МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ
УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

**ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
ТЕХНОЛОГИЧЕСКОГО ОБРАЗОВАНИЯ**
Кафедра информационных технологий и электронного обучения

Основная профессиональная образовательная программа
Направление подготовки 09.03.01 Информатика и вычислительная техника
Направленность (профиль) «Технологии разработки программного обеспечения»
форма обучения – очная

АНАЛИТИЧЕСКИЙ ОТЧЕТ

«Интеллектуальные системы (Artificial Intelligence)»

Обучающегося 4 курса
Гневнова Артема Евгеньевича

Научный руководитель:
кандидат физико-математических наук,
доцент кафедры ИТиЭО
Жуков Николай Николаевич

Санкт-Петербург
2025

ОГЛАВЛЕНИЕ

1 ТЕОРЕТИЧЕСКИЕ ОСНОВЫ И КЛАССИФИКАЦИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

1.1 Определение и сущность интеллектуальных систем

Интеллектуальная система (ИС) представляет собой сложный программно-аппаратный комплекс, способный находить решение задач, традиционно считающихся прерогативой человеческого разума. В отличие от классических автоматизированных систем, функционирующих по жестко заданным алгоритмам, ИС базируются на оперировании знаниями и механизмах логического вывода [1].

Современный этап развития ИС неразрывно связан с технологиями машинного обучения (Machine Learning), где фундаментом является способность системы к обобщению данных и принятию решений в условиях высокой неопределенности [5]. Анализ учебной литературы позволяет выделить ключевые характеристики, отличающие интеллектуальную систему:

- Наличие базы знаний: хранение не просто разрозненных данных, а семантических связей и правил предметной области [6].
- Способность к самообучению: автоматическая корректировка внутренних параметров модели на основе накопленного опыта без прямого перепрограммирования кода.
- Адаптивность: возможность системы корректно реагировать на новые, ранее не встречавшиеся входные ситуации [1].

1.2 Классификация интеллектуальных систем по архитектурному принципу

Для глубокого понимания состояния проблемы необходимо рассмотреть классификацию ИС, принятую в современной академической среде. Согласно

исследованиям в области компьютерных наук, системы разделяются на три фундаментальных класса [3]:

1. Символьные (классические) системы: базируются на представлении знаний в виде логических правил и декларативных описаний. К этому классу относятся экспертные системы, эффективные в узких, хорошо формализованных областях.

2. Нейросетевые (коннекционистские) системы: строятся по принципу имитации работы биологических нейронов. Именно этот класс систем (нейронные сети) обеспечил технологический прорыв последних лет в области распознавания образов и генерации контента [1].

3. Гибридные системы: объединяют логический вывод символьных систем и гибкость нейронных сетей для достижения максимальной точности и объяснимости принимаемых решений [5].

2 СОСТОЯНИЕ И ПЕРСПЕКТИВЫ ТЕХНОЛОГИЙ ОБРАБОТКИ ЕСТЕСТВЕННОГО ЯЗЫКА (NLP)

2.1 Эволюция подходов: от правил к архитектуре Transformer

Область обработки естественного языка (Natural Language Processing) прошла длительный путь развития, который можно разделить на несколько этапов. Первоначально системы использовали сложные словари и грамматические правила, написанные лингвистами вручную. Главным недостатком такого подхода была невозможность охватить всё многообразие и изменчивость живого человеческого языка [1].

Современное состояние проблемы характеризуется переходом к архитектуре Transformer, которая перевернула индустрию за счет механизма «внимания» (Attention). Этот механизм позволил моделям понимать глубокий контекст предложения, анализируя связи между всеми словами одновременно, что ранее было недоступно для рекуррентных сетей [10].

2.2 Большие языковые модели (LLM) в клиентской поддержке

На текущий момент доминирующее положение занимают большие языковые модели (LLM). Модели уровня GigaChat обучаются на сверхбольших массивах данных, что позволяет им не просто имитировать общение, а понимать логику и интенты (намерения) пользователя [8].

