames / D. Kahneman, A. Tversky // *New York : Cambridge University Press*, 2000
10. Goldstein D.G. Models of ecological rationality: the recognition heuristic / D.G. Goldstein, G. Gigerenzer // *Psychological review*. – 2002. – Vol. 109. – № 1. – P. 75

11. Rieskamp J. Ecological rationality / J. Rieskamp, T. Reimer // Encyclopedia of social psychology. – 2007. – Vol. 273. – P. 275
12. Todd P.M. Ecological rationality: Intelligence in the world / P.M. Todd, G.E. Gigerenzer. – Oxford University Press, 2012
13. Dryzek J.S. Ecological rationality / J.S. Dryzek // International Journal of Environmental Studies. – 1983. – Vol.7(21). – №. 1. – P. 5–10
14. Cleaver F. Moral ecological rationality, institutions and the management of common property resources / F. Cleaver // Development and change. – 2000. – Vol. 31. – № 2. – P. 361–383.
15. Агабекян Р. Л. Институциональная экономика. Бизнес и занятость / Р.Л. Агабекян, Г.Л. Баян-дурян. - М.: ИЛ, 2018. - 464 с.], поведенческая экономика [Simon H.A. Behavioral Economics // The New Palgrav / ed. by J. Eatwell, M. Milgate and P. Newman. N.Y.: W.W. Norton, 1987
16. Goldstein D.G. Models of ecological rationality: the recognition heuristic / D.G. Goldstein, G. Gigerenzer // Psychological review. - 2002. - Vol. 109. - № 1. - P. 75
17. Инновационный менеджмент / под ред. Л.Н. Оголевой. – М.: Инфра-М, 2003. – 238 с
18. Мирских И.Ю. Правовое регулирование интеллектуальной собственности как условие развития инновационных процессов // Исторические, философские, политические и юридические науки, культурология и искусствоведение. Вопросы теории и практики. - 2013. - № 10-1 (36). - С. 141-143
19. Сурин, А.В. Инновационный менеджмент: Учебник / А.В. Сурин, О.П., Молчанова. М.: ИНФРА-М, 2008.
20. Мироненков К.Н. Понятие инновационного цикла и его использование в управлении инновационной деятельностью организации // Корпоративный менеджмент, 2008. № 6. – С. 9496.], проведение пробного маркетинга продукта [Маркова В.Д. Маркетинг инноваций. Научно-

практическое издание / В.Д. Маркова [и др.]. – Новосибирск: Новосибирский государственный университет, 2005. 109 с.

21. Кетова Н.П. Реализация стратегий маркетинга инноваций: предпосылки и условия поддержания конкурентоспособности российских компаний в условиях экономического кризиса // Теория экономики и управления народным хозяйством. 2009. № 9. С. 9197.] [Короткова Т.Л. Роль маркетинга в коммерциализации инноваций // Практический маркетинг. 2010. № 3. С. 1016

22. Мироненков К.Н. Понятие инновационного цикла и его использование в управлении инновационной деятельностью организации // Корпоративный менеджмент, 2008. № 6. – С. 9496

23. Самолдин А.Н. Проведение маркетинговых исследований в области инновационной деятельности с учетом социально-экономических аспектов // Труд и социальные отношения. 2014. № 7. С. 5459

24. Рассел С., Норвиг П. Искусственный интеллект: современный подход. - Вильямс, 2006

25. Искусственный интеллект в современном мире / С.П. Турбин, Н.В. Картечина, Д.А. Шевякова, А.П. Турбина // В сборнике: Инженерное обеспечение инновационных технологий в АПК. Материалы Международной научно-практической конференции. Мичуринск-наукоград РФ, 2021. С. 226-228

26. Абалуев Р.Н., Завражнов А.А., Ланцев В.Ю. Разработка алгоритма работы специализированного программного обеспечения для систематизации посевной техники для овощных и пропашных культур // Восточно-Европейский научный журнал. 2021. No 10-2 (74). С. 30-40

27. Искусственный интеллект, принятие решений и моральная оценка [Электронный ресурс] - Режим доступа: <https://www.scu.edu/ethics/focus-areas/technology-ethics/resources/artificial-intelligence-decision-making-and-moral-deskilling/>, свободный. - (дата обращения: 23.04.2022)

28. Deep learning for improving the storage process: accurate and automatic segmentation of spoiled areas on apples / N. Stasenko, G. Ovchinnikov, E. Chernova, D. Shadrin, M. Pukalchik, I. Krivolapov // В сборнике: Conference Record - IEEE Instrumentation and Measurement Technology Conference. Сер. "I2MTC 2021 - IEEE International Instrumentation and Measurement Technology Conference, Proceedings" 2021. С. 9460071.6.

