Аксенов В.В.
Научный руководитель: к.т.н., доцент Рыжкова М.Н.
Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264 г. Муром, Владимирской обл., ул. Орловская, д. 23
a.vladimir@mail.ru

Алгоритм действий бота при уклонении от препятствий

С развитием и массовым внедрением квадрокоптеров всё большее значение приобретает качественная подготовка операторов. Особенно важно отрабатывать действия в нестандартных и стрессовых ситуациях, которых сложно добиться на статичных тренажёрах.

В данной работе предлагается интеллектуальный бот (игровой ИИ) для симулятора БПЛА, который реалистично имитирует поведение человека при обнаружении дрона. Бот работает на основе циклического алгоритма, объединяющего восприятие среды, оценку угрозы и выбор действий. Разработка ведётся в среде Unreal Engine 5 с использованием штатных систем ИИ, что позволяет обойтись без сбора данных и сложного программирования.

Алгоритм бота построен как непрерывный цикл. Сначала система восприятия (зрительная и звуковая) регистрирует присутствие дрона. Зрительный канал замечает аппарат на средних и близких дистанциях, а звуковой улавливает шум моторов – причём услышать дрон можно раньше, чем увидеть. Затем на основе удалённости, громкости и наличия препятствий вычисляется общий уровень угрозы. В зависимости от этого уровня бот выбирает одну из трёх стратегий поведения:

- если угроза низкая, бот просто проявляет осторожность: поворачивается в сторону источника, замедляется, но продолжает патрулирование;
- при средней угрозе он переходит к активному уклонению: ищет укрытие, оценивая окружающее пространство, а если укрытия нет – начинает двигаться зигзагом, чтобы сбить с толку наблюдателя;
- в случае высокой угрозы бот впадает в состояние «паника» – он немедленно убегает от дрона, используя ближайшее укрытие или хаотично меняя направление движения.

Все эти действия прерываются и заменяются более подходящими, если ситуация внезапно меняется.

После того как дрон удалился или угроза снизилась на продолжительное время, бот плавно возвращается к обычному патрулированию. Такая схема делает поведение бота естественным, разнообразным и непредсказуемым для обучаемого. Тренировка с таким ботом помогает оператору быстрее привыкать к резким изменениям обстановки и принимать верные решения в стрессовых условиях.

Практическая реализация использует стандартные инструменты Unreal Engine: Perception System (настройка зрения и слуха), Environment Query System (интеллектуальный поиск укрытий) и Behavior Tree (логика выбора действий). Все параметры можно менять прямо в редакторе без перекомпиляции кода, что упрощает настройку под разные сценарии. Разработанный бот повышает реалистичность тренировок, создавая вариативные и правдоподобные сценарии взаимодействия, что и является основным результатом работы.

Дополнительным преимуществом предложенного подхода является его гибкость: бота можно быстро адаптировать под разные типы занятий – от начальной выработки навыков до отработки действий в критических ситуациях. Например, инструктор может вручную менять чувствительность сенсоров бота, его склонность к панике или дальность поиска укрытий прямо во время тренировки, подстраивая уровень сложности под конкретного оператора. В перспективе алгоритм может быть расширен для поддержки нескольких ботов одновременно, что позволит моделировать групповое поведение или соревновательные сценарии. Таким образом, разработанный бот не только решает задачу реалистичного уклонения от БПЛА, но и предоставляет платформу для создания широкого спектра обучающих ситуаций без доработки программного кода.

Литература

1. Андриевский Б. Р., Попов А. М., Михайлов В. А., Попов Ф. А. Применение методов искусственного интеллекта для управления полетом беспилотных летательных аппаратов // Аэрокосмическая техника и технологии. 2023. №2. URL: https://cyberleninka.ru/article/n/primenenie-metodov-iskusstvennogo-intellekta-dlya-upravleniya-poletom-bespilotnyh-letatelnyh-apparatov (дата обращения: 02.04.2026).

2. Галкин Д.В., Петухов И.В., Танрывердиев И.О., Стешина Л.А., Стешин И.С., Курасов П.А. Разработка симулятора для обучения операторов беспилотных летательных аппаратов // Современные наукоемкие технологии, № 10, 2024. С. 27-31. URL: https://s.top-technologies.ru/pdf/2024/10/40167.pdf (дата обращения: 01.04.2026).

Андронов И.А.
Научный руководитель: к.т.н., доцент кафедры ФПМ Астафьев А.В.
Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264 г. Муром, Владимирской обл., ул. Орловская, д. 23
E-mail:h3r1zma@gmail.com

Применение мультиагентной архитектуры на основе больших языковых моделей для автоматизации генерации и рецензирования технических статей

С ростом объёма технической документации и потребности в качественном контенте для корпоративных баз знаний, блогов и образовательных платформ становится актуальной задача автоматизации процессов создания, проверки и публикации статей. Традиционный подход предполагает работу автора, редактора и юриста, что требует значительных временных и кадровых ресурсов. Использование больших языковых моделей (LLM) в сочетании с мультиагентной архитектурой позволяет автоматизировать этот конвейер, сохраняя контроль на каждом этапе. Исходя из этого, тема исследования является актуальной научно-технической задачей.

Целью исследования является разработка и практическая проверка мультиагентной системы автоматической генерации, рецензирования и публикации технических статей с использованием фреймворка Semantic Kernel и подхода RAG (Retrieval-Augment Generation).

Для достижения поставленной цели необходимо выполнить следующие задачи:

1) Проектирование мультиагентной архитектуры с выделением специализированных ролей: автор, юрист, фактчекер, SEO-аналитик, редактор.
2) Реализация конвейера генерации статей с поэтапной обработкой: создание черновика, параллельное рецензирование, применение правок, публикация.
3) Интеграция подхода RAG для детектирования дубликатов и связывания смежных публикаций.
4) Разработка веб-интерфейса администрирования с поддержкой ручного и автоматического режимов работы.
5) Экспериментальная проверка системы на задаче генерации статей по тематике администрирования ОС RedOS и Linux.

Проектирование мультиагентной архитектуры основывается на декомпозиции процесса подготовки статьи на независимые этапы, каждый из которых обслуживается отдельным LLM (Large Language Model)-агентом. Агент-автор (WritingAgent) генерирует черновик статьи по заданной теме. Агенты-рецензенты – юридический (LegalAgent), фактчекинговый (FactCheckingAgent) и SEO-агент (SeoAgent) – выполняют параллельный анализ черновика. Агент-редактор (EditorAgent) обобщает рецензии и вносит правки в исходный текст. Каждый агент реализован как наследник базового класса BaseLlmAgent и получает конфигурацию (модель, температуру, системный промпт) из централизированного хранилища настроек.

