

## ИНТЕЛЛЕКТУАЛЬНЫЕ ГОЛОСОВЫЕ ПОМОЩНИКИ ДЛЯ СЛУЖБ ТЕХНИЧЕСКОЙ ПОДДЕРЖКИ

В. А. Орлов<sup>1,a</sup>, А. В. Гавриленко<sup>2,b</sup>

<sup>1</sup> Сургутский филиал федерального государственного автономного учреждения «Федеральный научный центр Научно-исследовательский институт системных исследований Национального исследовательского центра «Курчатовский институт», г. Сургут, Российская Федерация

<sup>2</sup> Сургутский государственный университет, г. Сургут, Российская Федерация

<sup>a</sup> ORCID: <https://orcid.org/0009-0000-6742-1289>, ✉ [upsvadim@gmail.com](mailto:upsvadim@gmail.com)

<sup>b</sup> ORCID: <https://orcid.org/0000-0003-1837-5698>, [gavrilenko.anna.v@gmail.com](mailto:gavrilenko.anna.v@gmail.com)

*Аннотация:* в статье проведен комплексный обзор теоретических аспектов проектирования интеллектуальной автоматизированной системы — голосового помощника для служб технической поддержки. Рассмотрены ключевые компоненты голосового взаимодействия: автоматическое распознавание речи (ASR), обработка естественного языка (NLP), механизмы диалогового управления и синтеза речи (TTS). Особое внимание уделено архитектурным подходам к построению подобных систем, включая монолитную, клиент-серверную и микросервисную архитектуры, с выделением преимуществ и ограничений каждой. На основе анализа современных технологических решений обоснована целесообразность применения микросервисного подхода в условиях высоконагруженных сервисных сред. Проведен сравнительный анализ существующих голосовых ассистентов, адаптированных под русскоязычную аудиторию, с точки зрения их применимости в автоматизации процессов технической поддержки.

*Ключевые слова:* голосовой помощник, техническая поддержка, автоматизированная система, обработка естественного языка, микросервисная архитектура.

*Благодарности:* работа выполнена в рамках государственного задания НИЦ «Курчатовский институт» — НИИСИ по теме № FNEF-2024-0001 «Создание и реализация доверенных систем искусственного интеллекта, основанных на новых математических и алгоритмических методах, моделях быстрых вычислений, реализуемых на отечественных вычислительных системах» (1023032100070-3-1.2.1).

*Для цитирования:* Орлов В. А., Гавриленко А. В. Интеллектуальные голосовые помощники для служб технической поддержки. *Успехи кибернетики*. 2025;6(3):96–104.

*Поступила в редакцию:* 03.08.2025.

*В окончательном варианте:* 11.09.2025.

## INTELLIGENT VOICE ASSISTANTS FOR TECHNICAL SUPPORT SERVICES

V. A. Orlov<sup>1,a</sup>, A. V. Gavrilenko<sup>2,b</sup>

<sup>1</sup> Surgut Branch of Scientific Research Institute for System Analysis of the National Research Centre “Kurchatov Institute”, Surgut, Russian Federation

<sup>2</sup> Surgut State University, Surgut, Russian Federation

<sup>a</sup> ORCID: <https://orcid.org/0009-0000-6742-1289>, ✉ [upsvadim@gmail.com](mailto:upsvadim@gmail.com)

<sup>b</sup> ORCID: <https://orcid.org/0000-0003-1837-5698>, [gavrilenko.anna.v@gmail.com](mailto:gavrilenko.anna.v@gmail.com)

*Abstract:* we reviewed the theoretical foundations of designing an intelligent automated system – a voice assistant for technical support services. We examined the key components of voice interaction: automatic speech recognition (ASR), natural language processing (NLP), dialog management, and text-to-speech (TTS). We analyzed architectures of such systems, including monolithic, client-server, and microservice architectures, and highlighted the advantages and limitations of each. Based on this analysis, we validated the use of a microservice approach in high-load service environments. We also conducted a comparative analysis of existing voice assistants adapted for Russian-speaking users in terms of their applicability to technical support automation.