В контексте создания сервисов технической поддержки (на примере ЦБС Петроградского района) LLM решают проблему мгновенного реагирования. Однако анализ практики внедрения выявил проблему «галлюцинаций» — генерации ложных, но правдоподобных фактов. Решением данной проблемы является технология RAG (Retrieval-Augmented Generation). Суть метода заключается в принудительном ограничении области поиска знаний

официальной документацией организации, что гарантирует достоверность выдаваемых инструкций [3][10].

3 АРХИТЕКТУРНЫЕ РЕШЕНИЯ И ПРОТОКОЛЫ ВЗАИМОДЕЙСТВИЯ

3.1 Микросервисная архитектура и облачные технологии в ИИ

Современное состояние индустрии программного обеспечения диктует переход от монолитных структур к микросервисным архитектурам, особенно при разработке систем на базе искусственного интеллекта. В контексте создания AI-сервиса для ЦБС Петроградского района, такой подход позволяет изолировать ресурсозатратные задачи генерации ответов от основной бизнес-логики приложения.

Анализ специализированной литературы по облачным технологиям показывает, что язык Go (Golang) является эталонным инструментом для построения подобных систем благодаря своей модели конкурентности и эффективному управлению памятью [2]. Использование микросервисов позволяет независимо масштабировать компоненты системы: например, при резком росте количества обращений в Telegram-бот можно увеличить число инстансов сервиса обработки промптов, не затрагивая базу данных [11]. Исследования IEEE подтверждают, что интеграция сетевых технологий в сферу ИИ требует гибкой архитектуры для минимизации задержек и оптимизации вычислительных ресурсов [13].

3.2 Эффективность протокола gRPC и бинарной сериализации

Одной из центральных проблем распределенных интеллектуальных систем является скорость обмена данными между сервисами. Традиционные текстовые протоколы (REST/JSON) создают избыточную нагрузку на сеть из-за вербальности формата. Синтез знаний из области системного программирования позволяет выделить протокол gRPC как наиболее эффективное решение [7].

Преимущества использования gRPC в архитектуре AI-сервиса:

- Бинарный формат Protocol Buffers: передача данных в сжатом виде сокращает объем трафика, что критично для передачи длинных контекстных окон LLM [7].
- Двунаправленное потоковое вещание (Streaming): это позволяет системе выдавать ответ пользователю «слово за словом» по мере генерации его нейросетью, что значительно улучшает пользовательский опыт (UX) [6].
- Строгая типизация: использование .proto файлов гарантирует, что сервис бота и ядро ИИ будут всегда использовать идентичные структуры данных, исключая ошибки интеграции [7].

3.3 Реализация технологии RAG в современных ИС

Анализ состояния изученной проблемы указывает на то, что для корпоративных систем (таких как библиотеки или государственные учреждения) недопустимо использование «чистых» LLM из-за их склонности к генерации недостоверных данных. Технология RAG (Retrieval-Augmented Generation) признана стандартом индустрии для решения этой задачи [10]. В рамках данной технологии система сначала выполняет поиск по локальной базе знаний (PostgreSQL), а затем передает найденные факты в качестве контекста для модели GigaChat [8]. Это превращает систему из общего чат-бота в экспертный инструмент, опирающийся исключительно на регламенты и документацию заказчика.

4 ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ И НАДЕЖНОСТИ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ

4.1 Криптографическая защита и сетевая безопасность

При анализе безопасности интеллектуальных систем, интегрированных в сетевую инфраструктуру, на первый план выходит защита передаваемых данных. Согласно актуальным научным публикациям 2023 года, уязвимости в компьютерных сетях могут напрямую влиять на стабильность работы ИИ-агентов [4].

Для нейтрализации этих рисков в проекте предусмотрено использование протоколов TLS 1.2/1.3, которые обеспечивают сквозное шифрование между пользователем, бэкендом на Go и API GigaChat. Это гарантирует конфиденциальность переписки и защиту от атак типа «человек посередине» (MITM). Кроме того, применение современных методов аутентификации позволяет ограничить доступ к системе только авторизованным сотрудникам ЦБС, предотвращая нецелевое использование вычислительных мощностей нейросети.