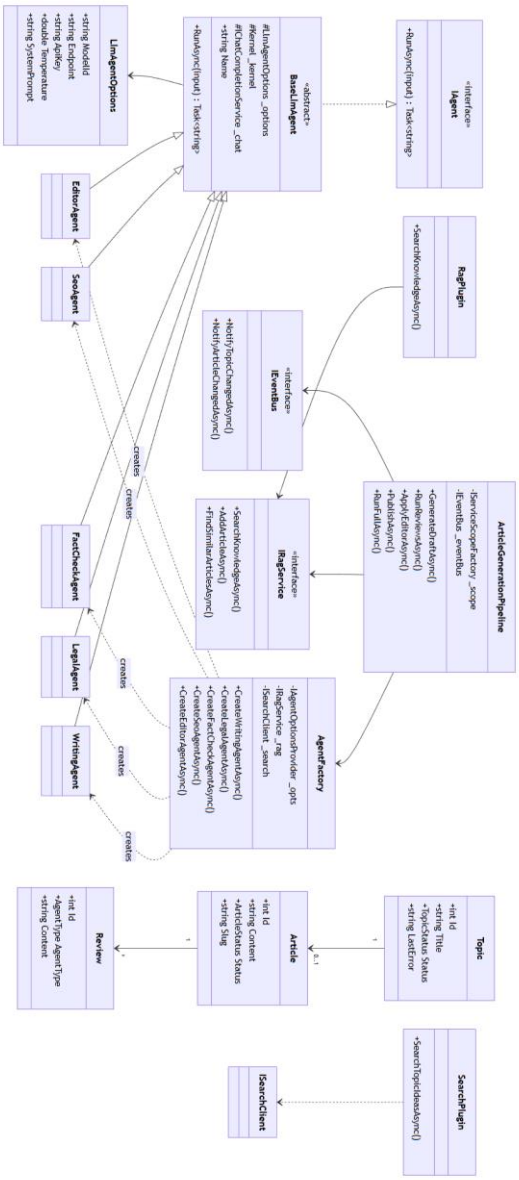
Конвейер генерации реализован в классе ArticleGenerationPipeline и декомпозирован на четырем метода: GenerateDraftAsync, RunReviewsAsync, ApplyEditorAsync и PublishAsync. В ручном режиме (ReviewMode.Manual) пользователь контролирует каждый этап: запускает рецензирование, выбирает, какие рецензии применить, и инициирует публикацию. В автоматическом режиме (ReviewMode.Auto) полный цикл выполняется без участия человека. Обработка ошибок осуществляется через статус Failed с сохранением текста ошибки в поле LastError сущности Topic.

Подход RAG используется для двух целей: детектирования дубликатов перед генерацией новой статьи и формирования перекрёстных ссылок между смежными публикациями. Опубликование статьи индексируется в in-memory хранилище, а при генерации нового черновика агент-автор через Semantic Kernel Plugin обращается к базе знаний для поиска семантических близких материалов.

Веб-интерфейс реализован на платформе Blazor Server (ASP.NET Core) и включает панель администрирования с разделами управления очередью тем, справочником статей, редактором статей с Markdown-предпросмотром и настройками агентов. Обновления статусов

отображается в реальном времени через in-memory шину событий (EventBus), что обеспечивается архитектурой SignalR, лежащей в основе Blazor Server.

Результатом исследования является работающий прототип системы Evidentrix, демонстрирующий применимость мультиагентного подхода к автоматизации полного цикла подготовки технических статей – от поиска тем до публикации с контролем качества.



Литература

1. Semantic Kernel Documentation[Электронный ресурс]: URL: https://learn.microsoft.com/en-us/semantic-kernel/ (дата обращения: 05.04.2026)

2. Lewis P. et al. Retrieval-Augment Generation for Knowledge-Intensive NLP Tasks [Электронный ресурс]: URL: https://arxiv.org/abs/2005.11401 (дата обращения: 05.04.2026)

3. ASP.NET Core Blazor [Электронный ресурс]: URL: https://learn.microsoft.com/en-us/aspnet/core /blazor/ (дата обращения: 05.04.2026)

Астраханцев И.А., Крекина С.А.
Научный руководитель: к.т.н. Кутарова Е.И., к.т.н. Рыжкова М.Н.
Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264 г. Муром, Владимирской обл., ул. Орловская, д. 23
E-mail: ewan.astrakhantsev@yandex.ru, skrekina@bk.ru

Пробный ЕГЭ по информатике на базе Муромского института как инструмент предэкзаменационной подготовки школьников

По данным Федерального института педагогических измерений (ФИПИ), средний балл ЕГЭ по информатике в 2025 году составил 70,2, при этом примерно 35% участников не сдали экзамен, в значительной мере вследствие стресса и незнания процедур проведения. Пробные экзамены снижают данный показатель на 25–30%, поскольку позволяют отработать не только предметные навыки, но и формат, регламент и техническое сопровождение государственной итоговой аттестации[1].

Пробный ЕГЭ по информатике, проведённый на базе Муромского института, был организован в условиях, максимально приближённых к реальному экзамену. В исследовании участвовали 25 школьников 11-х классов Муромского района, которым был предложен вариант пробного экзамена, соответствующий структуре и продолжительности ЕГЭ-2026 по информатике.

Целью пробного экзамена являлась не только проверка предметных знаний по информатике, но и формирование у школьников представлений о реальных условиях проведения ЕГЭ: использовании защищённой платформы, идентификации, жёсткого контроля временного режима и ограничения доступа к внешним устройствам и программам. Работа проводилась на платформе «Яндекс Учебник» — бесплатном онлайн-сервисе для подготовки к ЕГЭ по информатике и математике, содержащем актуальный банк заданий 2026 года, встроенный ИИ-помощник и готовые варианты, структурно соответствующие демоверсии ЕГЭ.

Процедура проведения пробного ЕГЭ

1. Запуск экзамена

Школьники авторизовывались в системе под индивидуальным логином и проходили идентификацию, подтверждающую личность. После успешной проверки доступа экзамен запускался автоматически, а на экране открывался интерфейс с обратным таймером, отображающим оставшееся время выполнения работы, и блокировкой функций скриншотов и обмена сообщениями, что имитировало условия реальной государственной итоговой аттестации.

2. Структура экзаменационной работы

Экзаменационная работа соответствовала общей продолжительности 235 минут (3 часа 55 минут), что соответствует формату ЕГЭ-2026 по информатике. Работа включала две части:

Часть 1. Теоретические задания (12 заданий, ориентировочно 60 минут). В этой части учащиеся выполняли задания с кратким ответом: тесты с выбором одного или нескольких вариантов, установление соответствий, задачи по алгоритмам, базам данных, информационным сетям и кодированию информации.

Часть 2. Задания по программированию (15 заданий, ориентировочно 175 минут). Вторая часть включала задачи различного уровня сложности — от простых циклических программ и обработки строк до сложных заданий на графы, рекурсию и алгоритмы поиска.

3. Завершение работы и проверка результатов

После истечения времени или досрочного завершения экзаменационной работы данные автоматически сохранялись в платформе «Яндекс Учебник», после чего участники завершали сессию под контролем организаторов. Проверка заданий осуществлялась в два этапа:

- на первом этапе платформа выполняла автоматическую проверку ответов, используя алгоритмы, сопоставимые с методами, применяемыми ФИПИ при оценке работ, и формировало баллы по каждому заданию;

- на втором этапе преподаватели института проводили дополнительную экспертную проверку наиболее сложных задач программирования.

Окончательные результаты выдавались участникам сразу после завершения проверки с указанием набранного балла и порогового значения (46 баллов), что способствовало формированию у школьников реалистичного представления об уровне подготовки к ЕГЭ по информатике.

Проведённый пробный ЕГЭ по информатике на базе Муромского института с использованием платформы «Яндекс Учебник» позволил создать условия, близкие к реальной государственной итоговой аттестации.

Анализ результатов представленных в таблице 1 показал, что 15 участников (60%) набрали достаточный балл для прохождения условного порога, тогда как 10 участников (40%) не достигли требуемого уровня. Большинство трудностей пришлось на задачи программирования (например, задания 2, 6, 7, 10, 15, 16, 18, 19, 20 представленные в таблице 2), которые оказались наиболее сложными для значительной части школьников. Использование платформы «Яндекс Учебник» позволило не только смоделировать экзаменационные условия, но и активизировать обучающий потенциал пробного экзамена: школьники познакомились с ИИ-помощником, актуальным банком заданий и форматом автоматической проверки, что повысило осознанность их подготовки.