*Keywords:* voice assistant, technical support, automated system, natural language processing, microservice architecture.

*Acknowledgements:* this study is a part of the FNEF-2024-0001 government order contracted to the Scientific Research Institute for System Analysis of the National Research Centre “Kurchatov Institute”, project No. 1023032100070-3-1.2.1 Development and Implementation of Trusted Artificial Intelligence Systems Based on new Mathematical Methods and Algorithms, Fast Computing Models for Domestic Computing Systems.

*Cite this article:* Orlov V. A., Gavrilenko A. V. Intelligent Voice Assistants for Technical Support Services. *Russian Journal of Cybernetics*. 2025;6(3):96–104.

*Original article submitted:* 03.08.2025.

*Revision submitted:* 11.09.2025.

## **Введение**

Сегодня мы становимся свидетелями переломного момента: общение человека с машиной все больше напоминает диалог с живым собеседником. Если раньше обращение в службу поддержки ассоциировалось с долгим ожиданием на линии и частым переключением между специалистами, то уже завтра на вопросы пользователей сможет мгновенно отвечать интеллектуальный голосовой помощник — быстрый, точный и всегда готовый к диалогу.

Рост потока обращений в ИТ-подразделения крупных предприятий закономерен. Цифровизация охватывает все новые процессы, расширяется спектр сервисов, растет число пользователей. При этом большая доля запросов остается типовой: она не требует тщательной экспертизы, но ежедневно отнимает часы труда операторов.

Есть и другая проблема, которая редко упоминается. На крупных предприятиях десятилетиями формировались колоссальные базы знаний — инструкции, регламенты, решения типовых ошибок. Эти массивы данных — настоящие «кладовые опыта», но зачастую они используются в ручном режиме. Сотруднику приходится искать нужную информацию в некоторой базе знаний, по ключевым словам, после чего предоставлять клиенту содержащееся там решение в ручном режиме.

В связи с этим был разработан подход — подключение интеллектуального голосового ассистента напрямую к корпоративным базам знаний. В этом случае пользователь получает не сухую ссылку на документ, а конкретный ответ в привычной речевой форме, а оператор освобождается от рутинных операций, связанных с поиском. Такой симбиоз технологий ASR, NLP и TTS с корпоративными базами знаний позволяет превратить накопленный опыт предприятия в живой инструмент.

В центре внимания настоящей работы — процессы технической поддержки и их трансформация под воздействием голосовых интерфейсов, интегрированных с корпоративными базами знаний.

## **Материалы и методы**

В центре внимания авторов данной статьи находится программная архитектура голосового помощника, ориентированная на функционирование в условиях высоконагруженных служб технической поддержки. Выдвигается гипотеза, что сочетание микросервисной организации, современных методов распознавания речи и унифицированных стандартов интеграции позволит разработать решение, которое обеспечит высокую точность ответов, удобство взаимодействия с пользователями и при этом минимальные затраты на масштабирование.

Главными методами исследования выступили системно-структурный анализ, позволяющий описать логику работы ключевых модулей и их взаимосвязей, сравнительный обзор существующих технологических стеков, а также обзорно-аналитический метод — для систематизации теоретических знаний о технологиях распознавания, синтеза речи и обработки естественного языка. Источниками информации послужили научные статьи, техническая документация платформ ASR/NLP/TTS, а также данные об архитектурных паттернах, применяемых в промышленности.

## **Механизмы функционирования служб технической поддержки**

Службы технической поддержки реализуют многоуровневую модель обработки запросов (L1–L3), рекомендованную ITIL:

1. L1 — первичный контакт с пользователем, регистрация и классификация запроса.
2. L2 — углубленная диагностика и устранение инцидентов.
3. L3 — привлечение экспертов и разработчиков для решения нестандартных задач.

Схематично порядок обращения пользователей в службу технической поддержки посредством системы Service Desk можно представить в общем виде (рис. 1).