29. Искусственный интеллект и принятие решений: как это работает? [Электронный ресурс] Режим доступа: [https://codernet.ru/articles/drugoe/iskusstvennyii\\_intellekt\\_i\\_prinyatie\\_reshenij\\_kaketo\\_rabotaet/](https://codernet.ru/articles/drugoe/iskusstvennyii_intellekt_i_prinyatie_reshenij_kaketo_rabotaet/), свободный. - (дата обращения: 12.04.2022)

30. Реализация проектов машинного обучения и искусственного интеллекта / В.Н. Торицына, н.В. Картечина, Т.К. Яшина, В.П. Васильев // В сборнике: Инженерное обеспечение инновационных технологий в АПК. Материалы Международной научно-практической конференции. Мичуринск-научкоград РФ, 2021. С. 224-225

31. Проектирование нейросетевой модели поведенческого анализа обучающегося электронной образовательной среды Moodle / Р.Н. Абалуев, А.И. Кочетыгов, А.М. Дорохова, В.А. Шацкий // Наука и Образование. 2021. Т. 4. No2

32. Алиев Р.А. Применение искусственного интеллекта в управлении жизненным циклом продукции // Информационные технологии в управлении. - 2019. - № 2. - С. 84-97

33. Gerber, A. Key concepts and skills for getting started in IoT / Anna Gerber, Jim Romeo // IBM

34. Методы прогнозирования спроса [Электронный ресурс] Режим доступа: <https://arxiv.org/abs/2306.03763> (дата обращения: 12.04.2022)

35. Сегаран Т. Программируем коллективный разум. - Пер. с англ. - СПб: Символ-Плюс, 2008. - 20 с

36. Ежов А.А. Нейрокомпьютинг и его применения в экономике и бизнесе / А.А. Ежов, С.А. Шумский. - М.: МИФИ, 1998. - 222 с.

37. Гаранин С.И., Попова Н.В. Анализ применения машинного обучения в задачах управления жизненным циклом продукции // Интернет-журнал «Научные статьи студентов и аспирантов». - 2018. - Т. 2. - С. 56-61
38. Основы естественного языка NLP [Электронный ресурс] Режим доступа: <https://arxiv.org/abs/2204.10483> (дата обращения: 12.04.2022)
39. Солуянов А. В., Юркин Ю. В. Исследование математической модели Call-центра на железнодорожном транспорте, - 2012.
40. Смыслова, Л. В. Чат-бот как современное средство интернет-коммуникаций / Л. В. Смыслова. - Текст: непосредственный // Молодой ученый. - 2018. - № 9 (195).- С. 36-39
41. Касевич В.Б. «Элементы общей лингвистики»// М. 1977 г.
42. ФлахП. Машинное обучение. М.: ДМК Пресс, 2015. 400 с
43. LeCun Y., Bengio Y., Hinton G. Deep Learning // Nature. 2015. Vol. 521. P. 436-444. DOI: 10.1038/nature14539
44. Этапы работы с естественным языком NLP Основы естественного языка NLP [Электронный ресурс] Режим доступа: <https://arxiv.org/abs/2210.14826> (12.04.2022)
45. Лебедев Г.С., Маслюков А.П., Шадеркин И.А., Шадеркина А.И. Глубокое машинное обучение (искусственный интеллект) в ультразвуковой диагностике // Журнал телемедицины и электронного здравоохранения. 2020. № 2. С. 22-29. DOI:10.29188/2542-2413-2020-6-2-22-29
46. Марахтанов А.Г., Паренченков Е.О., Смирнов Н.В. Определение электронного мошенничества методами машинного обучения в случае несбалансированного набора данных // Вестник пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. 2020. №36. С. 80-95
47. Рунова К.В., Юрин А.А. Классификация сердечно-сосудистых заболеваний с помощью инструментальных методов обработки информации на основе различных методов машинного обучения // Colloquium-journal. 2019. № 13-3(37). С. 115-120