Таблица 1. Детальное распределение баллов

Диапазон баллов	Сдали (n=15)	Не сдали (n=10)	Общий %
0–20	0	4	16
21–46	0	6	28
46–70	9	0	32
71+	6	0	24

Таблица 2. Анализ по блокам заданий

Номера заданий, вызвавших сложность	2	6	7	10	15	16	18	19	20
Кол-во не выполнивших	10	11	12	11	11	13	13	11	10
% не выполнивших	40	44	48	44	44	52	52	44	10

Литература

Бонкало Татьяна Ивановна, Полякова Ольга Борисовна, Гребенникова Вероника Михайловна Проблемное поле современных исследований учебного стресса студентов колледжей: мета-анализ публикаций scopus 2024 // Проблемы современного педагогического образования. 2025. №86-1.

Витюк В.Р.
Научный руководитель: канд. техн. наук, доцент Астафьев А.В.
Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264 г. Муром, Владимирской обл., ул. Орловская, д. 23
E-mail: vityuk-05@mail.ru

Разработка цифровой онлайн-лаборатории интерактивного обучения в области информационной безопасности

В настоящее время ИТ-отрасль в России всё больше связана с ускоренными темпами импортозамещения и, как следствие, ростом числа угроз информационной безопасности. Последние данные свидетельствуют о росте объёма рынка информационной безопасности в 2025 году [1. 2]. Данная тенденция создаёт спрос на высококвалифицированные кадры в области информационной безопасности. В результате одним из самых перспективных направлений в области технологий обучения стало внедрение виртуальных лабораторий. Виртуальные лаборатории позволяют моделировать угрозы в существующих решениях безопасных сред [3]. Ключевым недостатком существующих решений является большой упор на теоретическую сторону сферы, во многом без учёта специфики российских криптографических стандартов.

Целью работы является проектирование и разработка цифровой онлайн-лаборатории «Кибер.Практикум». Ключевым в данном проекте являются фокус на обучении методам отечественной криптографии, а также оценка при помощи искусственного интеллекта. Исходя из цели необходимо: разработать веб-платформу на современном фреймворке; реализовать модули для визуализации алгоритмов ГОСТ, которые позволят не только понимать теорию, но и предоставят функционал для визуализации изученного; интегрировать ИИ-агентов, позволяющих студенту получать материал в удобном формате; внедрить модель искусственного интеллекта, направленную на оценку студента без использования малорезультативных подходов.

К научной новизне проекта можно отнести использование поведенческих паттернов каждого студента для создания объективного цифрового профиля компетенций. «Кибер.Практикум», в отличие от традиционных платформ [4], для автоматического анализа логов активности будет использовать нейронную сеть. Данные активности будут хранится в базе. Это позволит оценивать не только взаимодействие с элементами в реальном времени, но также производить глубокий анализ логики принятия решений, частоту взаимодействия с интерактивными элементами и количество обращений к подсказкам.

Набор технологических решений проекта будет включать в себя Python для реализации логики работы элементов сайта. Данное решение позволит пользователям в режиме реального времени наблюдать за преобразованием данных при работе с представленными алгоритмами шифрования. Серверная часть будет реализована при помощи фреймворка Django, обеспечивающего простую разработку и быстрое взаимодействие клиент-сервер. Инновацией проекта являются ИИ-агенты, которые действуют как личные наставники. Заложенный системный промт гарантирует, что ИИ-агент будет наводить пользователя на решение через косвенные вопросы и взаимодействие с интерактивными элементами имитируя тем самым работу живого методиста [5].

Анализ рыночных ниш и существующий тренд на цифровизацию лабораторий доказывает бизнес-потенциал разработки [6, 7]. Замена зарубежных тренажеров, интеграция в образовательный процесс и корпоративные программы повышения квалификации специалистов информационной безопасности [8] являются приоритетными целями в разработке данной платформы. Развертывание системы на государственных платформах обеспечит соблюдение требований российского законодательства и защищенную технологическую независимость.

Разрабатываемый курс будет рассматривать в минимальном исполнении три раздела: криптография по ГОСТ, программирование сетевых протоколов, а также криптография в

клиент-серверных приложениях. Криптография по ГОСТ будет состоять из трёх обширный подразделов, а именно: Симметричные алгоритмы шифрования, ассиметричные алгоритмы шифрования и хеширование. Каждый раздел и подраздел будет включать в себя интерактивные элементы, а также ИИ-ассистента настроенного под тематику проходимого курса. На рисунке 1 карта курса исходя из описанного:

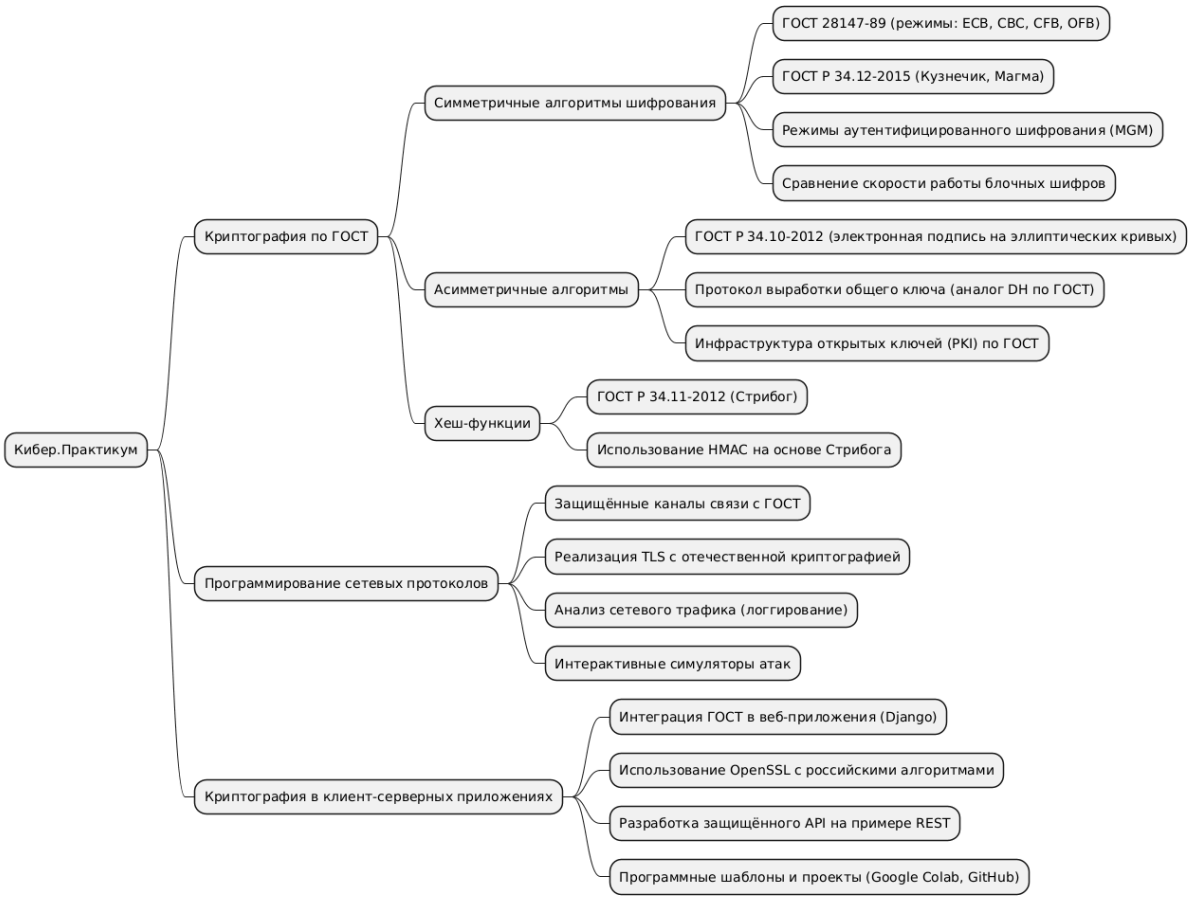


Рис. 1 — Карта курса

Литература

1. Рынок ИБ в России в 2025 году вырос почти на четверть. [Электронный ресурс]. URL: https://clck.ru/3SvKUh (дата обращения: 26.03.2026).