## **Технологии распознавания речи (ASR)**

В классической архитектуре систем автоматического распознавания речи принято выделять три взаимосвязанных функциональных блока.



**Рис. 1.** Порядок исполнения обращения в службу технической поддержки

В первом блоке происходит предварительная аудиообработка. Сырые данные, поступающие с микрофона, проходят каскад фильтрации, в который, помимо подавления стационарных и импульсных помех, обычно включают выравнивание динамического диапазона и компенсацию канал-зависимых искажений. Целью служит формирование максимально чистого сигнала для дальнейшего анализа.

Вторым этапом идет параметризация (извлечение акустических признаков). После предварительной очистки речевой поток разбивается на перекрывающиеся кадры, каждый из которых описывается компактным вектором дескрипторов. Наиболее распространенным выбором остаются мел-частотные кепстральные коэффициенты, дополненные дельта- и дельта-дельта-параметрами, однако все чаще используют и альтернативные спектро-темпоральные представления, адаптированные под конкретное оборудование и акустические условия.

Заключительным действием является лингвистическое декодирование. На этом этапе система отыскивает такую последовательность лексических единиц, которая с наибольшей апостериорной вероятностью объясняет наблюдаемые векторы признаков. Оптимизационная задача решается одной из модификаций алгоритма Витерби, а для ограничения вычислительных затрат применяют лучевой поиск и прочие эвристики. Итоговый гипотезный граф содержит не только финальный транскрипт, но и альтернативные пути, что упрощает постфактискую переоценку.

Современные алгоритмы автоматического распознавания речи (АРР) базируются на статистических моделях и методах машинного обучения. Классическая формулировка задачи распознавания — найти такую последовательность слов  $W$ , которая максимизирует вероятность  $P(X|W)$  при данном акустическом наблюдении  $X$ . Согласно правилу Байеса это эквивалентно максимизации произведения  $W = \operatorname{argmax} P(W) P(X|W)$ , где  $P(X|W)$  — вероятность порождения акустического наблюдения при данной словесной гипотезе (так называемая акустическая модель), а  $P(W)$  — языковая модель, отражающая априорную вероятность данной последовательности слов в языке [1]. Иначе говоря,  $P(X|W)$  моделирует соответствие звуков и фонем/слов, а  $P(W)$  — частотность и связность слов в речи. Такой подход позволяет разделить проблему на две части: акустическое моделирование и языковое моделирование.

### Обработка естественного языка

После этапа автоматического распознавания речи, приводящего акустический сигнал к текстовому представлению, выполняется многоуровневая обработка естественного языка (Natural Language Processing, NLP). Задача данного блока заключается в формировании формализованной репрезентации исходного текста, пригодной для последующего информационно-аналитического этапа системы.

В общем случае лингвистический конвейер русскоязычных систем включает три последовательно выполняемых процедуры:

- морфологический анализ;
- синтаксический разбор;
- семантическую интерпретацию.

Совокупность исполнения этих этапов обеспечивает формирование структурированного представления транскрибированного текста, что является необходимым условием для эффективного функционирования последующих когнитивных модулей (диалогового менеджера, модулей рекомендаций, аналитических подсистем).

### Механизмы рассуждения и принятия решений

После того как модуль понимания речи сформировал семантически размеченную реплику пользователя, система переходит к этапу решения (reasoning). Именно на данной стадии определяется, какое действие необходимо выполнить либо какой текст следует сгенерировать в ответ. В литературе такой компонент нередко обозначается термином «диалоговый управляющий» или «процессор рассуждения».

В наиболее общем виде задача сводится к выбору реакции  $r$  из множества возможных  $R$ , максимизирующей некоторый критерий серьезности/полезности:

$$\hat{r} = \arg \max_{r \in R} F(r, c, \delta),$$

где  $c$  — контекст диалога, а  $\delta$  — текущая семантическая репрезентация реплики. На практике функционал  $F$  варьируется от простого сопоставления с шаблоном до сложного стохастического функционала, включающего прагматические факторы и вероятностные оценки успешности.