48. Фомин В.В., Александров И.В. Об одном опыте применения web - инструментария машинного обучения // Моделирование и анализ сложных технических и технологических систем: сборник статей по итогам Международной научно-практической конференции. Самара. 2018. С. 131-137.] [ Saif M.A., Medvedev A.N., Medvedev M.A., Atanasova T. Classification of online toxic comments using the logistic regression and neural networks models // AIP Conference Proceedings 2048, 060011 (2018). DOI:10.1063/1.5082126
49. Shtovba S., Shtovba O., Yahymovych O., Petrychko M. Impact of the syntactic dependencies in the sentences on the quality of the identification of the toxic comments in the social networks // Scientific works of Vinnytsia national technical university. 2019. № 4. Pp. 35-42. DOI: 10.31649/2307-5392-2019-4-35-42
50. Обучение алгоритмам NLP [Электронный ресурс] Режим доступа: <https://arxiv.org/abs/2208.08367> (12.04.2022)
51. Метода обработки естественного языка NLP [Электронный ресурс] Режим доступа: <https://arxiv.org/abs/2210.10903> (12.04.2022)
52. Задачи естественного языка NLP [Электронный ресурс] Режим доступа: <https://arxiv.org/abs/2303.14576> (12.04.2022)
53. Принципы обучения естественного языка NLP [Электронный ресурс] Режим доступа: <https://arxiv.org/abs/2306.04980> (12.04.2022)
54. Jay Alammar (2020) — How GPT-3 Works: Visualizations and Animationsf
55. Сухин И.Г. Тенденции в применении цифровых технологий в зарубежном образовании // Современные технологии в мировом научном пространстве. - 2019. - С. 81-87
56. Prompt engineering – Википедия. [Электронный ресурс] Режим доступа: URL: [https://en.wikipedia.org/wiki/Prompt\\_engineering](https://en.wikipedia.org/wiki/Prompt_engineering)] (12.04.2022)
57. Методы взаимодействия с языковыми моделями [Электронный ресурс] Режим доступа: URL:<https://arxiv.org/abs/2306.01798> (12.04.2022)

58. Методики создания подсказок [Электронный ресурс] Режим доступа: <https://arxiv.org/abs/2303.03548> (12.04.2022)
59. Общие советы по созданию подсказок [Электронный ресурс] Режим доступа: <https://arxiv.org/abs/2303.07142> (12.04.2022)
60. Hugo Touvron, Thibaut Lavril, Gautier Izacard et al. “LLaMA: Open and Efficient Foundation Language Models” . URL: <https://arxiv.org/pdf/2302.13971.pdf> [1. Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, Dario Amodei “Scaling Laws for Neural Language Models” URL: <https://arxiv.org/abs/2001.08361>
61. Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, Denny Zhou “Chain-of-Thought Prompting Elicits Reasoning in Large Language Models” URL: <https://arxiv.org/abs/2201.11903>] [1. Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, Yusuke Iwasawa ”Large Language Models are Zero-Shot Reasoners” URL: <https://arxiv.org/abs/2205.11916>
62. Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed H. Chi, Sharan Narang† Aakanksha Chowdhery, Denny Zhou “Self-consistency improves chain of thought reasoning in language models” URL: <https://arxiv.org/pdf/2203.11171.pdf>
63. подсказка сгенерированных знаний (generated knowledge prompting) : В статье [Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, Hannaneh Hajishirzi “Generated Knowledge Prompting for Commonsense Reasoning” URL: <https://arxiv.org/pdf/2110.08387.pdf>
64. Общее понятие трансформеров [Электронный ресурс] Режим доступа: <https://arxiv.org/abs/1706.03762> (дата обращения 12.04.23)
65. Архитектура трансформеров [Электронный ресурс] Режим доступа: <https://arxiv.org/abs/2306.00708> (дата обращения 12.04.23)

66. «Тонкая настройка» работы LLM[Электронный ресурс] Режим  
доступа: <https://arxiv.org/abs/2306.04488>(дата обращения 12.04.23)