2. Обзор ключевых тенденций ИТ-рынка: что ждет отрасль в 2026 году [Электронный ресурс]. URL: https://clck.ru/3SvKaW (дата обращения: 27.03.2026).

3. Global Virtual Laboratories Platform Market Size, Growth Analysis & Global Forecast 2026-2034 [Электронный ресурс]. URL: https://clck.ru/3SvKcK (дата обращения: 27.03.2026).

4. «Лаборатория Касперского» запустила в России платформу с онлайн-тренингами для ИТ- и ИБ-специалистов — Kaspersky Cybersecurity Training. [Электронный ресурс]. URL: https://clck.ru/3SvUPm (дата обращения: 29.03.2026).

5. Обзор трендов EdTech 2025 года. [Электронный ресурс]. URL: https://clck.ru/3SvKNx (дата обращения: 30.03.2026).

6. Эффективность вместо гонки за ростом: как изменился рынок EdTech в 2025-м [Электронный ресурс]. URL: https://clck.ru/3SvVeJ (дата обращения: 31.03.2026).

7. Оценка рынка PAM-TAM-SAM-SOM. [Электронный ресурс]. URL: https://clck.ru/3SvVea (дата обращения: 31.03.2026).

8. The Cyber Lab Market size is set to grow rapidly over the forecast period from 2026 to 2033, at a CAGR of 4.9%. [Электронный ресурс]. URL: https://clck.ru/3SvVek (дата обращения: 01.04.2026).

Гарин Я.О.
Научный руководитель: к.т.н., доцент кафедры ФМП Астафьев А.В.
Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264 г. Муром, Владимирской обл., ул. Орловская, д. 23
E-mail: garinyaroslav13@mail.ru

Разработка интернет-платформы поэтапного контроля выполнения курсовых работ

Одной из распространённых проблем современных студентов является разработка курсовых работ не поэтапно, а в сжатые сроки перед самым последним сроком. Это приводит к недостаточной проработке темы, что затрудняет освоение материала. Преподаватели, в свою очередь, сталкиваются с трудностями, так как получают работы на проверку только в последний момент, что ограничивает их возможности для корректировки. Согласно исследованию Маковецкой Е. Н. [1], основными причинами являются нехватка самодисциплины и неумение организовать время.

Разрабатываемая платформа поэтапного контроля учебных проектов решает эту проблему, предоставляя механизм формирования этапов разработки. Эти этапы соотносятся с разделами пояснительной записки, позволяя студентам сдавать работу постепенно и получать своевременную обратную связь от преподавателя. Исходя из этого, тема исследования является актуальной научно-технической задачей.

Целью является разработка интернет-платформы поэтапного контроля выполнения курсовых работ.

Для достижения поставленной цели необходимо выполнить следующие задачи:

1. Обзор предметной области и анализ текущих процессов обучения в образовательных учреждениях.

2. Разработка структуры базы данных, соответствующая требованиям системы.

3) Программная реализация веб-приложения платформы поэтапного контроля учебных проектов на основе разработанной модели базы данных.

В ходе проведения работы над проектом была разработана модель базы данных веб-платформы, которая представлена на рисунке 1. Модель спроектирована с учётом ролевой модели доступа, поддержки версионности документов и автоматизированного контроля прогресса учебных проектов. Она включает сущности пользователей, учебных групп, шаблонов работ, поэтапных элементов пояснительной записки, истории изменений, дедлайнов и расписания защит.

На основе созданной модели реализовано веб-приложение, использующее клиент-серверную архитектуру. В разрабатываемой интернет-платформе реализованы такие модули как: управления пользователями, создания проектов и шаблонов, поэтапной сдачи разделов пояснительной записки, системы комментариев преподавателя.

Приложение успешно прошло функциональное тестирование и в настоящее время представляет собой рабочую систему, готовую к использованию в учебном процессе.

Рис. 1. Структура базы данных

В ходе выполнения работы достигнута поставленная цель — разработана интернет-платформа поэтапного контроля выполнения курсовых работ. Решены все поставленные задачи: проанализирована предметная область, определены требования к системе и спроектирована структура базы данных, реализована рабочее веб-приложение.

Полученные результаты позволяют автоматизировать процесс поэтапной сдачи разделов пояснительной записки, обеспечить своевременную обратную связь и повысить качество выполнения курсовых работ.

Литература

1. Маковецкая Е.Н. ФЕНОМЕН АКАДЕМИЧЕСКОЙ ПРОКРАСТИНАЦИИ В СТУДЕНЧЕСКОЙ СРЕДЕ // Образовательные ресурсы и технологии. – 2023. – № 4 (45). – С. 7-15. doi: 10.21777/2500-2112-2023-4-7-15

Гусев М.А.
Научный руководитель: к.т.н., доцент Астафьев А.В.
Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264 г. Муром, Владимирской обл., ул. Орловская, д. 23
e-mail: g.max01@mail.ru

Построение системы комплексной защиты веб-версии ИИ-сервиса технической поддержки «РедКопилот»

Стремительное внедрение ИИ-ассистентов в автоматизацию технической поддержки создаёт новые векторы кибератак. В условиях курса на технологический суверенитет и перехода на отечественные операционные системы (Ред ОС) возникает потребность в защите веб-интерфейсов RAG-систем, обрабатывающих персональные данные пользователей и корпоративные базы знаний. Существующие подходы к информационной безопасности требуют адаптации под специфику Large Language Models (LLM), включая риски инъекций промптов, отравления данных векторных хранилищ и неограниченного потребления ресурсов API. Целью работы является разработка и обоснование архитектуры комплексной защиты веб-версии ИИ-сервиса «РедКопилот», обеспечивающей конфиденциальность, целостность и доступность данных на всех этапах жизненного цикла приложения с учётом требований регуляторов РФ и международных стандартов безопасности ИИ.

Сервис развёрнут на инфраструктуре виртуального выделенного сервера (VPS, Ubuntu) и реализован на стеке Python/Django. В основе лежит RAG-архитектура: векторная СУБД Qdrant хранит эмбеддинги базы знаний по администрированию Ред ОС (модель intfloat/multilingual-e5-small, 384-мерные векторы), а генерация ответов осуществляется через защищённый API GigaChat. Клиентский уровень защищён санитизацией вывода (DOMPurify), серверный – встроенным security-middleware Django и обратным прокси Nginx. Для выявления критических уязвимостей на основе Банка данных угроз ФСТЭК России и OWASP Top 10 for LLM 2025 проведён анализ угроз. Из 23 первоначально выявленных угроз 12 признаны актуальными (коэффициент реализуемости $Y \geq 75\%$). Критическими (ранг 1) определены: угроза приведения системы в состояние «отказ в обслуживании» (УБИ.140/LLM10), использование уязвимых версий ПО (УБИ.192/LLM03) и незащищённое администрирование облачных услуг (УБИ.055). Специфическими для ИИ-компонентов признаны: нарушение функционирования средств ИИ через инъекции промптов (УБИ.220/LLM01), модификация модели путём искажения обучающих данных (УБИ.221/LLM04) и хищение векторных представлений базы знаний (УБИ.219/LLM08). Разработана модель недопустимых событий, устанавливающая пороговые значения ущерба: простой сервиса более 30 минут, утечка персональных данных более 100 записей, финансовые потери свыше 50 тыс. руб.