Существующие решения условно группируются в две большие категории, каждая из которых имеет собственные достоинства и ограничения. Описание категорий представлено в таблице 1.

Таблица 1

Категории механизмов принятия решений

Категория	Основная идея	Типичные инструменты
Экспертные (правилловые) системы	Расчет реакции по набору детерминированных правил IF–THEN, организованных в дерево или граф знаний	Продукционные системы, Prolog-подобные движки, Graph-based dialog managers
Нейросетевые (обучаемые) модели	Выбор или генерация ответа осуществляется нейросетью, обученной на парах «ситуация–действие» либо «контекст–ответ»	Seq2Seq LSTM, Transformer (GPT-семейство), Reinforcement Learning-based DM

В современных промышленных ассистентах чаще встречается гибридный вариант: экспертное ядро отвечает за критично важные сценарии (оплата, подписка), а нейросетевой компонент покрывает свободный разговор и обрабатывает непредусмотренные запросы.

### Синтез речи

Последним этапом голосового интерфейса является синтез речи, или технология Text-to-Speech (TTS) — преобразование произвольного текста в искусственно сгенерированную устную речь. Именно благодаря синтезу речи голосовые помощники могут общаться с пользователем в привычной акустической форме. К синтезу предъявляются высокие требования: речь на выходе должна быть разборчивой, естественной и интонационно выразительной, чтобы пользователь воспринимал голос ассистента комфортно. Разработка систем TTS — сложная инженерная задача, в которой сочетаются знания лингвистики (фонетики, просодии) и методы цифровой обработки сигналов и машинного обучения.

Примерно с 2016 года происходит революция в TTS благодаря глубокому обучению. Появляются нейросетевые модели, способные напрямую преобразовывать текст в звучание, минуя разбиение на фрагменты. Одной из первых прорывных разработок стал алгоритм Tacotron (Google, 2017), а затем

Tacotron 2 (2018) — нейросетевая архитектура, которая принимает на вход текст и вырабатывает спектрограмму (представление звука) с заданным произношением и интонацией. По сути Tacotron заменил собой классический каскад «фонетический анализ → просодика → генерация сигналов» на единую нейросеть последовательного декодирования [2]. Для получения конечного звука спектрограмма затем прогоняется через нейросетевой вокодер — например, алгоритм WaveNet (DeepMind, 2016) или его упрощенные варианты (Parallel WaveGAN, WaveRNN). WaveNet — это глубокая сверточная сеть, генерирующая аудиосигнал по одной выборке за раз, способная создавать очень натуральное звучание. Комбинация Tacotron 2 + WaveNet задала новый стандарт: качество синтезированной речи стало сопоставимо с записью человека по критерию MOS (Mean Opinion Score) в пределах 4.0–4.5 из 5.0. Позже появились улучшения: FastSpeech (Microsoft, 2019) — модель, убирающая авторегрессию Tacotron и позволяющая синтезировать речь значительно быстрее, Flowtron и др.

Отдельно исследуются способы управления голосовыми параметрами: например, задавать эмоциональную окраску, стиль речи, голосовой тембр. Для русского языка многие из этих архитектур были адаптированы энтузиастами и компаниями. Так, в 2020 г. компанией Tinkoff были открыты модели на базе Tacotron для русского языка (проект Tinkoff TTS на GitHub), появились голоса на основе VITS (2021, объединенная энд-ту-энд модель спектрограммы и вокодера на основе VAE-GAN). В академической среде упоминаются эксперименты с GAN-сетью WGANsG для синтеза речи, но пока нейросети GAN показали менее устойчивое качество по сравнению с авторегрессивными и поточными моделями. Как работает современный синтез речи? Он тоже модульный, хотя модули теперь могут быть внутри одной нейросети. Сначала выполняется предобработка текста: нормализация (преобразование чисел, сокращений в развернутый вид — например, «стр. 5» → «страница пятая»), расстановка ударений. Последнее для русского языка критично: от позиции ударения зависит звучание гласных.