4.2 Отказоустойчивость и использование контейнеризации

Надежность системы обеспечивается за счет применения технологии контейнеризации Docker. Анализ учебной литературы показывает, что изоляция микросервисов в контейнерах позволяет избежать конфликтов зависимостей и гарантирует идентичность работы системы в среде разработки и на серверах заказчика [2].

Для обеспечения бесперебойного функционирования 24/7 реализуются следующие механизмы:

1. Автоматизированный мониторинг: система отслеживает состояние каждого микросервиса в реальном времени.

2. Health Checks: при обнаружении сбоя в работе сервиса gRPC или подключения к БД, оркестратор автоматически перезапускает упавший контейнер, обеспечивая минимальное время простоя.

3. Логирование и аудит: запись всех запросов и ответов системы (без сохранения персональных данных) позволяет проводить анализ качества работы AI и оперативно исправлять ошибки в промпт-инжиниринге [11].

4.3 Этическое использование и фильтрация контента

Завершая анализ, необходимо отметить важность соблюдения этических норм при эксплуатации ИС. Наряду со встроенными фильтрами GigaChat [8], на уровне бэкенда системы реализуются дополнительные программные фильтры, исключающие генерацию контента, выходящего за рамки профессиональной компетенции техподдержки. Это соответствует современным трендам развития безопасного и ответственного искусственного интеллекта [10].

ЛИТЕРАТУРА

1. Рассел С., Норвиг П. Искусственный интеллект: современный подход. — 4-е изд. — Москва: Вильямс, 2021. — 1408 с. — ISBN 5-8459-0887-6.

2. Титмус, М. А. Облачный Go / М. А. Титмус; перевод с английского А. Н. Киселева. — Москва: ДМК Пресс, 2021. — 418 с. — ISBN 978-5-97060-965-1.

3. Chen G ., Yuan Q. Application and existing problems of computer network technology in the field of artificial intelligence / G. Chen, Q. Yuan // 2021 2nd International Conference on Artificial Intelligence and Computer Engineering (ICAICE). — IEEE, 2021. — DOI: 10.1109/ICAICE54393.2021.00035.

4. Ma, X. Application of artificial intelligence in computer network technology / X. Ma // 2023 2nd International Conference on Artificial Intelligence and Autonomous Robot Systems (AIARS). — IEEE, 2023. — DOI: 10.1109/AIARS59518.2023.00043

5. Баланов, А. Н. Машинное обучение и искусственный интеллект: учебное пособие для вузов / А. Н. Баланов. — 2-е изд., стер. — Санкт-Петербург: Лань, 2025. — 172 с. — ISBN 978-5-507-52891-2.

6. Боровская, Е. В. Основы искусственного интеллекта: учебное пособие / Е. В. Боровская, Н. А. Давыдова. — 6-е эл.изд. — Москва: Лаборатория знаний, 2024. — 130 с. — ISBN 978-5-93208-797-8.

7. «Спецификация Protocol Buffers и gRPC» Documentation | gRPC: сайт. — 2025. — URL: https://grpc.io/docs/ (дата обращения: 24.12.2025).

8. «Документация GigaChat API» GigaChat API: сайт. — 2025. — URL: https://developers.sber.ru/docs/ru/gigachat/api/overview (дата обращения: 24.12.2025).

9. «Язык программирования Go» Documentation – The Go Programming Language: сайт. — 2025. — URL: https://go.dev/doc/ (дата обращения: 25.12.2025).

10. «Тренды в NLP и LLM в 2025 году» Habr: сайт. — 2025. — URL: https://habr.com/ru/articles/981020/ (дата обращения: 25.12.2025).

11. «Просто о микросервисах» Habr: сайт. — 2025. — URL: https://habr.com/ru/companies/raiffeisenbank/articles/346380/ (дата обращения: 25.12.2025).