Для нейтрализации выявленных угроз предложена многоуровневая система безопасности. На сетевом и прикладном уровне реализовано межсетевое экранирование (UFW, политика Default-Deny), ограничение частоты запросов на базе алгоритма Token Bucket (Nginx limit_req_zone и кэш Django), защита от CSRF/XSS с автоматическим экранированием шаблонов и заголовками безопасности, принудительное использование HTTPS (TLS 1.3, сертификаты Let's Encrypt). Аутентификация построена на гибридной модели OAuth 2.0 через Яндекс ID с выдачей stateless JWT-токенов, разграничением прав на уровне представлений Django и изоляцией секретов в файле .env, исключённом из Git-репозитория. Защита ИИ-компонентов включает валидацию входных данных на уровне приложения (Regex-фильтрация ключевых слов атак), жёсткие системные инструкции для LLM, механизм цитирования источников в RAG-цепочке для снижения вероятности галлюцинаций и подмены контекста, а также фильтрацию вывода модели перед рендерингом. В рамках DevSecOps применяется фиксация зависимостей в requirements.txt с точными версиями, контроль целостности кода через Git-ветвление (main/develop/feature/*) с детерминированными контрольными точками верификации перед слиянием, автоматизированное ежедневное резервное копирование на уровне хостинг-провайдера и структурированное логирование событий безопасности (Django Logging). На рисунке 1 представлена архитектурная схема обработки запроса с контрольными

точками безопасности, отражающая последовательное прохождение запроса через уровни защиты: ограничение частоты, валидацию JWT-сессии, Regex-фильтрацию промпта, векторный поиск в Qdrant, генерацию ответа через GigaChat API и структурированное логирование.

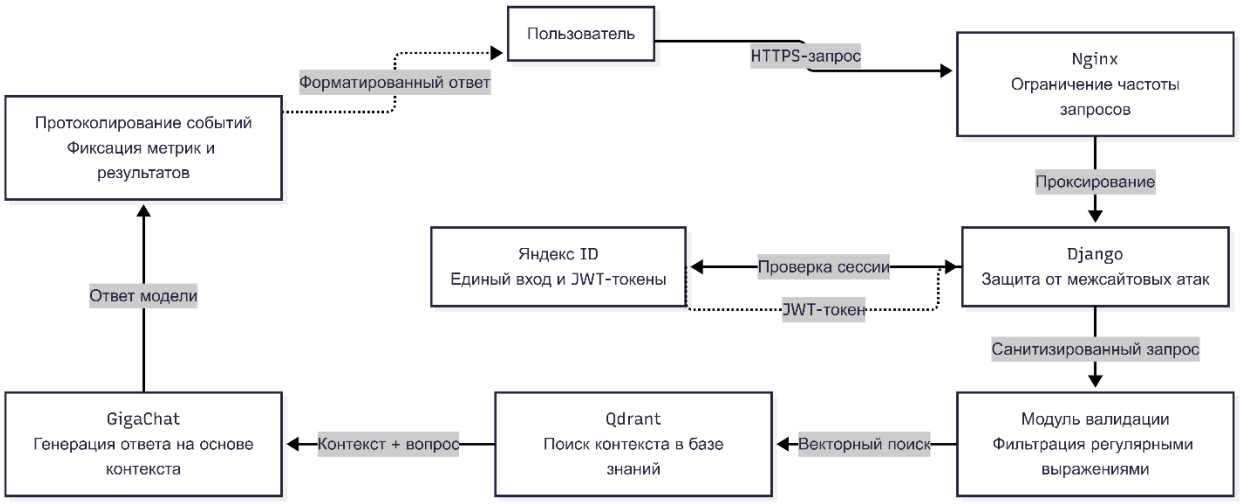


Рис. 1. Архитектурная схема обработки запроса с контрольными точками безопасности

Выбранные средства защиты полностью совместимы с открытым стеком технологий, не требуют приобретения дорогостоящих лицензий и адаптированы под развёртывание на VPS. Реализованная архитектура обеспечивает защиту как от традиционных веб-угроз (XSS, CSRF, DoS), так и от специфических атак на LLM-системы (prompt injection, data poisoning, неограниченное потребление). Сопоставление угроз ФСТЭК с рисками OWASP подтвердило необходимость интеграции отечественных нормативных требований и международных практик безопасности ИИ. Разработанные модели и конфигурации готовы к программной реализации, нагрузочному тестированию и внедрению в импортозамещённые инфраструктуры технической поддержки. Комплексный подход к защите RAG-систем, сочетающий многозвенную архитектуру, валидацию промптов, криптографическую изоляцию секретов и принципы Secure SDLC, позволяет нейтрализовать критические векторы атак на ИИ-ассистентов. Результаты работы могут быть масштабированы на другие веб-сервисы автоматизации, функционирующие в экосистеме отечественного программного обеспечения.

Денисова П.А.
Научный руководитель: к.т.н., доцент кафедры ФПМ Астафьев А.В.
Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264 г. Муром, Владимирской обл., ул. Орловская, д. 23
E-mail:polina.denisova.2013@gmail.com

Исследование влияния размера контекста и порядка подачи информации на качество RAG-системы для ИИ-ассистента «РедКопилот» технической поддержки операционной системы РедОС

В настоящее время активно развиваются системы дополненной генерации (Retrieval Augmented Generation, RAG) [1], объединяющие семантический поиск по базе знаний и генерацию ответов с помощью больших языковых моделей. ИИ-ассистент на основе RAG сначала находит релевантные документы в базе знаний, а затем генерирует ответ на запрос пользователя на основе этих данных. Однако качество их функционирования существенно зависит от объёма контекста и устойчивости модели к посторонней информации. Для разработки ИИ-ассистента необходимо понимание того, как эти факторы влияют на итоговое качество ответов.

Целью работы является оценка влияния размера контекста и порядка подачи чанков на метрики поиска и генерации в RAG-системе для последующего применения при разработке ИИ-ассистента «РедКопилот» технической поддержки операционной системы РедОС.

Для проведения экспериментов использовалась база знаний РедОС, содержащая 1700 документов, и тестовый набор из 200 вопросов, сгенерированных с помощью большой языковой модели GigaChat. Оценка производилась по метрикам поиска (Precision@1, Precision@5, MRR, Recall@5) и генерации (Faithfulness, Answer Relevance, Context Recall, Semantic Similarity).

В ходе исследования изучалось влияние размера контекста при использовании 3, 5 и 10 чанков. Оценка метрик поиска и генерации представлена на рисунке 1. Оптимальным для ИИ-ассистента оказалось значение k=5, при котором Recall@5 достигает 93%, а время ответа минимально и составляет 3,204 с. При увеличении контекста до 10 чанков происходит сильное падение качества поиска, Recall@5 снижается с 93% до 61,9%. Это означает, что при передаче в модель слишком большого объема информации релевантные документы начинают теряться среди шума, и система хуже справляется с их поиском, что некорректно для работы ИИ-ассистента.