Некоторые системы используют словари ударений, другие — обучают модель предсказывать ударение в слове по контексту. Далее текст разбивается на последовательность входных символов — чаще всего букв (графемы), иногда с пометками ударений, или сразу в фонемы. Затем основной блок — акустическая модель TTS — преобразует последовательность символов в акустические признаки речи. Современные Tacotron-подобные модели генерируют мел-спектрограмму (распределение энергии по частотам во времени). Этот выход далее поступает в вокодер, который преобразует спектрограмму в звуковую волну (PCM-сигнал). Вокодеры могут быть классическими (гриффин-лимовский алгоритм и фильтрация) или нейросетевыми (WaveNet, WaveGlow, HiFi-GAN и т.д.). Нейровокодеры, обученные на реальных аудио, как правило, дают гораздо более естественный тембр и интонационные нюансы. Например, WaveNet генерирует микроскопические особенности голоса человека, делая речь практически неотличимой на слух.

### Сравнительный анализ голосовых помощников для служб технической поддержки

Для сравнительного обзора были взяты следующие системы голосового взаимодействия для служб технической поддержки:

1. Алиса — голосовой помощник, запущенный Яндексом 10 октября 2017 года.
2. Ассистенты Джой и Афина из семейства Салют, которые стали первыми в России виртуальными ассистентами, способными разговаривать при помощи генеративной модели естественного языка ruGPT-3 с 760 млн параметров.
3. Голосовой помощник Олег от Т-Банка, который появился в июне 2019 года.

Для сравнения данных решений была составлена таблица 2, в которой описаны ключевые достоинства и недостатки представленных в анализе систем.

Все три решения демонстрируют высокую эффективность в автоматизации клиентских коммуникаций и пользовательской поддержки. Яндекс Алиса выигрывает благодаря широкому кругозору и способности вести гибкие диалоги; ее разумно выбирать, когда важны эрудиция ассистента и высокая узнаваемость бренда. Салют от Сбера выделяется глубокой интеграцией с государственными и корпоративными сервисами, поэтому подходит для проектов, где критична локальная адаптация и тесная связь с обширной экосистемой, а также возможность варьировать голосовой образ. Олег от Тинькофф — узкоспециализированный ассистент, идеально заточенный под финансовую поддержку и телефонные сценарии: его стоит рассматривать, если приоритетом являются точность распознавания речи и выверенные скрипты звонков. При выборе платформы для технической поддержки важно учитывать специфику задач. При этом ни одно из существующих сегодня решений не удовлетворяет

Таблица 2

## Сравнительный анализ аналогов

Критерии сравнения	Яндекс Алиса	Олег (Тинькофф)	Сбер Салют
1. Понимание русского языка (естественная речь)	+	+	+
2. Наличие офлайн-режима	-	-	-
3. Открытость платформы (доступность SDK/навыков)	+	-	+
4. Интеграция с экосистемой (банковской / государственной)	+/-	+	+
5. Универсальность использования (вне родной экосистемы)	+	-	+/-
6. Качество синтеза речи (натуральность голоса)	+	+	+/-
7. Адаптация под бренд (возможность изменить голос/образ)	-	+	+/-
8. Специализация на бизнес-сценарии	+/-	+	+
9. Точность АРР (автоматическое распознавание речи)	+	+	+
10. Удобство разработки (низкий порог входа)	+	-	-

требованиям к автономной офлайн-работе и бесшовному внедрению в ИТ-инфраструктуру промышленного предприятия.

### Выбор и обоснование архитектуры системы

При разработке архитектуры голосового помощника для внутренней службы поддержки необходимо рассмотреть несколько вариантов архитектурных подходов: монолитная архитектура, классическая клиент-серверная модель и современная микросервисная архитектура. Ниже проводится сравнительный анализ этих вариантов с точки зрения требований системы, после чего обосновывается выбор оптимального решения.