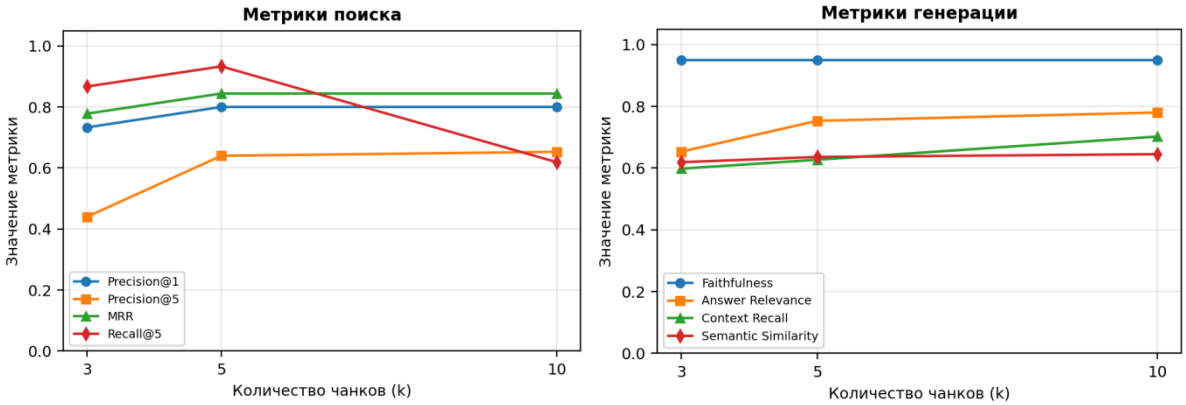


Рис.1. Оценка метрик поиска и генерации

Также проверялась гипотеза о том, что большая языковая модель не использует весь контекст равномерно. Эксперименты показали, что 60% вопросов чувствительны к порядку чанков. При обратном порядке подачи среднее сходство ответов с эталоном составляет всего

40%. Добавление нерелевантного чанка в начало контекста снижает качество ответов в среднем на 50%. Было выявлено, что шум в начале контекста оказывает наиболее разрушительное воздействие, тогда как шум в конце наименьшее.

Полученные результаты демонстрируют, что большая языковая модель не способна эффективно фильтровать нерелевантную информацию и фокусируется преимущественно на первых чанках контекста. Исследование для ИИ-ассистента «РедКопилот» показало, что оптимальный размер контекста k=5. Также рекомендуется фильтровать посторонние фрагменты перед подачей в модель. Важно ранжировать документы по релевантности, чтобы ключевая информация была в начале контекста.

Эти рекомендации помогут улучшить качество работы ИИ-ассистента и уменьшить вероятность ошибочных или неполных ответов.

Литература

1. Keith Bourne Unlocking Data with Generative AI and RAG. - 1 изд. - Birmingham: Packt Publishing, 2024. - 346 с.

Марков М.А.
Научный руководитель: к.с.н. Смолина Н.В.
Муромский институт (филиал) федерального государственного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264, г. Муром, Владимирская обл., ул. Орловская, 23

Визуализация решения заданий с графиками функций в ЕГЭ по профильной математике

В ЕГЭ по профильной математике одно из заданий заключается в нахождении точек пересечения различных графиков. Однако школьникам сложно мысленно достраивать графики функций. Поэтому цель нашей работы — проиллюстрировать решение задания с различными графиками с помощью программного инструмента, разработанного на языке Python.

Для достижения этой цели мы создали приложение с графическим интерфейсом, которое позволяет в реальном времени строить графики функций, изменять их параметры и автоматически вычислять точки пересечения.

Основные возможности программы.

1. Построение четырёх типов графиков:

- Две параболы вида $y = ax^2 + bx + c$
- Гипербола вида $y = \frac{a}{x+b} + c$.
- Прямая линия вида $y = kx + b$

2. Интерактивное управление:

- Пользователь может вводить коэффициенты функций в специальные поля на панели управления.
- Кнопки «Применить» обновляют графики с новыми параметрами.
- Каждый график можно скрыть или показать, чтобы не загромождать экран.

3. Автоматический поиск пересечений:

- Программа вычисляет и отображает цветными маркерами все точки пересечения между любыми двумя фигурами (*например, параболы и прямой, двух парабол и т.д.*).
- Рядом с каждой точкой выводятся её координаты с точностью до двух знаков, что помогает при проверке аналитических решений.

4. Визуальная настройка:

- Настроена удобная система координат с масштабированием и подписями осей.
- Каждый тип пересечений имеет свой цвет (*синий, фиолетовый, красный, жёлтый, белый*), что указано в информационной панели.

Практическая значимость.

Данная программа служит наглядным пособием для подготовки к ЕГЭ. Вместо абстрактного представления о взаимном расположении графиков ученик видит точную картину. Это помогает развить интуицию и проверить правильность аналитического решения уравнений.

Заключение

Разработанное приложение успешно решает поставленную задачу. Оно позволяет быстро и точно визуализировать сложные математические объекты, что делает процесс подготовки к экзамену более наглядным и эффективным.

Титаренко Д.Ю.
Научный руководитель: к. т. н., доц. Рыжкова М.Н.
Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264 г. Муром, Владимирской обл., ул. Орловская, д. 23
E-mail: dmitrijtitar07@gmail.com

Универсальный автопилот БПЛА с адаптивным уклонением от препятствий

Беспилотные летательные аппараты всё шире применяются в гражданских сферах деятельности, часто в сложных и динамически меняющихся средах. Основной задачей в таких условиях является обеспечение автономности полёта в реальном времени при наличии препятствий. В связи с развитием разнообразных сенсоров возникла потребность создавать универсальные и масштабируемые системы управления, способные работать с разным набором датчиков. Современные исследования показывают, что сочетание модульной архитектуры и интеллектуальных алгоритмов управления повышает надёжность БПЛА.

Цель проекта – разработать автопилот БПЛА, который обеспечивает быстрое и безопасное уклонение от препятствий без необходимости переработки всей системы при добавлении новых сенсоров или функциональных модулей. Система должна работать в условиях динамического окружения, выполнять мгновенные манёвры уклонения, сохраняя основную логику полёта. При этом не должно ухудшаться качество базовой навигации при интеграции дополнительных датчиков.

Для достижения поставленной цели предложена двухуровневая архитектура автопилота:

- низкоуровневый контроллер выполняет стабилизацию и следование траектории,
- высокоуровневый вычислитель, компаньон-компьютер, реализует логику планирования и избежания препятствий.

Общение между модулями происходит через стандартизированный протокол ROS2, что обеспечивает модульность и масштабируемость системы. Каждый сенсор подключается как отдельный узел-обработчик. Данные приводятся к единому внутреннему представлению. Это позволяет допускать любые комбинации сенсоров без изменения целостности системы. Для навигационного состояния применяется расширенный фильтр Калмана (EKF), объединяющий инерциальные данные и внешние измерения. Фильтр обеспечивает точную оценку положения и скорости даже при шумных и разреженных измерениях, передавая изменения координат положения БПЛА в пространстве в планировщик. Встроенный в полетный контроллер планировщик прокладывает оптимальную траекторию к цели с учётом известных препятствий. Для динамических препятствий вводится отдельный модуль реального времени – он использует комбинацию динамической системной модуляции (DSM) и гибридных нейронных сетей. Алгоритм работы модуля:

- на вход приходят данные о положении препятствия,
- рассчитывается расстояние до препятствия с помощью DSM,
- если расстояние до препятствия меньше заданного порога, активируется режим уклонения, в противном случае модуль не задействован:
- формируется несколько кандидатных направлений, на основе данных моделей нейронных сетей
- каждое направление оценивается по критерию стоимости J, учитывающему и продолжительность манёвра, и риск столкновения:

$$J = \int_0^T \left(\alpha \parallel v(t) \parallel^2 + \beta\Phi\left(d_{\min}(t)\right)\right) dt, \quad (1)$$

$$\Phi\left(d_{\min}(t)\right) = \begin{cases} \left(d_{\text{safe}} - d_{\min}(t)\right)^2, & d_{\min}(t) < d_{\text{safe}}, \\ 0, & d_{\min}(t) \geq d_{\text{safe}}. \end{cases} \quad (2)$$

где $d_{\min}(t)$ – расстояние до ближайшей помехи, d_{safe} – безопасная дистанция.

- выбирается минимальный по J маршрут.