Исходя из предъявляемых к системе требований: селективное масштабирование компонентов, бесшовная интеграция с внешними сервисами и управляемое развитие функционала — наиболее адекватным решением видится микросервисная архитектура. Для подтверждения этого вывода произведен сопоставительный анализ преимуществ и ограничений монолитного и микросервисного подходов. В таблице 3 представлено влияние каждой архитектуры на производительность, поддерживаемость, гибкость масштабирования и стойкость к изменениям окружения.

Таким образом, альтернативные подходы либо слишком монолитны (затрудняют масштабирование и модификацию), либо не обеспечивают достаточной модульности. Микросервисы позволяют естественно разделить функциональность голосового помощника на самостоятельные части и получить требуемую гибкость. В дальнейшем изложении в качестве целевого решения рассматривается микросервисная архитектура, а все компоненты системы будут выступать как отдельные сервисы, взаимодействующие по внутренним API.

### Результаты

В итоге можно сделать вывод, что наилучшие показатели устойчивости и качества работы голосового помощника достигаются при использовании микросервисной архитектуры, где каждый функциональный модуль — от распознавания речи до аналитики пользовательских запросов — существует в виде отдельного сервиса. Такой подход обладает следующими преимуществами:

1. Гибкое масштабирование: при увеличении потока обращений можно оперативно добавить вычислительные ресурсы только для тех модулей, которые испытывают повышенную нагрузку.
2. Отказоустойчивость: выход из строя одного модуля не приводит к полной недоступности системы, что критично для служб технической поддержки.
3. Упрощение обновлений: модернизация или дообучение языковых моделей может производиться без остановки всего комплекса.

Предлагаемая архитектура голосового помощника состоит из цепочки связанных микросервисов (рис. 2):

1. Сервис распознавания речи.
2. Сервис обработки естественного языка.

Таблица 3

## Сравнение архитектурных подходов к построению ПО

Критерий сравнения	Монолитная архитектура	Клиент-серверная архитектура	Микросервисная архитектура
Масштабируемость	Ограниченная	Средняя	Высокая
Отказоустойчивость	Низкая	Средняя	Высокая
Гибкость обновления	Низкая	Средняя	Высокая
Сложность разработки	Низкая (на начальных этапах)	Средняя	Высокая (на начальных этапах)
Сложность поддержки	Высокая	Средняя	Средняя
Использование ресурсов	Эффективное на малых проектах	Среднее	Среднее
Применимость для голосового помощника	Простые решения	Средние решения	Сложные решения
Производительность системы	Высокая	Высокая на сервере, возможны задержки	Средняя, возможны задержки
Легкость интеграции	Низкая	Средняя	Высокая

3. Диалоговый менеджер.
4. API шлюз.
5. Сервис генерации голоса.
6. Файловое хранилище.

Голосовой запрос от пользователя поступает в первый сервис, в котором голос преобразуется в текст, который подается на вход в сервис обработки естественного языка. В данном микросервисе происходит обработка текста, результатом работы которого являются выделенные смысловые объекты, которые подаются в диалоговый менеджер. В диалоговом менеджере происходит основная работа с полученной информацией. Именно в нем осуществляется поиск решения в имеющейся базе знаний, создается тикет в систему поддержки пользователей, определяется ответ пользователю. Полученная информация затем передается в сервис генерации ответов, на выходе из которого пользователь уже может получить готовый ответ в виде сгенерированного голоса. Таким образом реализуется полный цикл взаимодействия пользователя с системой: вопрос пользователя (голосом) — автоматический поиск ответа — голосовой ответ пользователю с возможностью эскалации при отсутствии ответа.

Также с целью разрешения спорных ситуаций предусмотрена система хранения диалога между пользователем и системой. Сервисом распознавания речи и сервисом генерации ответов сохраняются имеющиеся в них аудиопотоки в систему хранения данных. Это является наиболее подходящей структурой, потому что не требует необходимости передавать данные по всей сети системы.