Динамика полёта модифицируется согласно DSM, что позволяет плавно облететь препятствие. Для ускорения реакции внедрён обучаемый программный контроллер на основе

глубокого RL. Нейросеть, обученная на симуляторных и открытых данных, в автономном режиме получает на вход свёртку данных от LiDAR и камеры и выдаёт смещение желаемого направления полёта. Для дообучения в реальном режиме используется функция награды, которая объединяет положительное поощрение за движение к цели и штрафы за сближение с препятствиями. При таком подходе БПЛА обучается самостоятельно планировать кратчайший манёвр, минимизируя вероятность столкновения. В экспериментах работы «DRL-Based UAV Autonomous Navigation and Obstacle Avoidance» доказано, что SAC-P обеспечивает быструю сходимость и высокую надёжность навигации в запутанных средах.

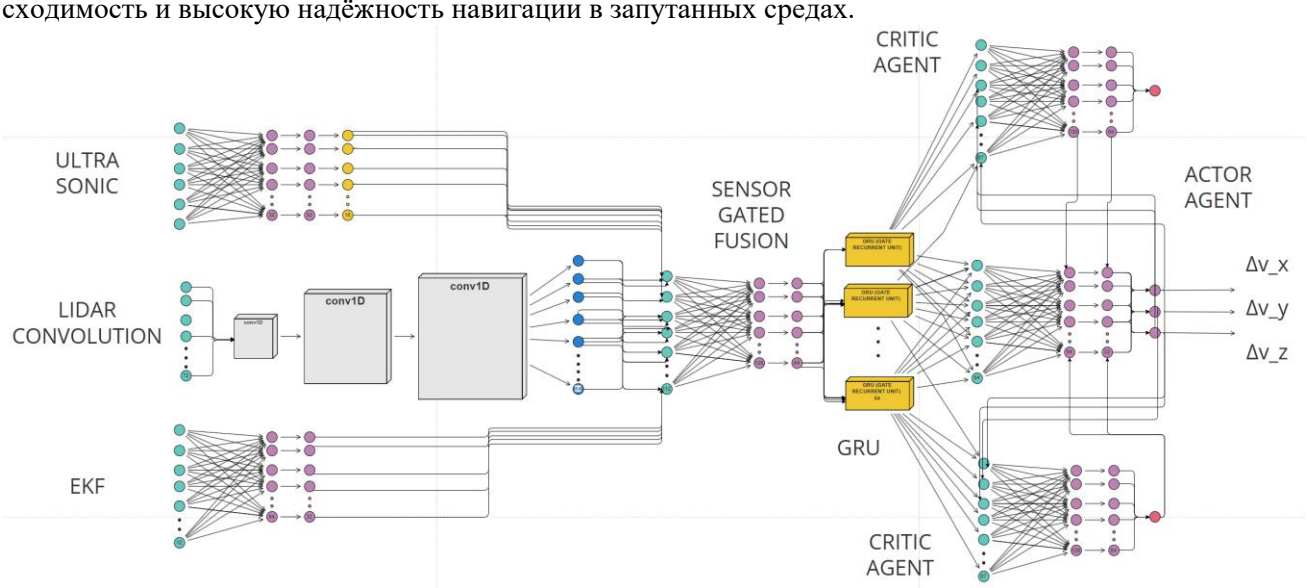


Рис. 1. Диаграмма состава автоматической корректировки курса

Система реализована по схеме SPA и является встроенной в ROS-платформу для поддержки модульности. Каждому сенсору соответствует узел, публикующий данные в унифицированном формате. Узел фильтра Калмана получает все сенсорные потоки и выдаёт оценку состояния с частотой фильтрации. Эти данные поступают в планировщик траектории. Алгоритм оценки траектории основывается на оптимизации интегрального функционала стоимости. В статических условиях используется подход ближайшего свободного маршрута, при обнаружении динамического препятствия алгоритм генерирует новый целевой вектор, используя соображения DSM. Стабильность полёта обеспечивается стандартными регуляторами углов и скоростей, входящими в состав FCU. Периодически (каждые несколько тактов) контроллер пересчитывает план с учётом новых данных. Нейросетевой модуль встроен в средний уровень системы: он непрерывно прогнозирует требуемое смещение траектории. В решении используется классическая формулировка критерия RL:

$$G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1}, \qquad \gamma \in (0,1) \tag{3}$$

где награда r_{t+k+1} включает термины, связанные с расстоянием до цели и штрафом за уменьшение. После обучения агент воспроизводит реакцию с минимальной задержкой.

Таким образом, за счет использования SPA архитектура сохраняет гибкость, а при добавлении нового датчика вводится только соответствующий адаптер, при этом ядро планирования и нейроконтроллера не меняется.

Фадеев Н.Д.
Научный руководитель: к.т.н., доцент кафедры ФПМ Астафьев А.В.
Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
602264 г. Муром, Владимирской обл., ул. Орловская, д. 23
nfadeev433@yandex.ru

Разработка и реализация алгоритма визуализации силуэта на основе анализа информации о состоянии канала связи WI-FI, полученной с помощью микроконтроллеров ESP32

В условиях роста потребности в системах мониторинга присутствия и активности человека в помещениях без нарушения приватности, технологии на основе сигналов Wi-Fi представляют собой перспективную альтернативу традиционным оптическим методам. Использование информации о состоянии канала (Channel State Information, CSI) позволяет извлекать тонкие изменения радиосигнала, вызванные движением и формой человеческого тела, что открывает возможности для реконструкции пространственного образа без применения камер. Однако большинство существующих исследований опирается на специализированное оборудование, такое как сетевые адаптеры или программно-определяемые радиосистемы, что ограничивает практическую применимость разработок. В данной работе решается задача создания доступного алгоритма визуализации силуэта на базе массовых микроконтроллеров ESP32.

Целью работы является разработка и программная реализация алгоритма, преобразующего временные ряды амплитудных и фазовых характеристик сигнала Wi-Fi в двумерные бинарные изображения силуэта человека. Для достижения поставленной цели был сформирован собственный датасет, включающий данные по шести различным позам человека. Сбор данных осуществлялся с использованием пары модулей ESP32, один из которых функционировал в режиме передатчика, а второй — приемника. Для извлечения сырых данных CSI на микроконтроллеры было установлено модифицированное программное обеспечение на базе ESP32-CSI-Tool, обеспечивающее доступ к регистрам физического уровня. Для каждой позы записывались последовательности амплитуд и фаз сигнала, а также соответствующие им эталонные маски силуэтов, полученные путем обработки видеоданных и приведенные к разрешению 64×64 пикселей с нормализацией значений к диапазону [0, 1]. В связи с различной частотой дискретизации каналов (пакеты CSI и кадры масок) применялась процедура согласования, при которой одна маска соотносилась с последовательностью из 20 пакетов CSI.

Ключевым этапом предобработки являлась подготовка последовательностей фиксированной длины для подачи на вход нейронной сети. Был выбран размер временного окна, равный 20 отсчетам, что соответствует одной целевой маске. Исходные данные, представляющие собой векторы длиной 192 значения (полученные путем интерполяции исходных поднесущих до единого размера), преобразовывались к тензорам формы (20, 192, 1) отдельно для амплитудной и фазовой компонент. Датасет был разделен в пропорции 80/20 на обучающую и тестовую выборки.