Разработанная концептуальная модель демонстрирует целостный и логически выстроенный процесс обработки обращения от пользователя. Каждый сервис системы выполняет определенную задачу и передает результаты далее по цепочке, что обеспечивает гибкость и масштабируемость решения. За счет прозрачности, последовательности операций и строгого разграничения обязанностей между модулями система остается управляемой и удобной для дальнейшей модификации. Таким образом, предложенная архитектура создает надежный фундамент для построения эффективных и адаптируемых голосовых помощников, способных обрабатывать значительные объемы запросов, интегрироваться с различными внешними системами и эволюционировать в соответствии с изменяющимися требованиями бизнеса и пользователей.

Кроме того, выявлено, что индекс процентной ошибки распознавания слов (Word Error Rate) существенно снижается при регулярном дообучении моделей на отраслевых терминах и реальных

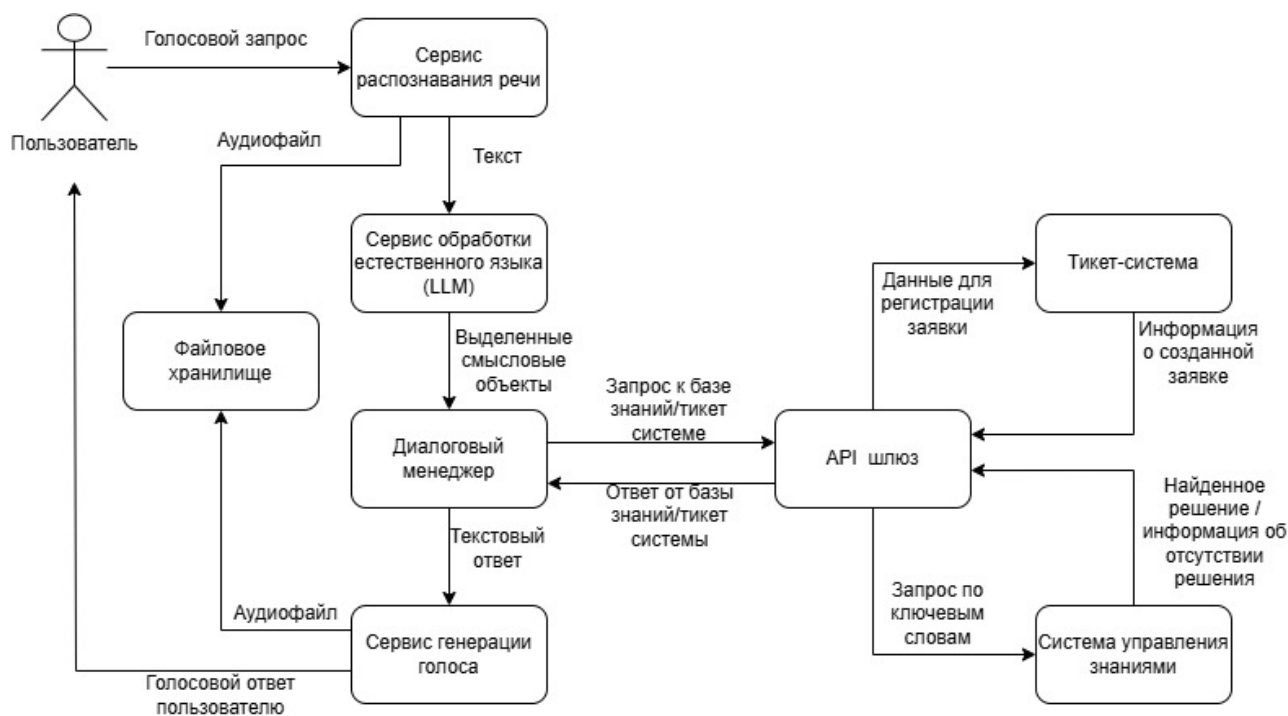


Рис. 2. Концептуальная модель системы

пользовательских диалогах. В среднем, если показатель ошибок уменьшается на 5–10 процентов, это благоприятно сказывается на качестве обслуживания и восприятии голосового помощника пользователями.

Интеграцию с корпоративными базами знаний и системами управления заявками (система управления взаимоотношениями с клиентами, тикет-сервисы и другие) рекомендуется осуществлять через единый шлюз приложений (API-шлюз), использующий REST или gRPC. Это упрощает архитектуру и дает возможность подключать новые сервисы с минимальными изменениями в существующих модулях. Крайне важно обеспечить защиту передаваемых данных, особенно если речь идет о конфиденциальной информации клиентов, например, путем шифрования соединений и разграничения прав доступа.

Сопоставление полученных данных с результатами предыдущих исследований подтверждает, что заявленная гипотеза о высокой эффективности комплексного подхода при проектировании голосового помощника, в целом, верна. Новизна представленных решений заключается в комплексном учете специфики технических запросов и необходимости обеспечения масштабируемости и надежности в условиях реального бизнеса.

### Заключение

Исследование охватило несколько ключевых направлений. Прежде всего, был выполнен всесторонний анализ предметной области и существующих подходов к автоматизации служб поддержки. Подробно изучены механизмы их функционирования, что позволило выделить типовые сценарии обработки обращений. Отдельное внимание уделено современным технологиям распознавания и синтеза речи, а также инструментам обработки естественного языка, которые сегодня формируют основу эффективного голосового взаимодействия.

Кроме того, был проведен сравнительный обзор уже существующих решений, что дало возможность выявить их сильные и слабые стороны. На этой базе разработана собственная концепция: спроектирована автоматизированная система, обоснован выбор архитектуры, подготовлены концептуальная и функциональная модели.

Перспективы развития проекта связаны с расширением языковых моделей на основе отраслевых текстовых корпусов, внедрением методов активного обучения для оперативного пополнения базы сценариев, а также интеграцией с системами телефонной аналитики. Последнее направление открывает новые возможности. Оценка эмоций и уровня удовлетворенности абонента позволит превратить

голосового помощника из инструмента автоматизации в полноценного участника диалога, способного не только решать задачи, но и учитывать эмоциональный контекст общения.

### ЛИТЕРАТУРА

1. Тампель И. Б., Карпов А. А. *Автоматическое распознавание речи*: учеб. пос. СПб: Университет ИТМО; 2017. 152 с.
2. Как устроена Алиса. Лекция Яндекса. *Сообщество IT-специалистов Хабр*. Режим доступа: <https://habr.com/ru/companies/yandex/articles/349372/>.
3. Кравченко Д. А. Микросервисная архитектура. *Интерактивная наука*. 2022;4. Режим доступа: <https://cyberleninka.ru/article/n/mikroservisnaya-arhitektura>.
4. Пацей Н. В., Шитько А. М. Интеграция микросервисов на основе RPC. *Эпоха науки*. 2021;27:32–37.
5. Бужин И. Г., Деревянкин А. Ю., Антонова В. М., Перевалов А. П., Миронов Ю. Б. Эффективность фреймворков для передачи информации в виртуализированной сети связи с микросервисной архитектурой. *Научные технологии в космических исследованиях Земли*. 2023;2:23–31.
6. Бондаренко Д., Данилов А. В. Виртуальные ассистенты Сбер и их особенность. *Экономика и инновации*: сб. ст. участников межвузовской научно-практической конференции. В 3 т. Т. 3. Москва: РЭУ им. Г. В. Плеханова; 2023:11–16.
7. Как в Тинькофф разрабатывают навыки голосового ассистента Олега. *Сообщество IT-специалистов Хабр*. Режим доступа: <https://habr.com/ru/companies/tbank/articles/650381/>.
8. Chandler H. Microservices vs. Monolithic Architecture. *Atlassian*. Режим доступа: <https://www.atlassian.com/microservices/microservices-architecture/microservices-vs-monolith>.