Архитектура нейронной сети представляет собой двухпоточную сверточную модель с общим декодером, реализующую принцип раннего слияния признаков. Каждый из двух входов — для фазовой и амплитудной компонент — обрабатывается независимым сверточным блоком, состоящим из слоя Conv2D с 64 фильтрами размером 3×3, функцией активации tanh и пакетной нормализацией с параметром импульса 0.3. Полученные признаки конкатенируются и дополнительно нормализуются, после чего применяется операция глобального усреднения по пространственным измерениям (GlobalAveragePooling2D) для агрегации временной информации. Далее следует полносвязный слой с 16 384 нейронами и активацией ReLU, результат которого преобразуется в тензор размером 8×8×256. Декодерная часть сети реализует постепенное восстановление пространственного разрешения посредством трех последовательных блоков, каждый из которых включает операцию увеличения размера (UpSampling2D с коэффициентом 2), сверточный слой с фильтрами 128, 64 и 32

соответственно, активацию tanh и пакетную нормализацию с импульсом 0.5. Финальный сверточный слой с одним фильтром и активацией tanh формирует выходное изображение размером 64×64. Выбор функции активации tanh на выходе обусловлен необходимостью получения значений в диапазоне [–1, 1], которые впоследствии линейно преобразуются к [0, 1] для визуализации. Модель компилировалась с использованием оптимизатора Nadam и функции потерь mean_squared_error, что обеспечивает плавную регрессию значений маски.

Обучение модели проводилось в течение 100 эпох с мониторингом потерь на валидационной выборке. Визуальный анализ кривых обучения демонстрирует устойчивую сходимость процесса без признаков переобучения. Качественная оценка результатов осуществлялась путем сравнения предсказанных масок с эталонными изображениями. Постобработка выходных данных модели включала линейное преобразование (pred + 1) / 2 для приведения значений из диапазона tanh к интервалу [0, 1]. Результаты визуализации показывают, что модель способна восстанавливать общие контуры человеческого силуэта, корректно отражая положение корпуса и конечностей. В качестве доказательства можно привести показатели метрик обучения, которые показаны в таблице 1.

Таблица 1 – Метрики обучения

Метрика	Показатель
MSE	0.2511
IoU	0.1085
Accuracy	0.8053
Precision	0.8607
Recall	0.1105
F1-Score	0.1958

Точность модели после обучения составляет примерно 0.8. Наиболее точные предсказания достигаются для поз с выраженной динамикой, тогда как статичные положения характеризуются несколько размытыми границами.

На рисунке 1 показан график обучения модели.

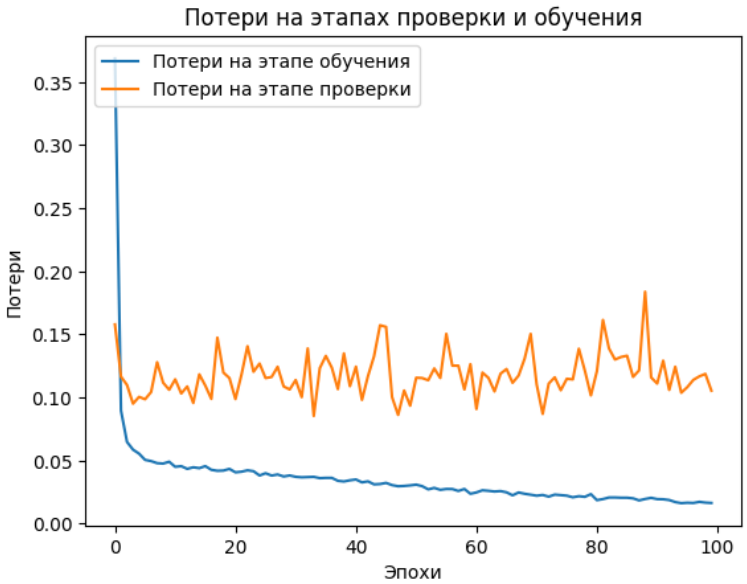


Рис. 1 – График обучения модели

Таким образом, в работе продемонстрирована принципиальная возможность использования микроконтроллеров ESP32 для задач визуализации пространственной формы человека на основе анализа данных CSI. Предложенная архитектура нейронной сети, сочетающая раздельную обработку амплитудно-фазовых компонент и декодер с поэтапным увеличением разрешения, обеспечивает восстановление силуэтов приемлемого качества.

Практическая значимость исследования заключается в создании доступного программного решения, не требующего дорогого специализированного оборудования. Перспективы дальнейшей работы включают расширение датасета за счет большего числа поз и сценариев движения, а также внедрение механизмов для улучшения детализации границ.

Юрлов Н.Е.
Научный руководитель: к.т.н. Белякова А.С.
Муромский институт (филиал) федерального государственного бюджетного образовательного учреждения высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых» 602264 г. Муром, Владимирской обл., ул. Орловская, д. 23
yurlov.nikita.2007@mail.ru

Перспективы развития программных средств контроля за состоянием здоровья при сахарном диабете

По данным Всемирной организации здравоохранения, в мире 422 миллиона человек имеют диагнозом «сахарный диабет». По расчетам Минздрава, в нашей стране это заболевание диагностировано у каждого 15-го россиянина. Около 95 % из них имеют сахарный диабет 2 типа, а остальные – сахарный диабет 1 типа. Пациенты с СД 1 типа для регуляции углеводного обмена подкожно вводят инсулин, а пациенты с СД 2 в основном используют диету и таблетированные препараты. Это объясняется различными особенностями углеводного обмена при этих заболеваниях. Важным и обязательным при этом является контроль показателей глюкозы крови.

Системы непрерывного мониторинга глюкозы крови представляют собой комплекс программно-технических средств: датчик содержит электрохимический сенсор, осуществляющий измерение уровня глюкозы в крови и связанные с ним электронные компоненты. Сенсор находится под кожей в тканевой жидкости, откуда в клетки поступает кислород и полезные вещества, включая глюкозу. Датчик посредством bluethooth сигнала взаимодействует с программным обеспечением, установленном на смартфоне пациента и позволяющим следить за уровнем глюкозы непрерывно. В зависимости от тенденций изменения показателей глюкозы крови пациент регулирует объем инсулинотерапии, что позволяет достигать более качественное компенсации сахарного диабета.

Значения глюкозы крови пациента зависят от количества и состава потребляемой пищи, корректности настроек базисной и болюсной инсулинотерапии, уровня и вида активности пациента, его антропометрических параметров, а также особенностей работы организма конкретного пациента.

Для оценки степень компенсации СД для пациента и для врача важно время нахождения глюкозы крови в диапазонах нормы.

Данная работа посвящена вычислению и анализу статистических характеристик глюкозы крови и их интерпретации. Исходные данные глюкозы крови представляют собой одномерный массив. Для оценки компенсации представляет собой интерес знать количество времени, когда глюкоза крови пациента находилась ниже диапазона нормы и выше диапазона нормы. Результаты работы представлены на рисунке 1:

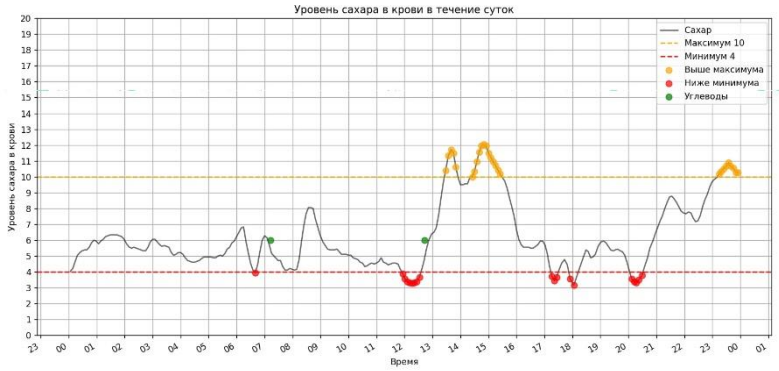


Рис 1. – График глюкозы крови с визуализацией отклонений от нормы

В дальнейшем работы будет направлена на расширение количества характеристик глюкозы крови, интеграции с дневниками питания и инсулинотерапии.